Abstract and Applied Analysis

Artificial Intelligence and Data Mining: Algorithms and Applications

Guest Editors: JianHong (Cecilia) Xia, Fuding Xie, Yong Zhang, and Craig Caulfield

Artificial Intelligence and Data Mining: Algorithms and Applications

Abstract and Applied Analysis

Artificial Intelligence and Data Mining: Algorithms and Applications

Guest Editors: Jianhong (Cecilia) Xia, Fuding Xie, Yong Zhang, and Craig Caulfield

Editorial Board

Contents

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 524720, 2 pages
http://dx.doi.org/10.1155/2013/524720

Editorial

Artificial Intelligence and Data Mining: Algorithms and Applications

Jianhong (Cecilia) Xia,[1] Fuding Xie,[2] Yong Zhang,[3] and Craig Caulfield[4]

[1] *Curtin University, Perth, WA, Australia*
[2] *School of Urban and Environmental Sciences, Liaoning Normal University, Dalian, China*
[3] *School of Computer and Information Technology, Liaoning Normal University, Dalian, China*
[4] *Edith Cowan University, Perth, WA, Australia*

Correspondence should be addressed to Jianhong (Cecilia) Xia; c.xia@curtin.edu.au

Received 20 June 2013; Accepted 20 June 2013

Artificial intelligence and data mining techniques have been used in many domains to solve classification, segmentation, association, diagnosis, and prediction problems. The overall aim of this special issue is to open a discussion among researchers actively working on algorithms and applications. The issue covers a wide variety of problems for computational intelligence, machine learning, time series analysis, remote sensing image mining, and pattern recognition. After a rigorous peer review process, 21 papers have been selected from 38 submissions. The accepted papers in this issue addressed the following topics: (i) advanced artificial intelligence and data mining techniques; (ii) computational intelligence in dynamic and uncertain environments; (iii) machine learning on massive datasets; (iv) time series data analysis; (v) Spatial data mining: algorithms and applications.

Among them, there are seven papers on new algorithm model design and optimisation. In "*Dictionary learning based on nonnegative matrix factorization using parallel coordinate descent*" by Z. Tang et al., the authors propose a novel method for learning a nonnegative, overcomplete dictionary for sparse representation of nonnegative signals. In "*A cost-sensitive ensemble method for class-imbalanced datasets*" by Y. Zhang and D. Wang, a cost-sensitive ensemble method is developed to solve imbalanced data classification. The proposed method is based on cost-sensitive support vector machine (SVM) and query by committee (QBC). In "*Vision target tracker based on incremental dictionary learning and global and local classification*" by Y. Yang et al., a robust global and local classification algorithm for visual target tracking in uncertain environment is suggested based on sparse representation. In "*Analysis of similarity/dissimilarity of DNA sequences based on chaos game representation*" by W. Deng and Y. Luan, the authors construct three kinds of CGR spaces and describe a DNA sequence by CGR-walk model. As an application, the authors compare the similarity/dissimilarity of exon-1 of β-globin genes for nine species. In "*A real-valued negative selection algorithm based on grid for anomaly detection*" by R. Zhang et al., a GB-RNSA algorithm is proposed for anomaly detection. In "*An enhanced Wu-Huberman algorithm with pole point selection strategy*" by Y. Sun and S. Ding, a novel pole point selection strategy for the Wu-Huberman algorithm is developed to filter pole points by introducing a sparse rate. Finally, in "*A novel bat-inspired krill herd algorithm for solving numerical optimization problems*" by G.-G. Wang et al., an effective bat-inspired krill herd (BKH) method is proposed to enhance the performance of numerical optimization methods.

Several authors deal with different aspects of time series analysis. In "*Piecewise trend approximation: a ratio-based time series representation*" by J. Dan et al., a time series representation PTA is developed to improve the efficiency of time series data mining in high dimensional large databases. In "*A dynamic fuzzy cluster algorithm for time series*" by M. Ji et al., a dynamic fuzzy cluster (DFC) is proposed based on improved a Fuzzy C-Means (FCM) algorithm and key points. The proposed algorithm works by determining

those time series whose class labels are vague and further partitions them into different clusters over time. In "*A new strategy for short-term load forecasting*" by Y. Yang et al., a hybrid model based on the seasonal ARIMA model and BP neural network is presented to improve the short-term load forecasting accuracy.

Papers collected in this special issue also focus on spatial data mining: algorithms and applications. In "*Ecological vulnerability assessment integrating the spatial analysis technology with algorithms: a case of the wood-grass ecotone of Northeast China*" by Z. Qiao et al., an assessment model of ecological vulnerability is developed using the analytical hierarchy process and a spatial analysis method. In "*Algorithms and applications in grass growth monitoring*" by J. Liu et al., a double logistic function-fitting algorithm is used to retrieve phenophases for grasslands in Northern China from a consistently processed Moderate Resolution Imaging Spectroradiometer (MODIS) dataset, and the accuracy of the satellite-based estimates is assessed using field phenology observations. Results show that the proposed method is valid for accurately identifying vegetation phenology. In "*The sustainable island development evaluation model and its application based on the nonstructural decision fuzzy set*" by Q. Wang et al., the authors discuss and establish a sustainable development indicator system and model and adopt a entropy method and the nonstructural decision fuzzy set theoretical model to determine the weight of the evaluating indicators. In "*Spatiotemporal simulation of tourist town growth based on the cellular automata model: the case of Sanpo town in Hebei province*" by J. Yang et al., the authors use a tourism urbanization growth model to simulate and predict the spatiotemporal growth of Sanpo town in Hebei province. In "*Model for the assessment of seawater environmental quality based on multiobjective variable fuzzy set theory*" by L. Ke and H. Zhou, a model based on a multiobjective variable fuzzy set theory is presented to evaluate seawater environmental quality. In "*Seismic design value evaluation based on checking records and site geological conditions using artificial neural networks*" by T. Kerh et al., several improved computational neural network models are proposed to evaluate seismic design values based on checking records and site geological conditions.

Finally, other applied problems are also considered. For example, in "*Crude oil price prediction based on a dynamic correcting support vector regression machine*" by L. Shu-rong and G. Yu-lei, a new accurate method of predicting crude oil prices is presented, which is based on an ε-support vector regression (ε-SVR) machine with a dynamic correction factor overcoming forecasting errors. The authors also propose a hybrid RNA genetic algorithm (HRGA) with the position displacement idea of bare bones particle swarm optimization (PSO) changing the mutation operator. In "*Mathematical model based on BP neural network algorithm for the deflection identification of storage tank and calibration of tank capacity chart*" by C. Li et al., the proposed method has better performance in terms of tank capacity chart calibration accuracy compared with other existing approaches and has a strong practical significance. In "*A study on coastline extraction and its trend based on remote sensing image data mining*" by Y. Zhang et al., data mining theory is applied to the pretreatment of remote sensing images. In "*Land use patch generalization based on semantic priority*" by J. Yang et al., the authors establish a neighborhood analysis model and patch features and simplify the narrow zones and the feature sidelines. In "*Nonstationary INAR(1) process with qth-order autocorrelation innovation*" by K. Yu et al., an integer-valued random walk process with qth-order autocorrelation is discussed.

Acknowledgments

The guest editors of this special issue would like to express their thanks to the authors who have submitted papers for consideration and the referees of the submitted papers.

Jianhong (Cecilia) Xia
Fuding Xie
Yong Zhang
Craig Caulfield

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 259863, 11 pages
http://dx.doi.org/10.1155/2013/259863

Research Article

Dictionary Learning Based on Nonnegative Matrix Factorization Using Parallel Coordinate Descent

Zunyi Tang,[1] Shuxue Ding,[2] Zhenni Li,[1] and Linlin Jiang[3]

[1] *Graduate School of Computer Science and Engineering, University of Aizu, Aizu-Wakamatsu City, Fukushima 965-8580, Japan*
[2] *School of Computer Science and Engineering, University of Aizu, Aizu-Wakamatsu City, Fukushima 965-8580, Japan*
[3] *Department for Student Affairs, University of Aizu, Aizu-Wakamatsu City, Fukushima 965-8580, Japan*

Correspondence should be addressed to Zunyi Tang; tangzunyi@gmail.com

Received 28 February 2013; Accepted 16 May 2013

Academic Editor: Yong Zhang

Sparse representation of signals via an overcomplete dictionary has recently received much attention as it has produced promising results in various applications. Since the nonnegativities of the signals and the dictionary are required in some applications, for example, multispectral data analysis, the conventional dictionary learning methods imposed simply with nonnegativity may become inapplicable. In this paper, we propose a novel method for learning a nonnegative, overcomplete dictionary for such a case. This is accomplished by posing the sparse representation of nonnegative signals as a problem of nonnegative matrix factorization (NMF) with a sparsity constraint. By employing the coordinate descent strategy for optimization and extending it to multivariable case for processing in parallel, we develop a so-called parallel coordinate descent dictionary learning (PCDDL) algorithm, which is structured by iteratively solving the two optimal problems, the learning process of the dictionary and the estimating process of the coefficients for constructing the signals. Numerical experiments demonstrate that the proposed algorithm performs better than the conventional nonnegative K-SVD (NN-KSVD) algorithm and several other algorithms for comparison. What is more, its computational consumption is remarkably lower than that of the compared algorithms.

1. Introduction

Dictionary learning, building a dictionary consisting of atoms or subspaces so that a class of signals can be efficiently and sparsely represented in terms of the atoms, is an important topic in machine learning, neuroscience, signal processing, and so forth. Since in some applications the nonnegativities of the signals and the dictionary are required, for example, multispectral data analysis [1, 2], nonnegative factorization for recognition [3, 4], and some other important problems [5, 6], the so-called nonnegative dictionary learning becomes necessary. In this paper, we mainly focus on this topic.

In the model of sparse representation of signals, a basic assumption is that using an overcomplete dictionary matrix $\mathbf{W} \in \mathbb{R}^{m\times r}$ that contains r atoms of size $m \times 1$ for columns, $\{\mathbf{w}_i\}_{i=1}^{r}$, each column vector of a signal matrix $\mathbf{Y} \in \mathbb{R}^{m\times n}$ can be represented as a linear combination of very few, which is meant by the terminology of sparse, atoms $\mathbf{w}_i$ of dictionary $\mathbf{W}$. Here, the term "overcomplete" means $m < r$. $\mathbf{Y} = \mathbf{WH}$ or $\mathbf{Y} \approx \mathbf{WH}$ satisfying $\|\mathbf{Y} - \mathbf{WH}\|_2 \leq \varepsilon$ are two ways to represent $\mathbf{Y}$. The corresponding matrix $\mathbf{H} \in \mathbb{R}^{r\times n}$ that contains the representation coefficients of signals $\mathbf{Y}$ is called the coefficient matrix. For dictionary $\mathbf{W}$, it can be either generated by a prespecified set of functions or learned by a given set of training signals. In practices [7, 8], learning a dictionary has proved to be critical to achieve superior results in the domains of signal and image processing.

Naturally, the problem of finding a dictionary and its sparse representation with the fewest number of atoms can be modeled by using the ℓ^0-norm. Considering the fact that the ℓ^0-norm optimization problem is generally NP-hard, one frequently used heuristic is the ℓ^1-minimization [9]. A series of studies has led to many dictionary learning algorithms. Several classical algorithms include LARS [10], K-SVD [11], ILS-DLA [12], ODL [13], and RLS-DLA [14].

Although these algorithms are very efficient in general, they are not always suitable for learning a nonnegative dictionary from nonnegative signals. For example, a nonnegative variant of K-SVD, which is termed "NN-KSVD" [15], is not as efficient as K-SVD because the negative elements generated in a dictionary matrix are intentionally set to zero to guarantee nonnegativity as the dictionary updates.

In recent years, nonnegative matrix factorization (NMF) [2, 16] has been widely applied to data analyses having nonnegativity constraints since NMF can factorize a nonnegative matrix into a product of two nonnegative factor matrices with different properties. Intuitively, NMF is similar to sparse representation of nonnegative signals to some extent. However, the standard NMF algorithms [17] do not impose any constraints on the two factors, except for nonnegativity, which is not sufficient to lead to a sparse enough representation. In order to obtain a sparser representation, various sparsity constrained NMF algorithms have been proposed. Hoyer et al. [18–20] considered enforcing the sparsity of coefficient matrix using ℓ^1-norm. Hoyer [21] also introduced a measure of sparsity based on the ratio of the ℓ^1-norm of a vector to the ℓ^2-norm. Some algorithms imposed sparsity constraints by using ℓ^2-norm [5, 22, 23]. Peharz et al. [24, 25] presented sparse NMF algorithms that constrain the ℓ^0-(pseudo-) norm of the coefficient matrix. In addition, several approaches based on other types of constraints, such as nonsmoothness constraint [26], squared ℓ^1-norm penalization [27], and mixed-norm [28], have been proposed recently.

Inspired by the sparsity constrained NMF, in this paper we present a new method for learning a nonnegative overcomplete dictionary for sparse representation of nonnegative signals. Differently from the optimization strategies used in the conventional sparsity constrained NMF, this method employs the coordinate descent strategy [29] and extends it to multivariable case for optimizing multiple independent variables in factors, thus resulting in the so-called parallel coordinate descent strategy. We present the update rules based on the new strategy and develop an algorithm, which is termed as the parallel coordinate descent dictionary learning (PCDDL) algorithm, to solve our objective problem. The proposed algorithm is very efficient since the objective problem has been cast as two sequential optimal problems of quadratic functions not involving the complicated calculations inherent to factorization. Through experimental evaluations, we have observed that the proposed algorithm achieves the best rate of atom recovery compared with the conventional algorithms [15, 18, 21, 25]. In addition, its performance is robust even if noise is quite heavy. Furthermore, the computation cost of our algorithm is much lower than that of other algorithms because it does not involve the complicated calculations.

The remaining part of the paper is organized as follows. In Section 2, we formulate the nonnegative dictionary learning problem. In Section 3, we describe the proposed PCDDL algorithm for nonnegative dictionary learning. In Section 4, we report the results of numerical experiments using PCDDL and compare these results with those of several other algorithms. These experiments involve two groups of synthetic datasets and two preliminary applications involving image processing. Finally, in Section 5, we draw our conclusions and discuss related research topics for the future.

2. Problem Formulation

Given a vector $\mathbf{y} \in \mathbb{R}^m$, whose components are a group of signals, we are now concerned with its sparse representation over an overcomplete dictionary $\mathbf{W} \in \mathbb{R}^{m\times r}$, each column of which is referred to an atom. That is, we attempt to find a linear combination of only few atoms, which can be close to $\mathbf{y}$ in value. To avoid trivial solutions, $\mathbf{W}$ is restricted to the set $\mathbb{C}$, which is defined as

$$\mathbb{C} \triangleq \left\{\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_n] : \mathbf{w}_j^T \mathbf{w}_j = 1,\ \forall j = 1, \ldots, n\right\}. \quad (1)$$

For a training set of n signals $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$, dictionary learning can be formulated as the following optimization problem:

$$\min_{\mathbf{W}\in\mathbb{C},\mathbf{H}} \frac{1}{n}\sum_{i=1}^{n} \mathscr{F}_i(\mathbf{h}_i, \mathbf{W}), \quad (2)$$

where $\mathbf{H} = \{\mathbf{h}_1, \ldots, \mathbf{h}_n\}$ and

$$\mathscr{F}_i(\mathbf{h}_i, \mathbf{W}) = \frac{1}{2}\|\mathbf{y}_i - \mathbf{W}\mathbf{h}_i\|_2^2 + P(\mathbf{h}_i, \lambda). \quad (3)$$

Here $P(\mathbf{h}_i, \lambda)$ is a penalty function with $\lambda > 0$, which is a tuning parameter controlling the tradeoff between the approximation error $(1/2)\|\mathbf{y}_i - \mathbf{W}\mathbf{h}_i\|_2^2$ and the penalty function $P(\mathbf{h}_i, \lambda)$.

Naturally, the problem of learning a dictionary $\mathbf{W}$ and finding a sparse representation $\mathbf{h}_i$ can be modeled by using the ℓ^0-norm, defining $P(\mathbf{h}_i, \lambda)$ as the ℓ^0-norm of $\mathbf{h}_i$; namely, $P(\mathbf{h}_i, \lambda) = \lambda\|\mathbf{h}_i\|_0$. However, the resulting optimization problem is usually NP-hard. Considering this difficulty, one frequently used heuristic is the ℓ^1-norm; that is, $P(\mathbf{h}_i, \lambda) = \lambda\|\mathbf{h}_i\|_1$ [9].

With the use of the ℓ_1-norm, the dictionary learning problem is expressed as follows:

$$\min_{\mathbf{W}\in\mathbb{C},\mathbf{H}} \frac{1}{n}\sum_{i=1}^{n} \left\{\frac{1}{2}\|\mathbf{y}_i - \mathbf{W}\mathbf{h}_i\|_2^2 + \lambda\|\mathbf{h}_i\|_1\right\}. \quad (4)$$

Noted that it is allowed to take different values of λ for different penalty functions $P(\mathbf{h}_i, \lambda)$. For the sake of simplicity, however, we assume here that the same λ is applied to every penalty function. Thus, (4) can be also rewritten as a matrix factorization problem with a sparsity penalty,

$$\min_{\mathbf{W}\in\mathbb{C},\mathbf{H}} \frac{1}{2}\|\mathbf{Y} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda\|\mathbf{H}\|_{1,1}, \quad (5)$$

where $\|\mathbf{H}\|_{1,1}$ denotes the ℓ_1-norm of the matrix $\mathbf{H}$, that is, the sum of the ℓ^1-norm of each column vector of the matrix $\mathbf{H}$.

Furthermore, if $\mathbf{Y}$ is nonnegative and factors $\mathbf{W}$ and $\mathbf{H}$ are both limited to be nonnegative, then the process is called nonnegative dictionary learning, which can be formulated as,

$$\min_{\mathbf{W}\in\mathbb{C},\mathbf{H}} \quad \frac{1}{2}\|\mathbf{Y} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda\|\mathbf{H}\|_{1,1} \quad \text{subject to} \quad \mathbf{W} \succeq 0,\ \mathbf{H} \succeq 0. \quad (6)$$

To solve the problem in (6), a natural strategy is to optimize between $\mathbf{W}$ and $\mathbf{H}$ alternatively. That is, minimize one while keeping the other fixed. The NN-KSVD algorithm [18] and some NMF algorithms including NNSC, NMFSC, and NMFℓ^0-H, just solve the problem in such a way.

3. The Proposed Method

3.1. Parallel Coordinate Descent Dictionary Learning (PCDDL). To solve the objective problem (6), we first employ alternating update strategy, that is, updating one of two factors while fixing the other. In the optimization of each factor, we propose optimizing each component in the factor one by one by generalizing the coordinate descent strategy [29], rather than optimizing the whole factor at a time as in the standard NMF algorithms [17]. Furthermore, we found that (6) can be separated into column-wise or row-wise subproblems, and each subproblem can just be solved alternately and explicitly by utilizing the properties of solving extreme value problem of a quadratic function, so that the whole problem can be solved efficiently.

We here derive the update rules for $\mathbf{H}$ and $\mathbf{W}$ of (6). In terms of the definition and properties of the Frobenius norm, for a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\|\mathbf{A}\|_F^2 = \mathrm{Tr}(\mathbf{A}\mathbf{A}^T) = \mathrm{Tr}(\mathbf{A}^T\mathbf{A})$. $\mathrm{Tr}(\cdot)$ denotes the trace of a square matrix. Thus, the objective function (6) can be decomposed as follows:

$$J = \frac{1}{2}\sum_{j=1}^{n}\mathbf{Y}_{j:}^T\mathbf{Y}_{:j} - \sum_{j=1}^{n}\left[\mathbf{Y}^T\mathbf{W}\right]_{j:}\mathbf{H}_{:j} + \frac{1}{2}\sum_{j=1}^{n}\mathbf{H}_{j:}^T\mathbf{W}^T\mathbf{W}\mathbf{H}_{:j} + \lambda\sum_{i=1}^{r}\sum_{j=1}^{n}\left|\mathbf{H}_{ij}\right|, \tag{7}$$

where $[\mathbf{Y}^T\mathbf{W}]_{j:}$ denotes the jth row of the multiplication of matrices $\mathbf{Y}^T$ and $\mathbf{W}$. Since the elements of $\mathbf{H}$ have nonnegativity, the absolute value operation in (7) can be omitted. If we fix $\mathbf{W}$ in (7), then (7) is a multivariable objective function of $\mathbf{H}_{ij}$ ($i = 1,\ldots,r$; $j = 1,\ldots,n$). First, let us explain the coordinate descent strategy for a single variable. For (7), we consider optimizing only a single variable $\mathbf{H}_{kj}$, while fixing the other components in $\mathbf{H}$. Thus, we obtain a quadratic function with regard to $\mathbf{H}_{kj}$ as follows:

$$J_{\mathbf{H}_{kj}} = \frac{1}{2}\left[\mathbf{W}^T\mathbf{W}\right]_{kk}\mathbf{H}_{kj}^2 + \mathbf{H}_{kj}\left(\sum_{l=1,l\neq k}^{r}\left[\mathbf{W}^T\mathbf{W}\right]_{kl}\mathbf{H}_{lj} - \left[\mathbf{W}^T\mathbf{Y}\right]_{kj} + \lambda\right), \tag{8}$$

where $[\mathbf{W}^T\mathbf{W}]_{kk}$ denotes the entry in the kth row and the kth column of the multiplication of matrices $\mathbf{W}^T$ and $\mathbf{W}$. $[\mathbf{W}^T\mathbf{W}]_{kk}$ is always positive because it is a diagonal element of Gram matrix $\mathbf{W}^T\mathbf{W}$ (no zero vectors exist in $\mathbf{W}$ here, also). Thus, when $\mathbf{H}_{kj} = ([\mathbf{W}^T\mathbf{Y}]_{kj} - \sum_{l=1,l\neq k}^{r}[\mathbf{W}^T\mathbf{W}]_{kl}\mathbf{H}_{lj} - \lambda)/[\mathbf{W}^T\mathbf{W}]_{kk}$, $J_{\mathbf{H}_{kj}}$ reaches the minimum. Considering the nonnegativity of factor $\mathbf{H}$, $\mathbf{H}_{kj}$ is set to 0 when it is negative. Note that, when updating $\mathbf{H}_{kj}$, the process involves only the elements $\mathbf{H}_{lj,l\neq k}$ of the jth column in $\mathbf{H}$. That is, the optimal value for a given entry of $\mathbf{H}$ does not depend on the other components of the same row containing the entry. Hence, one can optimize all elements of one row in $\mathbf{H}$ at the same time. This can be viewed as optimizing the elements in parallel, that is, parallel coordinate descent strategy for multiple variables. Thus, the update rule for $\mathbf{H}$ of (7) is given as follows:

$$\begin{aligned}\mathbf{H}_{k:}^* &= \arg\min_{\mathbf{H}_{k:}\geq 0}\|\mathbf{Y} - \mathbf{W}\mathbf{H}\|_F^2 \\ &= \max\left(0, \frac{\mathbf{W}_{k:}^T\mathbf{Y} - \sum_{l=1,l\neq k}^{r}\mathbf{W}_{k:}^T\mathbf{W}_{:l}\mathbf{H}_{l:} - \lambda}{\mathbf{W}_{k:}^T\mathbf{W}_{:k}}\right) \\ &= \max\left(0, \frac{\mathbf{W}_{k:}^T\mathbf{R}_k - \lambda}{\|\mathbf{W}_{:k}\|_2^2}\right),\end{aligned} \tag{9}$$

where $\mathbf{R}_k = \mathbf{Y} - \sum_{l=1,l\neq k}^{r}\mathbf{W}_{:l}\mathbf{H}_{l:}$.

Similar to the derivation of the update rule for $\mathbf{H}$, one can also obtain the corresponding update rule for $\mathbf{W}$. If fixing $\mathbf{H}$ in (7), then (7) is a multivariable objective function of $\mathbf{W}_{ij}$ ($i = 1,\ldots,m$; $j = 1,\ldots,r$). For (7), we now consider optimizing only one variable $\mathbf{W}_{ik}$, while fixing the other components in $\mathbf{W}$. We first select the items related to $\mathbf{W}_{ik}$ from (7) and obtain a quadratic function with regard to $\mathbf{W}_{ik}$ as follows:

$$J_{\mathbf{W}_{ik}} = \frac{1}{2}\left[\mathbf{H}\mathbf{H}^T\right]_{kk}\mathbf{W}_{ik}^2 + \mathbf{W}_{ik}\left(\sum_{l=1,l\neq k}^{r}\mathbf{W}_{il}\left[\mathbf{H}\mathbf{H}^T\right]_{lk} - \left[\mathbf{Y}\mathbf{H}^T\right]_{ik}\right). \tag{10}$$

One can find that (10) is very similar to (8). In terms of the properties of a single variable quadratic problem, $J_{\mathbf{W}_{ik}}$ obtains the minimum when $\mathbf{W}_{ik} = ([\mathbf{Y}\mathbf{H}^T]_{ik} - \sum_{l=1,l\neq k}^{r}\mathbf{W}_{il}[\mathbf{H}\mathbf{H}^T]_{lk})/[\mathbf{H}\mathbf{H}^T]_{kk}$. Considering the nonnegativity of factor $\mathbf{W}$, $\mathbf{W}_{ik}$ is set to 0 when it is negative. Similar to the update rule for $\mathbf{H}$, $\mathbf{W}$ in (7) can update by column. Thus, the update rule for $\mathbf{W}$ of (7) is expressed as follows:

$$\begin{aligned}\mathbf{W}_{:k}^* &= \operatorname*{argmin}_{\mathbf{W}_{:k}\geq 0}\|\mathbf{Y} - \mathbf{W}\mathbf{H}\|_F^2 \\ &= \max\left(0, \frac{\mathbf{Y}\mathbf{H}_{:k}^T - \sum_{l=1,l\neq k}^{r}\mathbf{W}_{:l}\mathbf{H}_{l:}\mathbf{H}_{:k}^T}{\mathbf{H}_{k:}\mathbf{H}_{:k}^T}\right) \\ &= \max\left(0, \frac{\mathbf{R}_k\mathbf{H}_{:k}^T}{\|\mathbf{H}_{k:}\|_2^2}\right).\end{aligned} \tag{11}$$

In addition, for preventing dictionary $\mathbf{W}$ from having arbitrarily large values, each column of $\mathbf{W}$ is normalized to the unit ℓ^2-norm when dictionary $\mathbf{W}$ is updating. Note that the way of maintaining the nonnegativity of two factor matrices in PCDDL is obviously different from that of NN-KSVD. The former can guarantee that the obtained nonnegative solutions are the optimal relative to each column-wise or row-wise updating, but the latter cannot.

Require: Data Matrix $\mathbf{Y} \in \mathbb{R}_+^{m\times n}$, initial matrices $\mathbf{W} \in \mathbb{R}_+^{m\times r}$, $\mathbf{H} \in \mathbb{R}_+^{r\times n}$, and λ;
(1) **while** stopping criterion not satisfied **do**
(2) Computing $\mathbf{P} = \mathbf{Y}\mathbf{H}^T$ and $\mathbf{Q} = \mathbf{H}\mathbf{H}^T$;
(3) **for** $k = 1$ to r **do**
(4) $\mathbf{W}_{:k} \leftarrow \max\left(0, \dfrac{\mathbf{P}_{:k} - \sum_{l=1,l\neq k}^{r} \mathbf{W}_{:l}\mathbf{Q}_{lk}}{\mathbf{Q}_{kk}}\right)$
(5) Normalizing $\mathbf{W}_{:k} \leftarrow \dfrac{\mathbf{W}_{:k}}{\|\mathbf{W}_{:k}\|_2}$
(6) **end for**
(7) Computing $\mathbf{U} = \mathbf{W}^T\mathbf{Y}$ and $\mathbf{V} = \mathbf{W}^T\mathbf{W}$;
(8) **for** $k = 1$ to r **do**
(9) $\mathbf{H}_{k:} \leftarrow \max\left(0, \dfrac{\mathbf{U}_{k:} - \sum_{l=1,l\neq k}^{r} \mathbf{V}_{kl}\mathbf{H}_{l:} - \lambda}{\mathbf{V}_{kk}}\right)$
(10) **end for**
(11) Using the fixed λ or adaptively tuning λ according to the change of the sparsity of $\mathbf{H}$;
(12) **end while**

Algorithm 1: PCDDL.

Remark 1. According to the above derivation, it can be observed that our objective function (7) can be cast as two sequential optimal problems of quadratic functions, each of which can be alternately optimized in parallel by the generalized coordinate descent strategy.

Remark 2. The sparsity of $\mathbf{H}$ can be flexibly controlled by tuning the regularization parameter λ.

Remark 3. The method is suitable not only for the case of overdetermined dictionary matrices ($m > r$) but also for the case of underdetermined dictionary matrices ($m < r$), even though these matrices have different physical meanings in different applications.

3.2. Choice of Parameter λ and Summary of Algorithm. In the step of updating $\mathbf{H}$ with a fixed $\mathbf{W}$, the parameter $\lambda > 0$ can be adjusted for controlling the tradeoff between the approximation error $(1/2)\|\mathbf{Y} - \mathbf{W}\mathbf{H}\|_F^2$ and the sparsity of coefficient matrix $\mathbf{H}$ and plays an important role in the proposed algorithm. To steer the solution toward a global, optimal solution, the parameter λ can be determined by two kinds of ways, off-line calibrating and adaptive tuning.

For the first way, one can repeat an experiment with different λ and determine what value for λ is the optimal according to the output results.

For the second way, we give an easy-to-use rule as follows. First, λ should be less than $\|\mathbf{W}_{k:}^T\mathbf{R}_k\|_\infty$ in terms of (9); otherwise $\mathbf{H}_{k:}$ will become a zero vector. We may initialize λ with a very small value, for example, 0.001, which can generally satisfy the above condition. Next, we alternately update $\mathbf{H}$ and $\mathbf{W}$ in terms of (9) and (11) and adjust λ according to the rule defined as follows:

$$\lambda^{(k+1)} = \begin{cases} \lambda^{(k)} + 0.001 & \text{if } S\left(\mathbf{H}^{(k-1)}\right) - S\left(\mathbf{H}^{(k)}\right) < 10^{-3}, \\ & S\left(\mathbf{H}^{(k)}\right) - S^* > 10^{-3} \\ \lambda^{(k)} & \text{otherwise,} \end{cases} \tag{12}$$

where $S(\mathbf{H})$ is a sparsity measure, defined as $\|\mathbf{H}\|_0/(r \times n)$, which calculates the ratio of the number of nonzero elements and the number of all elements in $\mathbf{H}$. $\lambda^{(k)}$ and $S(\mathbf{H}^{(k)})$ denote the value of λ and the sparsity of $\mathbf{H}$ in the kth iteration, respectively. S^* denotes the expected or *a prior* sparsity of $\mathbf{H}$. The rule means that if the sparsity of $\mathbf{H}$ varies very slowly and is far from the expected one, one may appropriately increase the stepsize of λ; otherwise, keep the current λ. Experiments show that the values of λ obtained by the two ways are very close. If λ is self-tuned for adapting to signal, however, more iterations are usually needed for convergence.

According to the analysis above, the proposed PCDDL algorithm for nonnegative dictionary learning is summarized in Algorithm 1.

3.3. Convergence Analysis of PCDDL Algorithm. The standard NMF algorithms [17] belong to two-block convex optimization scheme since each factor can be viewed as a block, and optimizing one of two factors while fixing the other is separately convex. Grippo and Sciandrone analyzed the convergence of the two-block convex optimization problems in [30]. They demonstrated that under the condition of continuously differentiable objective function, a two-block convex optimization algorithm does not require each subproblem to have a unique solution for convergence, and any limit point of the sequence of optimal solutions of two-block subproblems is a stationary point. Obviously, PCDDL is such a two-block convex optimization algorithm, so that we can make analysis of its convergence by using the facts in [30]. During iterations, PCDDL can obtain a sequence of the limit points that can guarantee the reduction of objective function. Additionally, in terms of the definition of ℓ^1-norm, the penalty term $\|\mathbf{H}\|_{1,1}$ in (6) can be decomposed into $\sum_{i=1}^{r}\sum_{j=1}^{n}\mathbf{H}_{ij}$ since $\mathbf{H} \geq 0$. Thus, under the conditions of $\lambda > 0$, the objective function (6) is differentiable with respect to $\mathbf{W}$ and $\mathbf{H}$, respectively. The existence of limit points and the differentiability of the objective function in (6) imply that the assumptions of Grippo and Sciandrone's Corollary

[30] are satisfied, so that we can establish that the two-block minimization processes of PCDDL converge.

4. Numerical Experiments

In this section, first we present the results of two experiments using PCDDL with synthetic signals. The aims of these experiments are (1) to test whether the PCDDL algorithm can recover the true dictionary, which is used to generate the test data; and (2) to compare the results with those of other algorithms, such as NNSC (online available: http://www.cs.helsinki.fi/u/phoyer/) [18], NN-KSVD (online available: http://www.cs.technion.ac.il/~elad/) [15], NMFSC (online available: http://www.cs.helsinki.fi/u/phoyer/) [21], and NMFℓ^0-H (online available: http://www.spsc.tugraz.at/tools/nmf-l0-sparseness-constraints) [25]. Next, we apply PCDDL to a conventional digital image processing problem, image denoising, to verify the applicability of the proposed algorithm in a real-world environment. Finally, we carry out an experiment of learning a global-based representation on a face dataset in order to demonstrate the practicality of the proposed algorithm for further large-scale data analysis. In the experiments, all programs were coded in Matlab and were run within Matlab 7.8 (R2009a) on a PC with a 3.2 GHz Intel Core i5 CPU and 4 G of memory.

4.1. Recovery Experiment of Random Dictionary. To evaluate the learning capacity of the proposed algorithm for a nonnegative dictionary, we conducted an experiment of recovering a random dictionary from synthetic observation signals generated from the random dictionary. By comparing the recovery rate of the dictionary, adaptability, runtime, and so forth, we assess the algorithms under consideration (see above). The processes are as follows. We generated a stochastic nonnegative matrix of size 20×50 with i.i.d. uniformly distributed entries, as described in [11]. Each vector was normalized to unit ℓ^2-norm. The stochastic nonnegative matrix was referred to as the true dictionary $\mathbf{W}$, which was not used in the learning but was used only for evaluation. We then synthesized 1500 test signals $\mathbf{Y}$ of dimension 20, each of which was produced by a linear combination of three different atoms in the true dictionary, with three corresponding coefficients in random and independent positions. We executed NNSC, NMFSC, NN-KSVD, NMFℓ^0-H, and PCDDL on the test signals. For the five algorithms, the initialized dictionary matrices of size 20×50 were composed of the randomly selected parts of the test signals. For NNSC, NMFSC, and PCDDL, the corresponding coefficient matrices were initialized with i.i.d. uniformly distributed random nonnegative entries. NN-KSVD and NMFℓ^0-H do not require a specified coefficient matrix, as they can generate the corresponding coefficient matrix by sparse coding.

Next, we compared the learned dictionaries with the true dictionary. These comparisons were done by sweeping through the columns of the true and the learned dictionaries and finding the closest column (in ℓ^2-norm distance) between the two dictionaries. A distance of less than 0.01 was considered a success. The experiment is similar to the one conducted in [11], except for the nonnegative condition. Obviously, the five iterative algorithms described above have different convergence properties. To provide fair limits on the number of the respective iterations, we executed these algorithms with the same iterations as many times as possible and determined respective iteration number in terms of the results shown in Figures 1, 2, and 3. NNSC and NMFSC, respectively, took about 3000 iterations to reach convergence, while NMFℓ^0-H took only dozens of iterations. In addition, we also considered the runtime of each algorithm as showed in Figure 2. Thus, we set the maximum numbers of iterations for NNSC, NMFSC, NN-KSVD, NMFℓ^0-H, and PCDDL to 3000, 3000, 300, 30, and 500, respectively. Certainly, the iteration of any algorithm can be terminated in advance if it has learned 100% of the atoms before reaching the maximum number of iterations.

Besides the noiseless condition, we also made experiments in which the uniformly distributed positive noise of varying signal-to-noise ratios (SNRs) was corrupted to the test signals in order to evaluate the performance and robustness of antinoise. All trials were repeated 15 times with different initialized dictionaries. Figure 4 shows the results of the experiment for noise levels of 10, 20, and 30 dB and for the noiseless case. Obviously, NMFSC and NN-KSVD performed worst, especially under lower SNR conditions. NMFℓ^0-H performed better than NNSC, NMFSC, and NN-KSVD under various conditions. The proposed PCDDL performed best on dictionary learning, although it performed only slightly better than NMFℓ^0-H under various conditions. The average runtime of each trial for these algorithms was 35 s, 146 s, 244 s, 24 s, and 4 s, respectively. Obviously, PCDDL has a remarkable advantage in computational consumption. Note that, in the experiment, NN-KSVD and NMFℓ^0-H required a specified, exact number of nonzero elements in the coefficient matrix ($3/50 = 0.06$ for the case) as shown in Figure 3, and NMFSC was executed with a sparsity factor of 0.8 on the coefficients. For NNSC and PCDDL, the sparsity of the coefficient matrices was adjusted via the regularization parameters λ. In the experiment, the corresponding parameters λ were set to 0.2 in both the cases, which was calibrated off-line through several trials. The two parameters λ were fixed during iterations in order to reduce the number of iterations and computational cost.

4.2. Recovery Experiment of Decimal Digits Dictionary. To further investigate the potential practicality of the proposed PCDDL algorithm, we considered the 10 decimal digits dataset from [15]. The dataset is composed of 90 images of size 8×8, representing 10 decimal digits with various position shifts. Note that a mistake exists in the original dataset, in which some atoms are duplicated. In the original dataset, for example, the atoms of the first column are the same as the ones of the fifth column. Before the experiment, we corrected the problem by making all the atoms different.

Before beginning the experiment, we first generated 3000 training signals of size 64×1, each of which is a random linear combination of 5 different atoms with random positive coefficients. That is, there are uniformly 5 nonzero elements

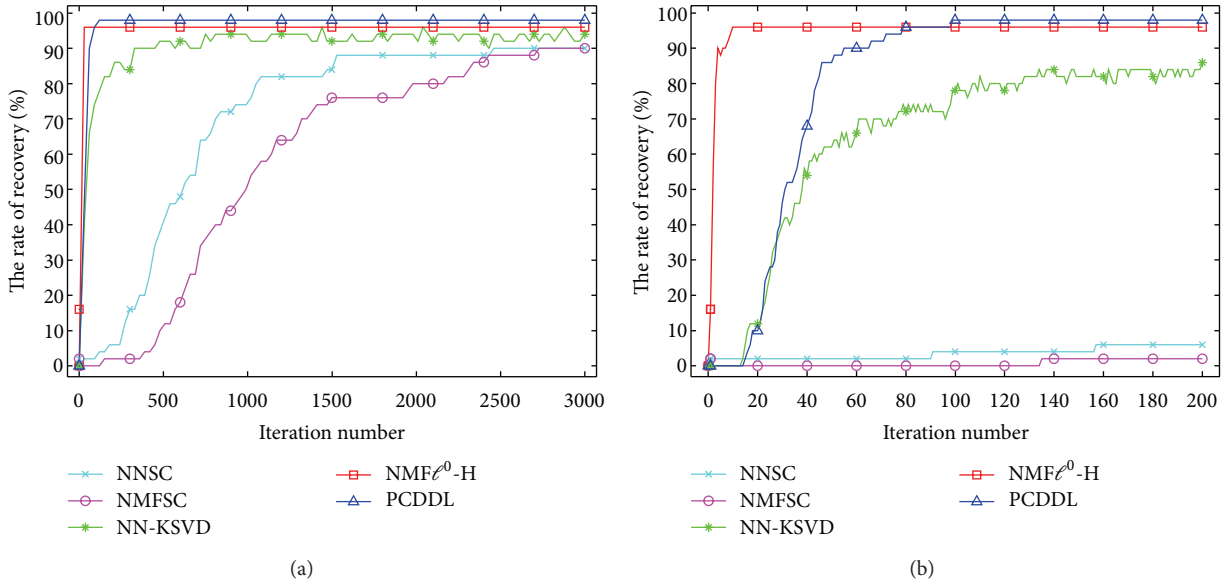


Figure 1: Evolution of the rate of atom recovery versus the iteration number of five algorithms. (a) It shows 3000 iterations. (b) It is a close-up view of the former 200 iterations for (a).

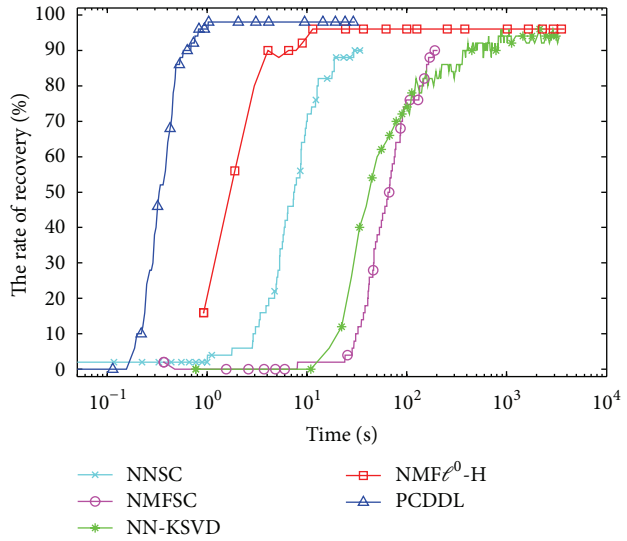


Figure 2: Evolution of the rate of atom recovery versus the runtime of five algorithms. These algorithms run 3000 iterations, respectively. PCDDL achieved the best rate of recovery in the least time.

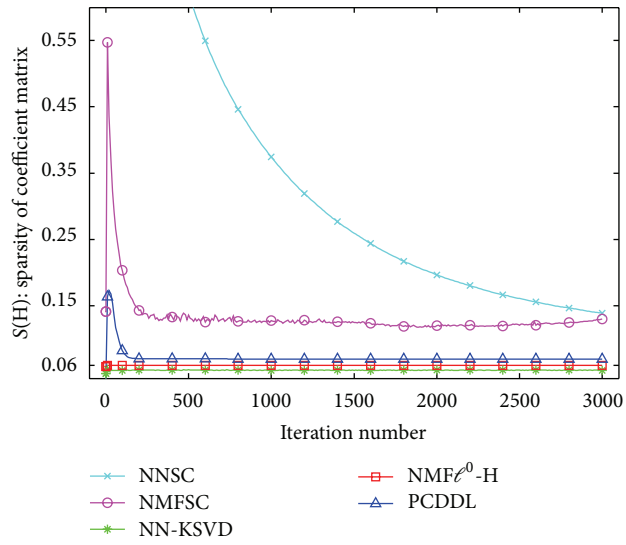


Figure 3: Evolution of the sparsity of the coefficient matrix versus the iteration number of five algorithms.

in each vector of the corresponding coefficient matrix. In order to learn original dictionary, the training signals were input into the five algorithms, NNSC, NMFSC, NN-KSVD, NMFℓ^0-H, and PCDDL. We also added the uniformly distributed positive noise of varying SNR to the training signals in order to evaluate the robustness of antinoise. The obtained results are shown in Figure 5.

As the results of the experiments in the above subsection, PCDDL performed better than the other four algorithms at three noise levels and in the noiseless case. The results of NN-KSVD were not as good as described in [15], because we corrected the above-mentioned mistake in the original dataset (i.e., removed duplicated atoms). The duplicated atoms in the original dataset led to the better, but wrong, result in [15] compared with the results of our experiment. Surprisingly, NNSC performed worst in this experiment, and it could almost not learn any correct atoms no matter how the parameters had been chosen. In a typical run, the average runtime of each trial was 412 s, 473 s, 822 s, 136 s and 23 s, respectively. This fact further shows that PCDDL has a remarkable advantage in computational consumption. In Figure 6, we give an example of the experiment under noiseless conditions, in which the four algorithms except

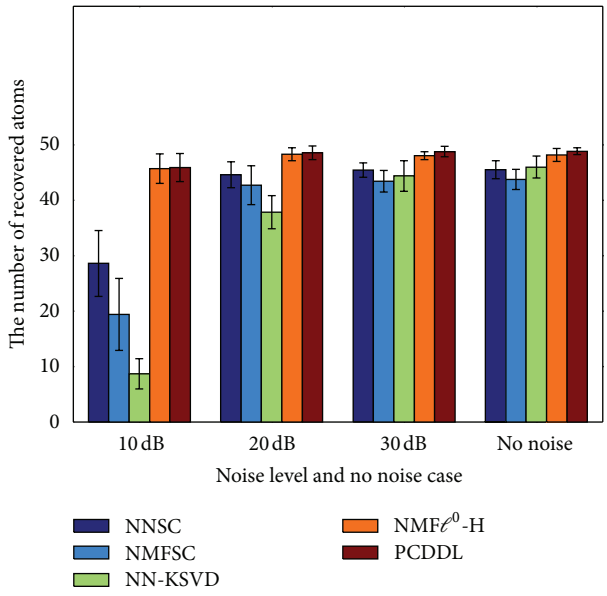


Figure 4: Results of a synthetic experiment with a dictionary of size 20 × 50. For each of the tested algorithms and for each noise level, 15 trials were performed. Averaged values of learned atoms and corresponding deviation values are displayed.

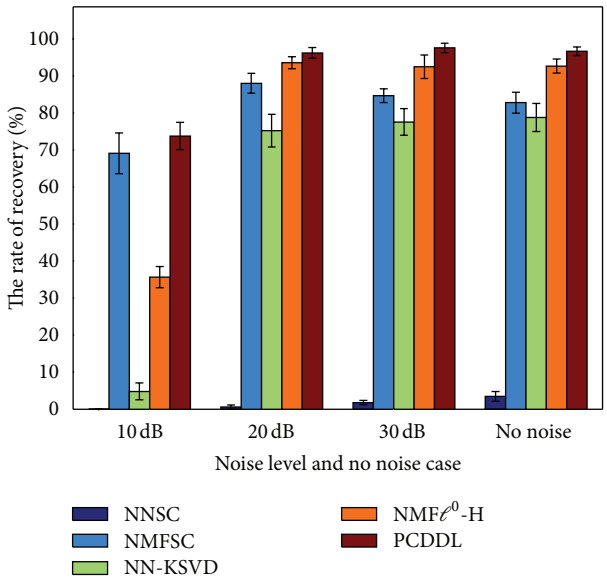


Figure 5: Results of a synthetic experiment with a decimal digits dictionary of size 64 × 90. For each of the tested algorithms and for each noise level, 10 trials were performed. Averaged values of learned atoms and corresponding deviation values are displayed.

NNSC recovered 77, 72, 86, and 89 atoms of 90 atoms, respectively. The result for NNSC was not showed in Figure 6 since it could almost not learn any correct atoms. Figure 6(a) shows the dataset revised by us. Figure 6(f) shows the result obtained by PCDDL, where only one digit 8 could not be recovered correctly. Certainly, PCDDL can either recover 100% of the atoms in considerable cases.

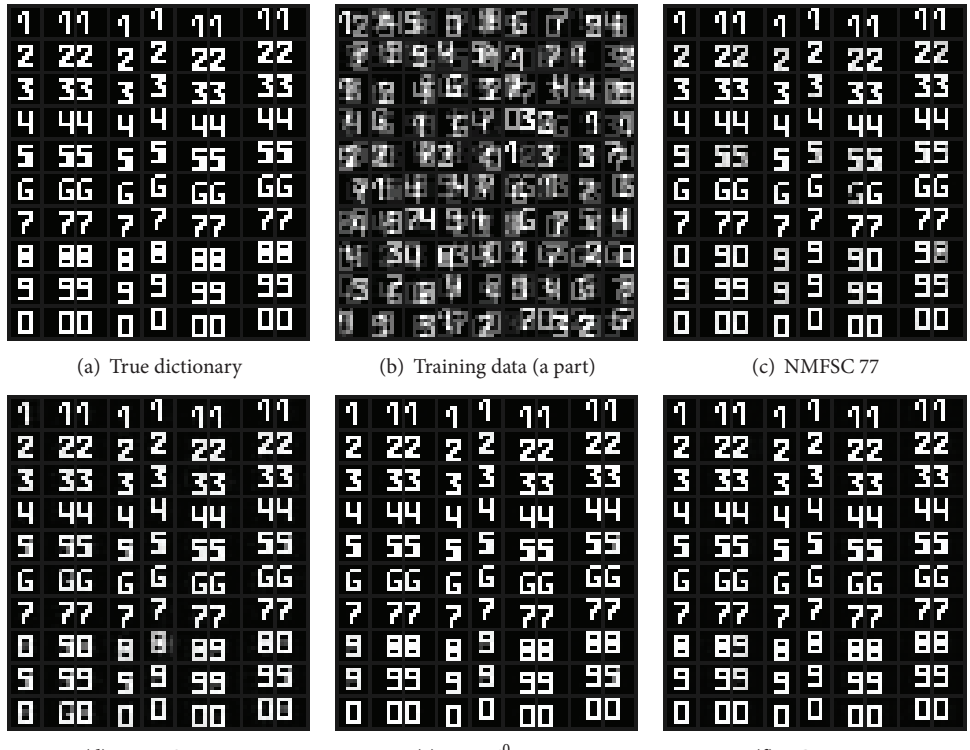
4.3. Image Denoising of Nature Images. Image denoising problem is important, not only because of the obvious applications that it serves. Being the simplest possible inverse problem, it provides a convenient platform through which image processing ideas and techniques can be assessed. In this sense, we intend to apply nonnegative dictionary learning to image denoising problem. Using redundant representations and sparsity as driving forces for denoising of signals constitutes significant progress [31, 32]. In these studies, a typical noise model is $\mathbf{Y} = \mathbf{X} + \mathbf{V}$, where $\mathbf{X} \in \mathbb{R}^{m\times n}$ is the clean image, $\mathbf{V} \in \mathbb{R}^{m\times n}$ is assumed to be white Gaussian noise with a fixed standard deviation σ (the case of nonuniform σ is dealt with in [33]), and $\mathbf{Y} \in \mathbb{R}^{m\times n}$ is the noisy observed image. Here, the noise is assumed to be uniformly distributed with nonnegative values, instead of zero-mean white and homogeneous Gaussian noise, since this paper is for studying the sparse representation of nonnegative signals. For solving the denoising problem, we adopted the algorithm presented in [31], which is based on a sparse and redundant representation model on small image patches. In the procedure, the original dictionary learning algorithm is replaced with our proposed PCDDL.

In this set of experiments, the dictionaries used were of size 64 × 256, which were designed to handle image patches of size 8 × 8 pixels. All reported results are presented as an average of three experiments, having different realizations of the noise. Some standard test images including Barbara (512 × 512), House (256 × 256), Boats (512 × 512), Lena (512 × 512), and Peppers (256 × 256) were used in the experiment. We added noise of various levels to the test images. We used two quality measures, the peak SNR (PSNR) and the structural similarity (SSIM), to assess the denoised images. Let $\mathbf{X}$ and $\widehat{\mathbf{X}}$ denote the ideal image and the deteriorated image, respectively. We calculate the PSNR value of $\widehat{\mathbf{X}}$ by $\mathrm{PSNR}(\widehat{\mathbf{X}}) = 10 \cdot \log_{10}(1/(\mathbf{X} - \widehat{\mathbf{X}})^2)$. For SSIM, its value range is between 0 and 1, and its value equals 1 if $\mathbf{X} = \widehat{\mathbf{X}}$. For more information about the SSIM index, please refer to references in [34].

In the experiment, we focused on tests with higher noise levels, because it may be more critical. We chose the conventional Wavelets denoising algorithm [35] and the known nonlocal means (NL-means) algorithm [36] as the compared objects. Additionally, we also chose the NMFℓ^0-H because of its better performance in previous experiments. It is notable that NMFℓ^0-H is very time-consuming for the dictionary learning procedure, as described in the two experiments above. Table 1 summarizes the results of the denoising experiment. We concluded that the denoising algorithm using the PCDDL dictionary achieved highly competitive PSNR and SSIM performance outcomes compared to that of Wavelets, NL-means, and NMFℓ^0-H algorithms. When comparing PSNR, the denoising algorithm using the PCDDL dictionary outperformed NL-means in the range of about 0.7 dB~2 dB and performed much better than the Wavelets and NMFℓ^0-H algorithms. When comparing the SSIM index, the denoising algorithm using the PCDDL dictionary returned results comparable to that of the NL-means algorithm. Subjective quality comparisons for two

Table 1: PSNR (dB) and SSIM results for different algorithms. In each cell, four groups of denoising results are shown. Top row, Wavelets; second row, NL-means; third row, NMFℓ^0-H; bottom row, PCDDL.

Input PSNR	Lena		Barbara		Boat		House		Pepper		Average	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
29.97	32.91	0.8775	29.61	0.8633	30.28	0.8093	33.05	0.8669	31.20	0.8881	31.41	0.8610
	37.76	0.9370	36.62	0.9583	35.55	0.9205	38.17	0.9357	36.02	0.9485	36.82	0.9400
	29.23	0.9224	33.15	0.9538	35.95	**0.9451**	33.68	0.9519	35.76	0.9568	33.55	0.9460
	39.39	**0.9491**	**38.69**	**0.9641**	**38.08**	0.9445	**40.10**	**0.9580**	**38.63**	**0.9577**	**38.98**	**0.9547**
20.12	28.52	0.7982	25.03	0.7168	26.63	0.6988	28.03	0.7856	26.24	0.7902	26.89	0.7579
	33.45	0.8756	31.74	0.8914	31.04	0.8148	34.06	0.8729	32.06	**0.8929**	32.47	0.8695
	29.61	0.8680	30.82	**0.9056**	29.17	0.8232	33.93	**0.8846**	28.22	0.8788	30.35	**0.8720**
	34.39	**0.8791**	**32.58**	0.8870	**32.24**	**0.8376**	**34.32**	0.8714	**32.70**	0.8839	**33.25**	0.8718
14.09	25.74	0.7312	22.71	0.6044	24.23	0.6101	24.90	0.7220	23.15	0.7021	24.15	0.6740
	30.14	**0.8039**	27.52	**0.7867**	27.69	**0.7145**	30.44	0.8087	28.52	**0.8235**	28.86	**0.7875**
	27.68	0.7870	24.63	0.7411	24.10	0.6578	27.64	**0.8170**	22.87	0.7604	25.38	0.7527
	31.24	0.7953	**28.71**	0.7700	**28.80**	0.7137	**31.21**	0.7981	**29.27**	0.7933	**29.85**	0.7741
8.82	23.47	0.6712	21.07	0.5249	22.32	0.5384	22.51	0.6666	20.56	0.6164	21.99	0.6035
	27.18	**0.6876**	24.32	**0.6402**	24.99	**0.5938**	26.60	0.6767	24.97	**0.7114**	25.61	**0.6619**
	21.58	0.6677	19.60	0.5175	20.38	0.5190	21.41	0.6746	19.31	0.6232	20.46	0.6004
	28.38	0.6727	**25.26**	0.5986	**26.08**	0.5693	**28.37**	**0.6917**	**26.27**	0.6643	**26.87**	0.6393



(a) True dictionary (b) Training data (a part) (c) NMFSC 77

(d) NN-KSVD 72 (e) NMFℓ^0-H 86 (f) PCDDL 89

Figure 6: (a) True dictionary composed of 90 atoms. (b) Part of the total training data. (c)–(f) Learned dictionaries from NMFSC, NN-KSVD, NMFℓ^0-H, and PCDDL algorithms. The numbers of learned atoms are 77, 72, 86, and 89, respectively. Note that these resulting dictionaries have been realigned to facilitate comparison with the original dictionary.

typical test images (Boat and House) are shown in Figures 7 and 8. The PCDDL dictionary learned from the noisy House image in Figure 8 is illustrated in Figure 9.

4.4. Human Face Image Analysis. In this subsection, we describe our experiment on learning a global-based representation [21] using a face dataset. The learning process can be considered to be one kind of principal component analysis. We used the ORL dataset of faces (online available: http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html). Since the ORL dataset includes 400 facial images of size 92 × 112 pixels, the dataset can be considered to be large scale. Using the dataset, we can evaluate the computational performances of the PCDDL and the other

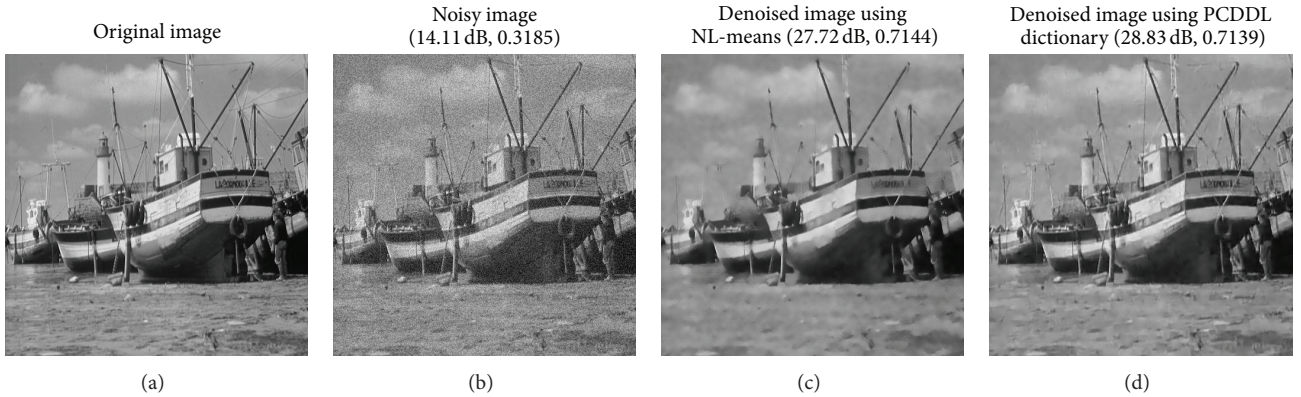


Figure 7: Example of denoising results for the image "Boat" with a noise level of 14.11 dB. In brackets, the former items denote PSNR values, and the latter items denote the SSIM index.

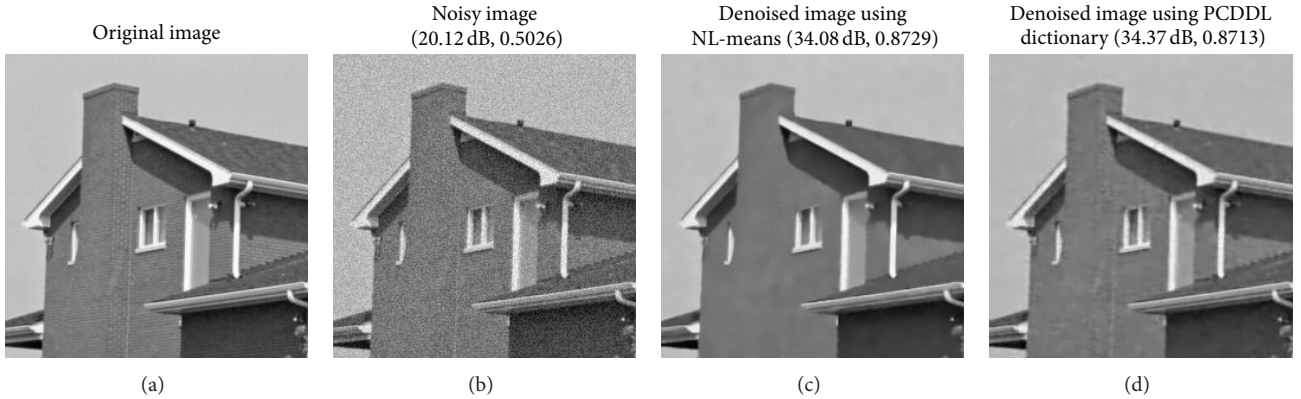


Figure 8: Example of the denoising results for the image "House" with the noise level of 20.12 dB. In brackets, the former items denote PSNR values, and the latter items denote the SSIM index.

compared algorithms. To assess the experiment fairly, we drove the compared algorithms to obtain the corresponding coefficient matrices and forced them to reach as comparable level of sparsity as possible (based on ℓ^0-norm). By using the Hoyer's sparsity measure for a vector $\mathbf{x} \in \mathbb{R}^n$, defined as

$$\text{Sparsity}(\mathbf{x}) = \frac{\sqrt{n} - \|\mathbf{x}\|_1 / \|\mathbf{x}\|_2}{\sqrt{n} - 1} \in [0, 1], \tag{13}$$

we compared the average sparsity of all column vectors in these coefficient matrices. Additionally, we computed the respective relative errors defined below and counted the respective runtime

$$\text{Relative Error} = \frac{\|\mathbf{Y} - \mathbf{WH}\|_F}{\|\mathbf{Y}\|_F}. \tag{14}$$

In the experiment, we performed a global-based feature learning of rank $r = 36$ and constrained the coefficient matrices to have a sparsity of about 0.08; that is, each facial image was required to be represented with three facial features ($36 \times 0.08 \approx 3$). Besides NMFSC and NMFℓ^0-H, we chose another sparse NMF algorithm (denoted as SNMF) [20] as the compared objective. Note that NN-KSVD was not included in this experiment, since it has exceedingly high computational consumption. Each of these algorithms required some initialization parameters and a limit on the number of its iterations. For SNMF, we allowed 3000 iterations; and for the parameter α, which is used to adjust sparsity, we chose 100. For NMFSC, we only constrained the

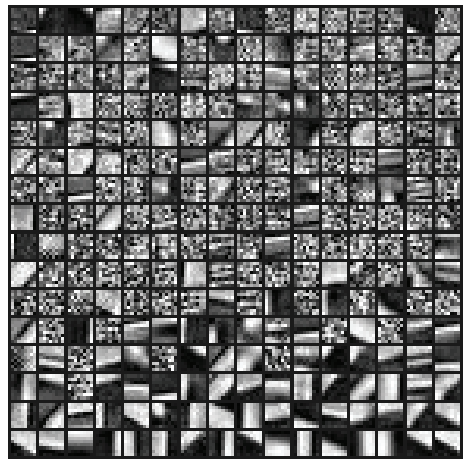
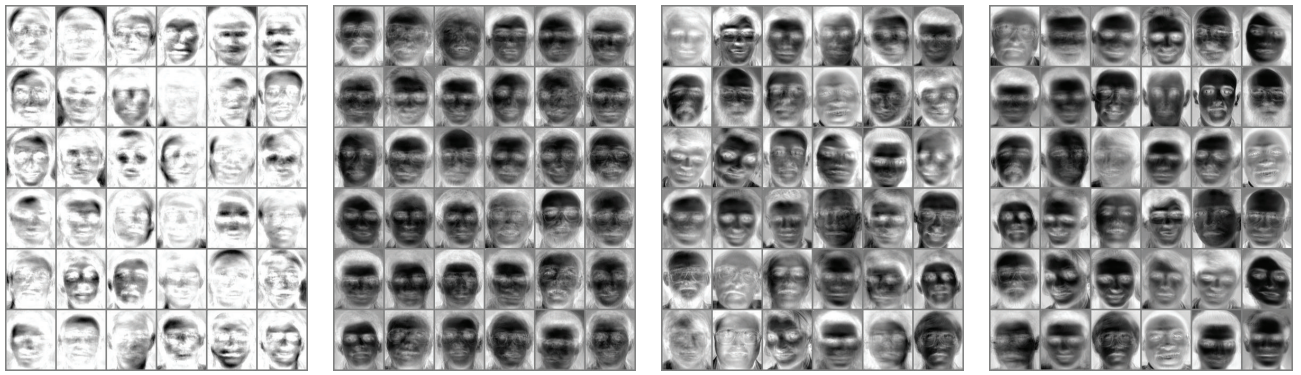


Figure 9: The PCDDL dictionary has a size of 64×256, which was learned from the noisy House image in Figure 8.



(a) SNMF (b) NMFSC (c) NMFℓ^0-H (d) PCDDL

Figure 10: Globally featured faces learned by SNMF, NMFSC, NMFℓ^0-H, and PCDDL.

sparsity of coefficient factor $\mathbf{H}$ to 0.9 in terms of (13) and executed at most 3000 iterations, which was necessary for convergence. For NMFℓ^0-H, we set the maximum number of nonzero elements of vectors in factor $\mathbf{H}$ to 3 ($3/36 \approx 0.0833$, close to 0.08) and allowed 30 iterations, considering the high computational consumption of NMFℓ^0-H. For the proposed PCDDL, we allowed at most 200 iterations, and λ was set to 10, that is, calibrated through several trials. All four algorithms were run three times with the same initial random matrices (for NMFℓ^0-H, it was not necessary to initialize coefficient $\mathbf{H}$). The averaged results are reported in Table 2.

Through Table 2, it can be observed that SNMF seems to be incapable of obtaining an actual sparse representation, despite the fact that it is designed to enhance sparsity by introducing the ℓ^1-norm. The other three algorithms obtained similar results and produced much sparser solutions, that is, more global-based representations. NMFSC and NMFℓ^0-H produced lower relative errors but took much more runtime than PCDDL. The runtime of NMFSC and NMFℓ^0-H was about 14 and 23 times longer than that of PCDDL. In view of its high efficiency, PCDDL is more suitable for large-scale data analysis. In Figure 10, we show an illustration of the global-based features learned by the four algorithms in a typical run.

Table 2: Comparisons of $S(\mathbf{H})$-based sparsity, Hoyer's sparsity (based on (13)), relative error (based on (14)), and runtime for SNMF, NMFSC, NMFℓ^0-H, and PCDDL.

Algorithm	$S(\mathbf{H})$	Sparsity	Relative error	Time (s)
SNMF	96.65	0.4314	0.9904	940
NMFSC	8.00	0.9490	0.2520	415
NMFℓ^0-H	8.33	0.8957	0.1852	662
PCDDL	8.00	0.9447	0.2925	28

5. Conclusion

In this paper, we presented a novel and efficient method for learning nonnegative dictionaries for sparse representation of nonnegative signals. In this method, we generalized the coordinate descent strategy for optimization for being able to be applied to a multivariable case, so that it can process in a parallel way. By this strategy we developed an efficient algorithm, which has been named as the parallel coordinate descent dictionary learning (i.e., PCDDL) algorithm. The algorithm updates the dictionary in a column-wise manner and the coefficient matrix in a row-wise manner. In each column-wise or row-wise updating, PCDDL optimizes a series of optimal problems sequentially, each of which is an optimization of a quadratic function. Furthermore, such optimization problems can be solved explicitly, so that the algorithm can be processed very precisely and quickly from a global perspective according to the properties of the univariate quadratic problem. For this reason, the proposed algorithm can efficiently solve the nonnegative dictionary learning problem with very high accuracy.

Results of experiments on dictionary recovery showed that PCDDL can correctly learn a nonnegative, overcomplete dictionary, regardless of wether the objective signals are synthetic data or are natural images. Additionally, further experiments supported the potential application of PCDDL in the field of image processing, such as image denoising, image classification, and large-scale data processing due to its low computational consumption. We are currently working on applying this method to some practical problems in image processing, for example, large-scale image classification. The results from these ongoing studies will be presented in the future.

References

[1] V. P. Pauca, J. Piper, and R. J. Plemmons, "Nonnegative matrix factorization for spectral data analysis," *Linear Algebra and its Applications*, vol. 416, no. 1, pp. 29–47, 2006.

[2] L. Miao and H. Qi, "Endmember extraction from highly mixed data using minimum volume constrained nonnegative matrix factorization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, pp. 765–777, 2007.

[3] S. Li, X. Hou, H. Zhang, and Q. Cheng, "Learning spatially localized, parts-based representation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, pp. 207–212, 2001.

[4] I. Kotsia, S. Zafeiriou, and I. Pitas, "A novel discriminant non-negative matrix factorization algorithm with applications to facial image characterization problems," *IEEE Transactions on Information Forensics and Security*, vol. 2, pp. 588–595, 2007.

[5] F. Shahnaz, M. W. Berry, V. P. Pauca, and R. J. Plemmons, "Document clustering using nonnegative matrix factorization," *Information Processing & Management*, vol. 42, pp. 373–386, 2006.

[6] M. Wang, W. Xu, and A. Tang, "A unique "nonnegative" solution to an underdetermined system: from vectors to matrices," *IEEE Transactions on Signal Processing*, vol. 59, no. 3, pp. 1007–1016, 2011.

[7] M. Plumbley, T. Blumensath, L. Daudet, R. Gribonval, and M. Davies, "Sparse representations in audio and music: from coding to source separation," *Proceedings of the IEEE*, pp. 995–1005, 2010.

[8] M. Elad, M. Figueiredo, and Y. Ma, "On the role of sparse and redundant representations in image processing," *Proceedings of the IEEE*, vol. 98, pp. 972–982, 2010.

[9] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ^1 minimization," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 100, no. 5, pp. 2197–2202, 2003.

[10] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[11] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transaction on Signal Processing*, vol. 54, pp. 4311–4322, 2006.

[12] K. Engan, K. Skretting, and J. H. Husoy, "Family of iterative LS-based dictionary learning algorithms, ILS-DLA, for sparse signal representation," *Digital Signal Processing*, vol. 17, pp. 32–49, 2007.

[13] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *Journal of Machine Learning Research*, vol. 11, pp. 19–60, 2010.

[14] K. Skretting and K. Engan, "Recursive least squares dictionary learning algorithm," *IEEE Transactions on Signal Processing*, vol. 58, no. 4, pp. 2121–2130, 2010.

[15] M. Aharon, M. Elad, and A. M. Bruckstein, "K-SVD and its non-negative variant for dictionary design," in *Proceedings of the SPIE Conference Wavelets*, pp. 327–339.

[16] D. D. Lee and H. S. Seung, "Learning the parts of objects by nonnegative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[17] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems*, pp. 556–562, 2000.

[18] P. O. Hoyer, "Non-negative sparse coding," in *Proceedings of the IEEE Workshop on Neural Networks for Signal Processing*, pp. 557–565.

[19] J. Eggert and E. Korner, "Sparse coding and NMF," in *Proceedings of IEEE International Joint Conference on Neural Networks*, pp. 2529–2533.

[20] W. Liu, N. Zheng, and X. Lu, "Non-negative matrix factorization for visual coding," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03)*, pp. 293–296, 2003.

[21] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research*, vol. 5, pp. 1457–1469, 2004.

[22] V. P. Pauca, F. Shahnaz, M. W. Berry, and R. J. Plemmons, "Text mining using non-negative matrix factorizations," in *Proceedings of the Fourth SIAM International Conference on Data Mining*, pp. 452–456, SIAM, Philadelphia, Pa, USA, 2004.

[23] Y. Gao and G. Church, "Improving molecular cancer class discovery through sparse non-negative matrix factorization," *Bioinformatics*, vol. 21, pp. 3970–3975, 2005.

[24] R. Peharz, M. Stark, and F. Pernkopf, "Sparse nonnegative matrix factorization using ℓ^0-constraints," in *IEEE International Workshop on Machine Learning for Signal Processing (MLSP '10)*, pp. 83–88, 2010.

[25] R. Peharz and F. Pernkopf, "Sparse nonnegative matrix factorization using ℓ^0-constraints," *Neurocomputing*, vol. 80, pp. 38–46, 2012.

[26] A. Pascual-Montano, J. M. Carazo, K. Kochi, D. Lehmann, and R. D. Pascual-Marqui, "Nonsmooth nonnegative matrix factorization (nsNMF)," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 403–415, 2006.

[27] H. Kim and H. Park, "Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis," *Bioinformatics*, vol. 23, pp. 1495–1502, 2007.

[28] R. Tandon and S. Sra, "Sparse nonnegative matrix approximation: new formulations and algorithms," Tech. Rep. 193, MPI, 2010.

[29] J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani, "Pathwise coordinate optimization," *The Annals of Applied Statistics*, vol. 1, no. 2, pp. 302–332, 2007.

[30] L. Grippo and M. Sciandrone, "On the convergence of the block nonlinear Gauss-Seidel method under convex constraints," *Operations Research Letters*, vol. 26, no. 3, pp. 127–136, 2000.

[31] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006.

[32] W. Dong, X. Li, L. Zhang, and G. Shi, "Sparsity-based image denoising via dictionary learning and structural clustering," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 457–464, 2011.

[33] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Transactions on Image Processing*, vol. 17, no. 1, pp. 53–69, 2008.

[34] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, pp. 600–612, 2004.

[35] R. Baraniuk, H. Choi, R. Neelamani, and V. Ribeiro, "Rice Wavelet Toolbox," 2011, http://dsp.rice.edu/software/rice-wavelet-toolbox/.

[36] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, pp. 60–65, 2005.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 268639, 15 pages
http://dx.doi.org/10.1155/2013/268639

Research Article

A Real-Valued Negative Selection Algorithm Based on Grid for Anomaly Detection

Ruirui Zhang, Tao Li, and Xin Xiao

College of Computer Science, Sichuan University, Chengdu 610065, China

Correspondence should be addressed to Ruirui Zhang; zhangruiruisw@gmail.com

Received 15 March 2013; Accepted 13 May 2013

Academic Editor: Fuding Xie

Negative selection algorithm is one of the main algorithms of artificial immune systems. However, candidate detectors randomly generated by traditional negative selection algorithms need to conduct self-tolerance with all selves in the training set in order to eliminate the immunological reaction. The matching process is the main time cost, which results in low generation efficiencies of detectors and application limitations of immune algorithms. A novel algorithm is proposed, named GB-RNSA. The algorithm analyzes distributions of the self set in real space and regards the n-dimensional $[0, 1]$ space as the biggest grid. Then the biggest grid is divided into a finite number of sub grids, and selves are filled in the corresponding subgrids at the meantime. The randomly generated candidate detector only needs to match selves who are in the grid where the detector is and in its neighbor grids, instead of all selves, which reduces the time cost of distance calculations. And before adding the candidate detector into mature detector set, certain methods are adopted to reduce duplication coverage between detectors, which achieves fewer detectors covering the nonself space as much as possible. Theory analysis and experimental results demonstrate that GB-RNSA lowers the number of detectors, time complexity, and false alarm rate.

1. Introduction

In the past decade, the artificial immune systems have caused great concerns as a new method to solve complex computational problems. At present, there are four main areas in the studies of artificial immune systems [1]: the negative selection algorithm (NSA) [2], the artificial immune network (AINE) [3], the clonal selection algorithm (CLONALG) [4], the danger theory [5], and dendritic cell algorithms [6]. By simulating the immune tolerance in T-cell maturation process of biological systems, NSA removes self-reactive candidate detectors to effectively recognize nonself antigens, and is successfully applied to pattern recognition, anomaly detection, machine learning, fault diagnosis, and so forth [7, 8].

The negative selection algorithm is proposed by Forrest et al. [7]. This algorithm adopts strings or binary strings to encode the antigens (samples) and antibodies (detectors) and r-continuous-bit matching method to compute affinities between antigens and detectors, which is denoted SNSA [7]. The work in [9, 10] pointed out that the generation efficiency of detectors in SNSA is low. Candidate detectors become mature through negative selection. Given that N_s is the training set size, P' is the matching probability between random antigen and antibody, and P_f is the failure rate; then the number of candidate detectors $N = -\ln(P_f)/(P'(1 - P')^{N_s})$, which is exponential to N_s, and the time complexity of SNSA, is $O(N \cdot N_s)$.

Because many problems in practical applications are easy to be defined and studied in the real space, a real-valued negative selection algorithm (RNSA) is put forward in [11]. The algorithm adopts n-dimensional vectors in real space $[0, 1]^n$ to encode antigens and antibodies and Minkowski distance to calculate affinities. A real-valued negative selection algorithm with variable-sized detector (V-Detector) is proposed in [12, 13], resulting in better results. The algorithm dynamically determines the radius of a detector to generate mature ones, by computing the nearest distance between the center of the candidate detector and self-antigens. This algorithm also proposes a method for calculating detectors' coverage rate based on the probability. In the work of [14],

genetic-based negative selection algorithm is put forward, and in the work of [15], clonal optimization-based negative selection algorithm is put forward. Detectors of these two algorithms need to be processed by optimization algorithms, to gain greater coverage of nonself space. Superellipsoid detectors are introduced in [16] in the negative selection algorithm and superrectangular detectors in [17], to achieve the same coverage rate with less detectors compared with sphere ones. A self detector classification method is proposed in [18]. In this method, selves are viewed as self detectors with initial radius and the radius of selves is dynamically determined by the ROC analysis in the training stage, to increase the detection rate. A negative selection algorithm based on the hierarchical clustering of self set is put forward in [19]. This algorithm carries out the hierarchical clustering preprocess of self set to improve the generation efficiency of detectors.

Because of the low generation efficiency of mature detectors, the time cost of negative selection algorithms seriously limits their practical applications [18, 19]. A real-valued negative selection algorithm based on grid is proposed in this paper, denoted GB-RNSA. The algorithm analyzes distributions of the self set in the shape space and introduces the grid mechanism, in order to reduce the time cost of distance calculations and the duplication coverage between detectors. The remainder of this paper is organized as follows. The basic definitions of real-valued negative selection algorithms which are also the background of this paper are described in Section 2. The basic idea, implementation strategies, and analyses of GB-RNSA are described in Section 3. The effectiveness of GB-RNSA is verified using synthetic datasets and University of California Irvine (UCI) datasets in Section 4. Finally, the conclusion is given in the last section.

2. Basic Definitions of RNSA

The SNS (self/nonself) theory states that the body relies on antibodies (T cells and B cells) to recognize self antigens and nonself antigens, in order to exclude foreigners and maintain the balance and stability of the body [2, 8]. Inspired by this theory, antibodies are defined as detectors to identify nonself antigens in the artificial immune system, and their quality determines the accuracy and effectiveness of the detection system. However, randomly generated candidate detectors may identify self antigens and raise the immune self-reaction. According to the immune tolerance mechanism and mature process of immune cells in the biological immune system, Forrest put forward the negative selection algorithm to remove detectors which can recognize selves [7]. The algorithm discussed in this paper is based on real value. The basic concepts of RNSA are as follows.

Definition 1 (antigens). $Ag = \{ag \mid ag = \langle x_1, x_2, \ldots, x_n, r_s\rangle, x_i \in [0,1],\ 1 \le i \le n,\ r_s \in [0,1]\}$ are the total samples in the space of the problem. ag is an antigen in the set. n is the data dimension, x_i is the normalized value of the ith attribute of sample ag which represents the position in the real space, and r_s is the radius of ag which represents the variability threshold of ag.

Definition 2 (self set). $Self \subset Ag$ represents all the normal samples in the antigen set.

Definition 3 (nonself set). $Nonself \subset Ag$ represents all the abnormal samples in the antigen set. *Self*/*Nonself* have different meanings in various fields. For network intrusion detections, *Nonself* represents network attacks, and *Self* represents normal network access; for virus detections, *Nonself* represents virus codes, and *Self* represents legitimate codes.

$$Self \cap Nonself = \emptyset, \qquad Self \cup Nonself = Ag. \tag{1}$$

Definition 4 (training set). $Train \subset Self$ is a subset of *Self* and is the priori detection knowledge. N_s is the size of the training set.

Definition 5 (set of detectors). $D = \{d \mid d = \langle y_1, y_2, \ldots, y_n, r_d\rangle,\ y_j \in [0,1],\ 1 \le j \le n,\ r_d \in [0,1]\}$. d is a detector in the set. y_j is the jth attribute of detector d, r_d is the radius of the detector, and N_d is the size of the detector set.

Definition 6 (matching rule). $A(ag, d) = dis(ag, d)$, and $dis(ag, d)$ is the Euclidean distance between antigen ag and detector d. In the detectors' generation process, if $dis(ag, d) \le r_s + r_d$, the detector d arises the immune self-reaction and cannot become a mature detector. In the detectors' testing process, if $dis(ag, d) < r_d$, the detector d recognizes the antigen ag as a nonself.

Definition 7 (detection rate). *DR* means the proportion of non-self samples which are correctly identified by detectors in the total non-self samples and is represented by (2). *TP* is short for true positive, which means the number of nonselves which are correctly identified by detectors. *FN* is short for false negative, which means the number of non-selves which are wrongly identified:

$$DR = \frac{TP}{TP + FN}. \tag{2}$$

Definition 8 (false alarm rate). *FAR* means the proportion of self samples which are wrongly identified as non-selves in the total self samples and is represented by (3). *FP* is short for false positive, which means the number of selves which are wrongly identified by detectors, and *TN* is short for true negative, which means the number of selves which are correctly identified:

$$FAR = \frac{FP}{FP + TN}. \tag{3}$$

In general, the generation process of detectors which is the basic idea of RNSA is shown in Algorithm 1.

In the algorithm of RNSA, the randomly generated candidate detectors need to do the calculation $dis(d_{\text{new}}, ag)$ with all the elements in the training set. With the increase of the number of selves N_s, the execution time is in exponential growth, while the probability of coverage overlaps between detectors also raises, resulting in a large number of invalid detectors and low efficiency. The aforementioned problems greatly limit the practical applications of the negative selection algorithms.

RNSA($Train, r_d, maxNum, D$)
Input: the self training set $Train$, the radius of detectors r_d, the number of needed detectors $maxNum$
Output: the detector set D
Step 1. Initialize the self training set $Train$;
Step 2. Randomly generate a candidate detector d_{new}. Calculate the Euclidean distance between d_{new} and all the selves in $Train$. If $dis(d_{\text{new}}, ag) < r_d + r_s$ for at least one self antigen ag, execute Step 2; if not, execute Step 3.
Step 3. Add d_{new} into the detector set D;
Step 4. If the size of D satisfies $N_d > maxNum$, return D, and the process ends; if not, jump to Step 2.

Algorithm 1: The algorithm of RNSA.

3. Implementations of GB-RNSA

This section describes the implementation strategies of the proposed algorithm. The basic idea of the algorithm is described in Section 3.1. Sections 3.2, 3.3 and 3.4 are the detailed descriptions of the algorithm. The grid generation method is introduced in Section 3.2. Coverage calculation method of the non-self space is introduced in Section 3.3. And the filter method of candidate detectors is introduced in Section 3.4. Performance analysis of the algorithm is given in Section 3.5. Time complexity analysis of the algorithm is given in Section 3.6.

3.1. Basic Idea of the Algorithm. A real-valued negative selection algorithm based on grid GB-RNSA is proposed in this paper. The algorithm adopts variable-sized detectors and expected coverage of non-self space for detectors as the termination condition for detectors' generation. The algorithm analyzes distributions of the self set in the real space and regards $[0, 1]^n$ space as the biggest grid. Then, through divisions step-by-step until reaching the minimum diameter of the grid and adopting 2^n-tree to store grids, a finite number of subgrids are obtained, meanwhile self antigens are filled in corresponding sub grids. The randomly generated candidate detector only needs to match with selves who are in the grid where the detector is and in its neighbor grids instead of all selves, which reduces the time cost of distance calculations. When adding it into the mature detector set, the candidate detector will be matched with detectors within the grid where the detector is and neighbor grids, to judge whether the detector is in existing detectors' coverage area or its covered space totally contains other detector. This filter operation decreases the redundant coverage between detectors and achieves that fewer detectors cover the non-self space as much as possible. The main idea of GB-RNSA is as shown in Algorithm 2.

Iris dataset is one of the classic machine learning data sets published by the University of California Irvine [20], which are widely used in the fields of pattern recognition, data mining, anomaly detection, and so forth. We choose data records of category "setosa" in the dataset Iris as self antigens, choose "sepalL" and "sepalW" as antigen properties of first dimension and second dimension, and choose top 25 records of self antigens as the training set. Here, we use only two features of records, for that two-dimensional map is intuitive to illustrate the ideas, which does not affect comparison results. Figure 1 illustrates the ideas of GB-RNSA and the classical negative selection algorithms RNSA and V-Detector. RNSA generates detectors with fixed radius. V-Detector generates variable-sized detectors by dynamically determining the radius of detectors, through computing the nearest distance between the center of the candidate detector and self antigens. Detectors generated by the two algorithms need to conduct tolerance with all self antigens, which will lead to redundant coverage of non-self space between mature detectors with the increase of coverage rate. GB-RNSA first analyzes distributions of the self set in the space, and forms grids. Then, the randomly generated candidate detector only needs to perform tolerance with selves within the grid where the detector is and neighbor grids. Certain strategies are conducted for detectors which have passed tolerance, to avoid the duplication coverage and make sure that new detectors cover uncovered non-self space.

3.2. Grid Generation Method. In the process of grid generation, a top-down method is selected. First, the algorithm regards the n-dimensional $[0, 1]$ space as the biggest grid. If there are selves in this grid, divide each dimension into two parts and get 2^n sub grids. Then, continue to judge and divide each sub grid, until a grid does not contain any selves or the diameter of the grid reaches the minimum. Eventually, the grid structure of the space is obtained, and then the algorithm searches each grid to get neighbors in the structure. This process is shown in Algorithms 3 and 4.

Definition 9 (minimum diameter of grids). $r_{gs} = 4r_s + 4r_{ds}$, where r_s is the self radius and r_{ds} is the smallest radius of detectors. Suppose that the diameter of a grid is less than r_{gs}, then divide this grid; the diameter of sub grids is less than $2r_s + 2r_{ds}$. If there are selves in the sub grid, it is probably impossible to generate detectors in the sub grid. So, set the minimum diameter of grids $4r_s + 4r_{ds}$.

Definition 10 (neighbor grids). If two grids are adjacent at least in one dimension, these two grids are neighbors, which are called the basic neighbor grids. If selves of the neighbor grid are empty, add the basic neighbor grid of it in the same direction as the attached neighbor grid. The neighbors of a grid include the basic neighbor grids and the attached ones.

The filling process of neighbor grids is shown in Algorithm 5.

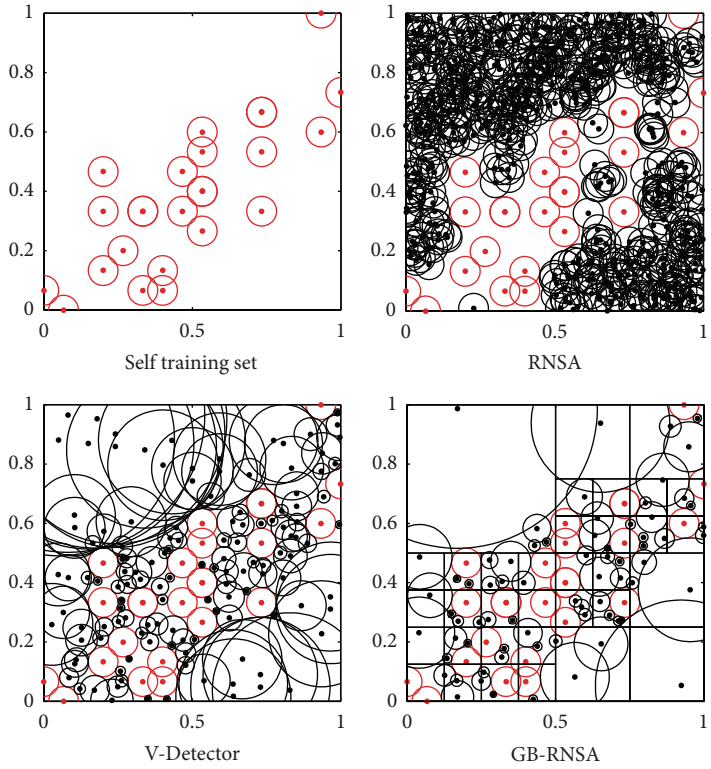


Figure 1: Comparison of RNSA, V-Detector, and GB-RNSA. (To reach the expected coverage $C_{exp} = 90\%$, three algorithms resp., need 561, 129, and 71 mature detectors, where the radius of self is 0.05, the radius of detector for RNSA is 0.05, and the smallest radius of detectors for V-Detector and GB-RNSA is 0.01).

GB-RNSA($Train, C_{exp}, D$)
Input: the self training set $Train$, expected coverage C_{exp}
Output: the detector set D
N_0: sampling times in non-self space, $N_0 > \max(5/C_{exp}, 5/(1 - C_{exp}))$
i: the number of non-self samples
x: the number of non-self samples covered by detectors
CD: the set of candidate detectors $CD = \{d \mid d =< y_1, y_2, \ldots, y_n, r_d >, y_j \in [0, 1], 1 \le j \le n, r_d \in [0, 1]\}$
Step 1. Initialize the self training set $Train$, $i = 0$, $x = 0$, $CD = \emptyset$, $N_0 = ceiling(\max(5/C_{exp}, 5/(1 - C_{exp})))$
Step 2. Call *GenerateGrid*($Train, TreeGrid, LineGrids$) to generate grid structure which contains selves, where $TreeGrid$ is the 2^n-tree storage of grids and $LineGrids$ is the line storage of grids;
Step 3. Randomly generate a candidate detector d_{new}. Call *FindGrid*($d_{new}, TreeGrid, TempGrid$) to find the grid $TempGrid$ where d_{new} is;
Step 4. Calculate the Euclidean distance between d_{new} and all the selves in $TempGrid$ and its neighbor grids. If d_{new} is identified by a self antigen, abandon it and execute Step 3; if not, increase i;
Step 5. Calculate the Euclidean distance between d_{new} and all the detectors in $TempGrid$ and its neighbor grids. If d_{new} is not identified by any detector, add it into the candidate detector set CD; if not, increase x, and judge whether it reaches the expected coverage C_{exp}, if so, return D and the algorithm ends;
Step 6. Judge whether i reaches sampling times N_0. If $i = N_0$, call *Filter*(CD) to implement the screening process of candidate detectors, and put candidate detectors which passed this process into D, reset i, x, CD; if not, return to Step 3.

Algorithm 2: The algorithm of GB-RNSA.

GenerateGrid(*Train*, *TreeGrid*, *LineGrids*)
Input: the self training set *Train*
Output: *TreeGrid* is the 2^n-tree storage of grids, *LineGrids* is the line storage of grids
Step 1. Generate the grid of *TreeGrid* with diameter 1, and set properties of the gird, including lower sub grids, neighbor grids, contained selves, and contained detectors;
Step 2. Call *DivideGrid*(*TreeGrid*, *LineGrids*)to divide grids;
Step 3. Call *FillNeighbours*(*LineGrids*) to find neighbors of each grid.

Algorithm 3: The process of grid generation.

DivideGrid(*grid*, *LineGrids*)
Input: *grid* the grid to divide
Output: *LineGrids* the line storage of grids
Step 1. If there are not any self or the diameter reaches r_{gs} of *grid*, don't divide, add *grid* into *LineGrids*, and return; if not, execute Step 2;
Step 2. Divide each dimension of *grid* into two parts, then get 2^n sub grids, and map selves of *grid* into the sub grids;
Step 3. For each sub grid, call *DivideGrid*(*grid.sub*, *LineGrids*).

Algorithm 4: The process of *DivideGrid*.

FillNeighbours(*LineGrids*)
Input: *LineGrids* the line storage of grids
Step 1. Obtain the basic neighbor grids for each grid in the structure *LineGrids*;
Step 2. For each basic neighbor of every grid, if selves of this neighbor are empty, complement the neighbor of this neighbor in the same direction as an attached neighbor for the grid;
Step 3. For each attached neighbor of every grid, if selves of this neighbor are empty, complement the neighbor of this neighbor in the same direction as an attached neighbor for the grid.

Algorithm 5: The filling process of neighbor grids.

Figure 2 describes the dividing process of grids. The self training set is also selected from records of category "setosa" of the Iris data set. Select "sepalL" and "sepalW" as antigen properties of first dimension and second dimension. As shown in Figure 2, the two-dimensional space is divided into four sub grids in the first division, and then continue to divide sub grids whose selves are not empty, until the subs cannot be divided.

Figure 3 is a schematic drawing of neighbor grids, and grids with slashes are the neighbors of grid $[0, 0.5, 0.5, 1]$ which positions in the up-left of the space.

3.3. Coverage Calculation Method of the Nonself Space. The non-self space coverage P is equal to the ratio of the volume V_{covered} covered by detectors and the total volume V_{nonself} of nonself space [12], as is shown in the following:

$$P = \frac{V_{\text{covered}}}{V_{\text{nonself}}} = \frac{\int_{\text{covered}} dx}{\int_{\text{nonself}} dx}. \tag{4}$$

Because there is redundant coverage between detectors, it is impossible to calculate (4) directly. In this paper, the probability estimation method is adopted to compute the detector coverage P. For detector set D, the probability of sampling in the non-self space covered by detectors obeys the binomial distribution $b(1, P)$ [13]. The probability of sampling m times obeys the binomial distribution $b(m, P)$.

Theorem 11. *When the number of non-self specimens of continuous sampling* $i \leq N_0$, *if* $(x/\sqrt{N_0P(1-P)}) - \sqrt{N_0P/(1-P)} > Z_\alpha$, *the non-self space coverage of detectors reaches* P. Z_α *is a percentile point of standard normal distribution,* x *is the number of non-self specimens of continuous sampling covered by detectors, and* N_0 *is the smallest positive integer which is greater than* $5/P$ *and* $5/(1-P)$.

Proof. Random variable $x \sim B(i, P)$. Set $z = x - N_0P/\sqrt{N_0P(1-P)} = (x/\sqrt{N_0P(1-P)}) - \sqrt{N_0P/(1-P)}$. We consider two cases.

(1) If the number of non-self specimens of continuous sampling $i = N_0$, known from De Moivre-Laplace theorem, when $N_0 > 5/P$ and $N_0 > 5/(1-P)$, $x \sim AN(N_0P, N_0P(1-P))$. That is, $x - N_0P/\sqrt{N_0P(1-P)} \sim AN(0, 1)$, $z \sim AN(0, 1)$. Do assumptions that H_0: the non-self space coverage of detectors $\leq P$; H_1: the non-self space coverage of detectors $> P$. Given significance level a,

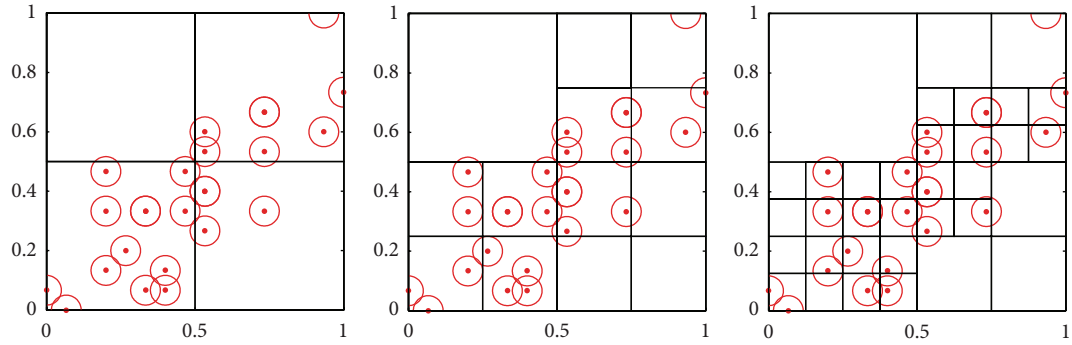


Figure 2: The process of grid division.

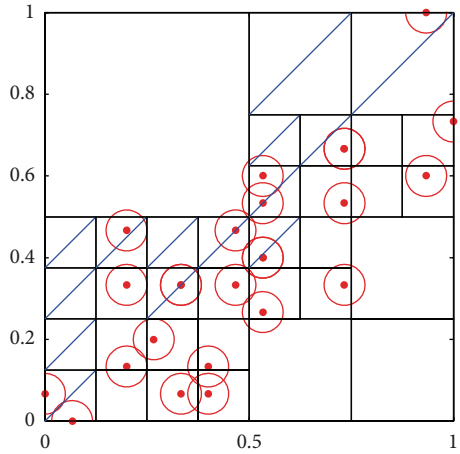


Figure 3: The neighbor grids.

$P\{z > Z_\alpha\} = a$. Then, the rejection region $W = \{(z_1, z_2, \ldots, z_n) : z > Z_\alpha\}$. So, when $(x/\sqrt{nP(1-P)}) - \sqrt{nP/(1-P)} > Z_\alpha$, z belongs to the rejection region, reject H_0, and accept H_1. That is, the non-self space coverage of detectors $> P$.

(2) If the number of non-self specimens of continuous sampling $i < N_0$, $i \cdot P$ is not too large, x approximately obeys the Poisson distribution with λ equaling $i \cdot P$. Then $P\{z > Z_\alpha\} < a$. When $(x/\sqrt{N_0P(1-P)}) - \sqrt{N_0P/(1-P)} > Z_\alpha$, the non-self space coverage of detectors $> P$. Proved.

□

From Theorem 11, in the process of detector generation, only the number of non-self specimens of continuous sampling i and the number of non-self specimens covered by detectors x need to be recorded. After sampling in the non-self space, determine whether the non-self specimen is covered by detectors of D. If not, generate a candidate detector with the position vector of this non-self specimen, and then add it into the candidate detector set CD. If so, compute whether $(x/\sqrt{N_0P(1-P)}) - \sqrt{N_0P/(1-P)}$ is larger than Z_α. If it is larger than Z_α, the non-self space coverage reaches the expected coverage P, and the sampling process stops. If not, increase i. When i is up to N_0, put candidate detectors of CD into the detector set D to change the non-self space coverage, and then set $i = 0$, $x = 0$ to restart a new round of sampling. With the continuous addition of candidate detectors, the size of the detector set D is growing, and the non-self space coverage gradually increases.

3.4. Filter Method of Candidate Detectors. When the number of sampling times in the non-self space reaches N_0, detectors of candidate detector set will be added into the detector set D. At this time, not all candidate detectors will join D, and the filtering operation will be performed for these detectors. The filtering operation consists of two parts.

The first part is to reduce the redundant coverage between candidate detectors. First, sort detectors in the candidate detector set in a descending order by the detector radius, and then judge whether the candidate detectors in the back of the sequence have been covered by the front ones. If so, this sampling of the non-self space is invalid, and the candidate detector generated from the position vector of this sampling should be deleted. There is no complete coverage between candidate detectors which have survived the first filtering operation.

The second part is to decrease the redundant coverage between mature detectors and candidate ones. The candidate detector will be matched with detectors within the grid where the detector is and neighbor grids when adding it into the detector set D, to judge whether it totally covers some mature detector. If so, the mature detector is redundant and should be removed. The filtering operations ensure that every mature detector will cover the uncovered non-self space.

The filtering process of candidate detectors is shown in Algorithm 6.

3.5. Performance Analysis. This section analyzes the performance of the algorithm from the probability theory. Assuming that the number of all the samples in the problem space is N_{Ag}, the number of antigens in the self set is N_{Self}, the number of antigens in the training set is N_s, and the number of detectors is N_d. The matching probability between a detector and an antigen is P', which is associated with

Filter(*CD*)
Input: the candidate detector set *CD*
Step 1. Sort *CD* in a descending order by the detector radius;
Step 2. Make sure that centers of detectors in the back of the sequence do not fall into the covered area of front detectors. That is to say, $dis(d_i, d_j) > r_{di}$, where $1 <= i < j <= N_{cd}$, r_{di} is the radius of detector d_i, and N_{cd} is the size of *CD*;
Step 3. Add candidate detectors into *D*, and ensure that they do not entirely cover any detector in *D*. That is to say, $dis(d_i, d_j) > r_{di}$ *or* $dis(d_i, d_j) <= r_{di}$ and $2r_{dj} > r_{di}$, where $1 <= i <= N_{cd}$, $1 <= j <= N_d$, r_{di} and r_{dj} are the radiuses of d_i and d_j respectively, and N_{cd} and N_d are the sizes of *CD and D* respectively.

Algorithm 6: The filtering process of candidate detectors.

the specific matching rule [7, 9]. $P(A)$ is defined as the probability of occurrence of event A [21].

Theorem 12. *The probability of matching an undescribed self antigen for a detector which is passed self-tolerance is* $P_d = (1 - P')^{N_s} \cdot (1 - (1 - P')^{N_{self} - N_s})$.

Proof. From the proposition, a given detector passing the self-tolerance indicates that this detector does not match any antigen in the self training set. Let event A be "the given detector does not match any antigen in the self set," event B "the given detector matches at least one antigen which is not described," then $P_d = P(A)P(B)$. In the event A, the number of times for a detector matching antigens in the self set X meets the binomial distribution, $X \sim b(N_s, P')$. Therefore, $P(A) = P(X = 0) = (1 - P')^{N_s}$. In the event B, the number of times for a detector matching undescribed self antigens Y meets the binomial distribution, $Y \sim b(N_{Self} - N_s, P')$. Then, $P(B) = 1 - P(Y = 0) = 1 - (1 - P')^{N_{Self} - N_s}$.

So, $P_d = P(A)P(B) = (1 - P')^{N_s} \cdot (1 - (1 - P')^{N_{self} - N_s})$. Proved. □

Theorem 13. *The probability of correct identification for a non-self antigen is* $P_{tp} = 1 - (1 - P')^{N_d \cdot (1 - P_d)}$, *and the probability of erroneous identification for a non-self antigen is* $P_{fn} = (1 - P')^{N_d \cdot (1 - P_d)}$. *The probability of correct identification for a self antigen is* $P_{tn} = (1 - P')^{N_d \cdot P_d}$, *and the probability of erroneous identification for a self antigen is* $P_{fp} = 1 - (1 - P')^{N_d \cdot P_d}$.

Proof. Let event A be "the given non-self antigen matches at least one detector in the detectors set." In the event A, the number of times for a non-self antigen matching detectors X meets the binomial distribution, $X \sim b(N_d \cdot (1 - P_d), P')$. Therefore, $P_{tp} = P(A) = 1 - P(X = 0) = 1 - (1 - P')^{N_d \cdot (1 - P_d)}$, and $P_{fn} = 1 - P_{tp} = (1 - P')^{N_d \cdot (1 - P_d)}$.

Let event B be "the given self antigen does not match any detector in the detectors set." In the event B, the number of times for a self antigen matching detectors Y meets the binomial distribution, $Y \sim b(N_d \cdot P_d, P')$. Therefore, $P_{tn} = P(B) = P(Y = 0) = (1 - P')^{N_d \cdot P_d}$, and $P_{fp} = 1 - P_{tn} = 1 - (1 - P')^{N_d \cdot P_d}$. Proved. □

P' is substantially constant for specific matching rules [7, 9]. Assuming that $P' = 0.005$ and $N_{Self} = 1000$, then Figure 4 shows variations of P_{tp}, P_{fn}, P_{fp}, and P_{tn} under the effects of N_d and N_s. As can be seen from the figure, when the number of selves in the training set N_s and the number of detectors N_d are larger, the probability of correct identification for an arbitrary given non-self antigen P_{tp} is greater, the probability of erroneous identification P_{fn} is small, and variation tendencies of P_{tp} and P_{fn} are not large while N_d and N_s change. Thus, when the coverage of non-self space for the detector set is certain, the detection rates of different algorithms are relatively close. When N_s and N_d are larger, the probability of correct identification for an arbitrary given self antigen P_{tn} is greater, the probability of erroneous identification P_{fp} is small, and variation tendencies of P_{tn} and P_{fp} are large while N_d and N_s change. So, when the coverage of non-self space for the detector set is certain, the false alarm rate of GB-RNSA is smaller for that the algorithm significantly reduces the number of detectors.

3.6. Time Complexity Analysis

Theorem 14. *The time complexity of detector generation process in GB-RNSA is* $O((|D|/(1 - P'))(N_s + |D|^2))$, *where* N_s *is the size of the training set,* $|D|$ *is the size of the detector set, and* P' *is the average self-reactive rate of detectors.*

Proof. For GB-RNSA, the main time cost of generating a new mature detector includes the time spending of calling *FindGrid* to find the grid, the time spending of self-tolerance for candidate detectors, and the time spending of call *Filter* to screen detectors.

Known from Section 3.2, the depth of 2^n-tree is $Ceil(\log_2(1/(4r_s + 4r_{ds})))$. So, for a new detector, the time complexity of finding the grid $grid'$ where the detector is $t1 = O(Ceil(\log_2(1/(4r_s + 4r_{ds})))^n)$. n is the space dimension, r_s is the radius of selves, and r_{ds} is the smallest radius of detectors. So, $t1$ is relatively constant.

Calculating the radius of the new detector needs to compute the nearest distance with selves in the grid where the detector is and neighbors. The time complexity is $t2 = O(N_{s'})$, where $N_{s'}$ is the number of selves in $grid'$ and neighbors.

The time complexity of calculating whether the new detector is covered by existing detectors is $t3 = O(D')$, where D' is the number of detectors in $grid'$ and neighbors.

The time complexity of calling *Filter* to screen detectors includes the time spending of sorting the candidate detectors

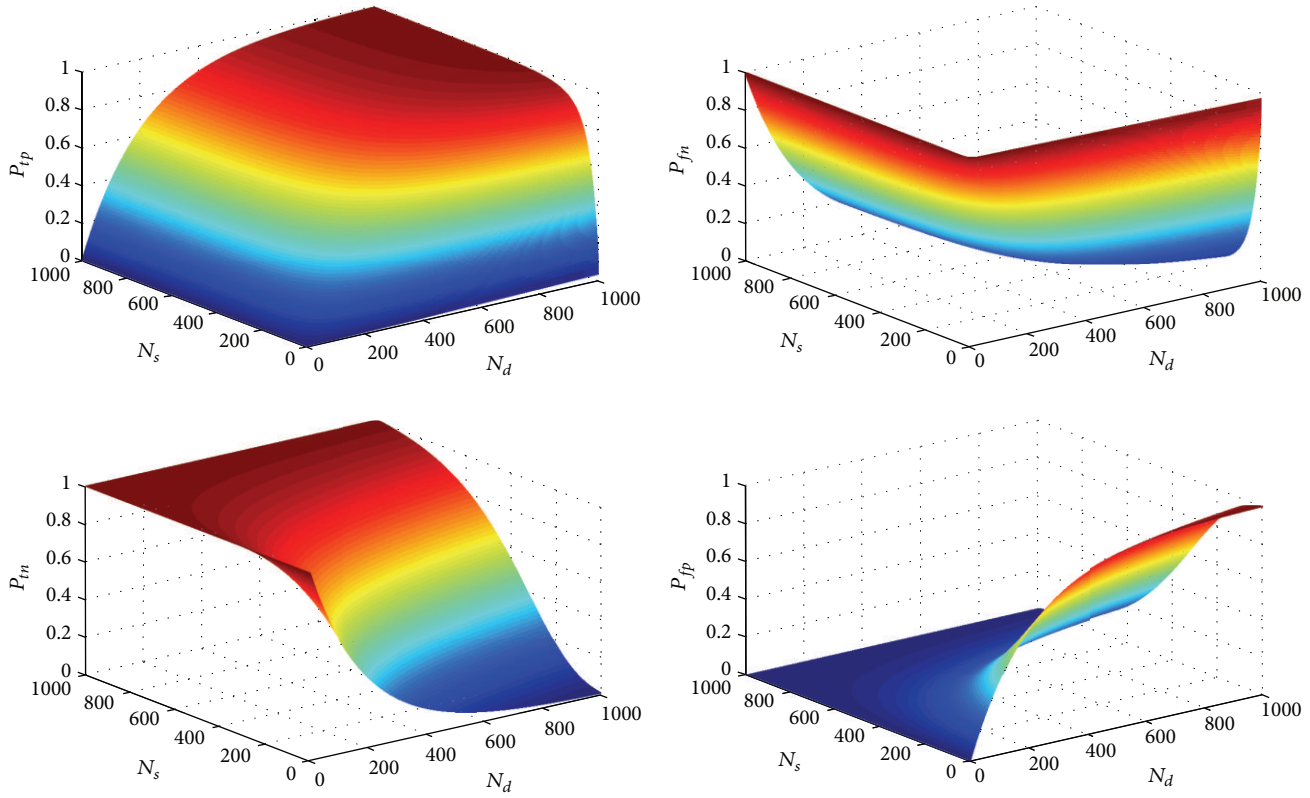


Figure 4: Simulations of Theorem 13.

and judging whether redundant coverage exists; that is, $t4 = O(N_0^2 + N_0 \cdot D')$.

Suppose N' is the number of candidate detectors to generate the detector set D, then the time complexity of sampling is $N' \cdot (t1 + t2) + N' \cdot (1 - P') \cdot t3 + (N'/N_0) \cdot t4$. And $N' \approx |D|/(1 - P')$, so, the time complexity of generating the detector set D is as follows:

$$O\left(\frac{|D|}{1-P'}\left(t1 + \sum N_{s'}\right) + |D|\left(\sum D'\right) + \frac{|D|\left(N_0 + \sum D'\right)}{(1-P')}\right)$$
$$= O\left(\frac{|D|}{1-P'}N_s + |D|^2 + \frac{|D|^2}{1-P'}\right)$$
$$= O\left(\frac{|D|}{1-P'}\left(N_s + |D|^2\right)\right). \quad (5)$$

So, the time complexity of detector generation process in GB-RNSA is $O((|D|/(1 - P'))(N_s + |D|^2))$. Proved. □

SNSA, RNSA, and V-Detector are the main detector generation algorithms and are widely used in the fields of artificial immune-based pattern recognition, anomaly detection, immune optimization, and so forth. Table 1 shows the comparisons of these negative selection algorithms and GB-RNSA. As seen from Table 1, the time complexity of traditional algorithms is exponential to the size of selves N_s. When the number of self elements increases, the time cost will rapidly increase. GB-RNSA eliminates the exponential impact and reduces the influence of growth of selves' scale on the time cost. So, GB-RNSA lowers the time complexity of the original algorithm and improves the efficiency of detector generation.

Table 1: Comparisons of time complexity.

Algorithm	Time complexity
SNSA	$O\left(\frac{-\ln(P_f) \cdot N_s}{P(1-P')^{N_s}}\right)$ [7]
RNSA	$O\left(\frac{\lvert D\rvert \cdot N_s}{(1-P')^{N_s}}\right)$ [11]
V-Detector	$O\left(\frac{\lvert D\rvert \cdot N_s}{(1-P')^{N_s}}\right)$ [13]
GB-RNSA	$O\left(\frac{\lvert D\rvert}{1-P'}\left(N_s + \lvert D\rvert^2\right)\right)$

4. Experimental Results and Analysis

This section validates the effectiveness of GB-RNSA through experiments. Two types of data sets are selected for the experiments which are commonly used in the study of real-valued negative selection algorithms, including 2D synthetic datasets [22] and UCI datasets [20]. 2D synthetic datasets

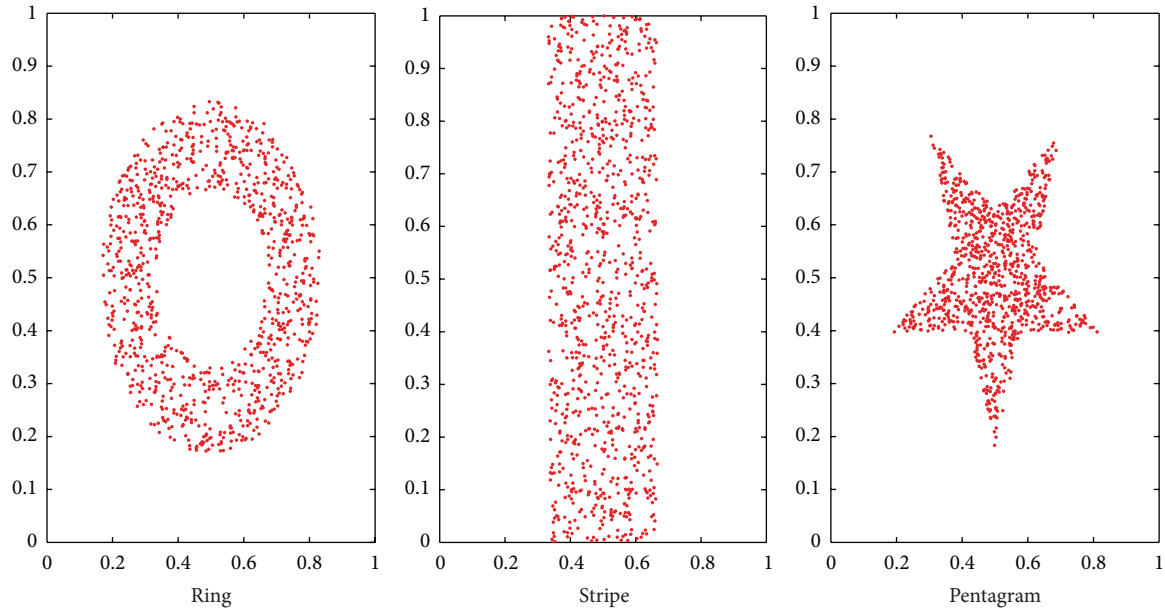


Figure 5: Distributions of Ring, Stripe, and Pentagram datasets.

Table 2: Effects of different self radiuses.

Datasets	Self radius $r_s = 0.02$		Self radius $r_s = 0.1$		Self radius $r_s = 0.2$	
	DR%	*FAR*%	*DR*%	*FAR*%	*DR*%	*FAR*%
Ring	81.55 (1.02)	62.11 (2.14)	61.77 (1.39)	12.04 (1.24)	32.39 (1.42)	0.00 (0.00)
Stripe	80.21 (1.24)	63.34 (1.90)	58.52 (1.18)	11.20 (2.47)	25.93 (1.88)	0.00 (0.00)
Pentagram	77.09 (1.38)	67.02 (2.32)	57.65 (2.31)	13.19 (1.63)	22.78 (1.59)	0.00 (0.00)

Table 3: Effects of different sizes of the training set.

Datasets	Size of the training set $N_s = 100$		Size of the training set $N_s = 500$		Size of the training set $N_s = 800$	
	DR%	*FAR*%	*DR*%	*FAR*%	*DR*%	*FAR*%
Ring	22.54 (1.22)	76.26 (2.05)	86.09 (1.16)	8.21 (1.21)	95.92 (1.37)	0.00 (0.00)
Stripe	18.25 (1.98)	78.92 (2.32)	80.13 (1.87)	9.05 (1.44)	87.63 (1.78)	0.00 (0.00)
Pentagram	12.20 (1.55)	88.29 (2.87)	72.33 (1.91)	11.42 (1.41)	82.18 (1.49)	0.00 (0.00)

Table 4: Experimental parameters of UCI datasets.

Datasets	Record numbers	Properties	Types	Self sets	Nonself sets	Training set and its size	Test set and its size
Iris	150	4	Real	Setosa: 50	Versicolour: 50 Virginica: 50	Setosa: 25	Setosa: 25 Versicolour: 25 Virginica: 25
Haberman's Survival	306	3	Integer	Survived: 225	Died: 81	Survived: 150	Survived: 50 Died: 50
Abalone	4177	8	Real, integer	M: 1528	F: 1307 I: 1342	M: 1000	M: 500 F: 500 I: 500

are authoritative in the performance test of real-valued negative selection algorithms [13, 19, 22], which is provided by Professor Dasgupta's research team of Memphis University. UCI datasets are classic machine learning data sets, which are widely used in the tests of detectors' performance and generation efficiencies [11, 18, 19, 23]. In the experiments, two traditional real-valued negative selection algorithms RNSA and V-Detector are chosen to compare with.

The number of mature detectors *DN*, the detection rate *DR*, the false alarm rate *FAR*, and the time cost of detectors

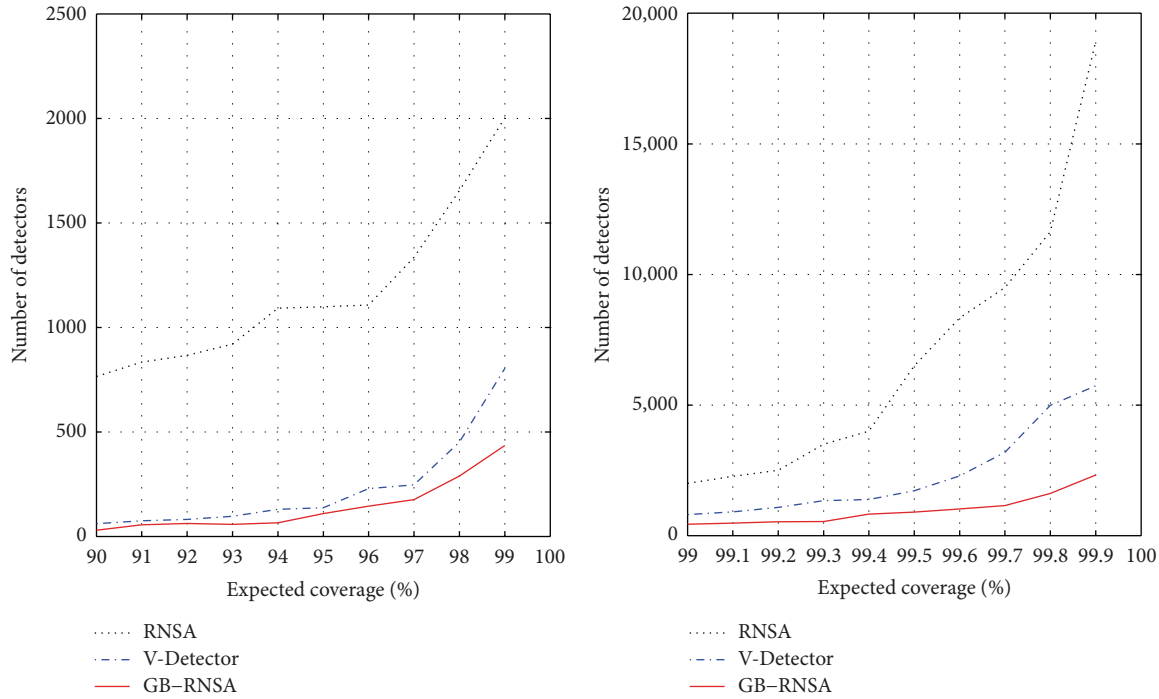


Figure 6: Comparisons of the numbers of detectors for RNSA, V-Detector, and GB-RNSA (dataset of Haberman's Survival is adopted; the radius of self antigen is 0.1).

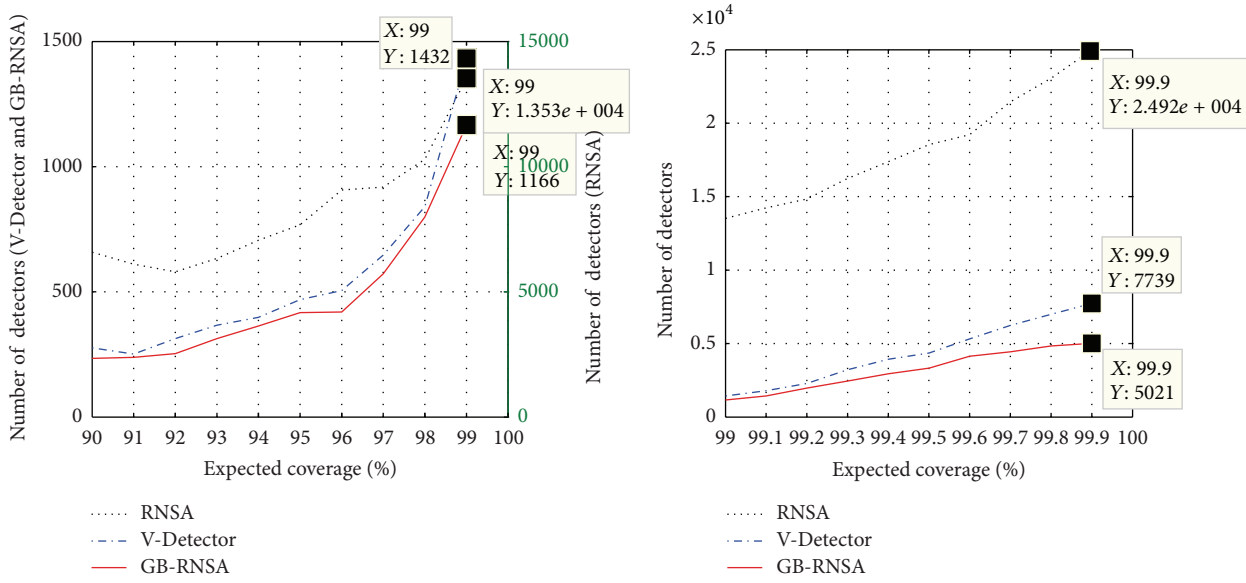


Figure 7: Comparisons of the numbers of detectors for RNSA, V-Detector, and GB-RNSA (dataset of Iris is adopted; the radius of self antigen is 0.1).

generations *DT* are adopted to measure the effectiveness of the algorithms in the experiments. Because the traditional algorithm RNSA uses the preset number of detectors as the termination condition, this paper modified RNSA and uses the expected coverage of non-self space as the termination condition, in order to ensure that the three algorithms are under the same experimental conditions to make valid comparisons.

4.1. 2D Synthetic Datasets. These datasets consist of several different subdatasets. We choose Ring, Stripe, and Pentagram subdatasets to test the performance of detectors generation of GB-RNSA. Figure 5 shows the distributions of these three datasets in two-dimensional real space.

The size of self sets of the three datasets is $N_{Self} = 1000$. The training set is composed of data points randomly selected from the self set, and the test data is randomly

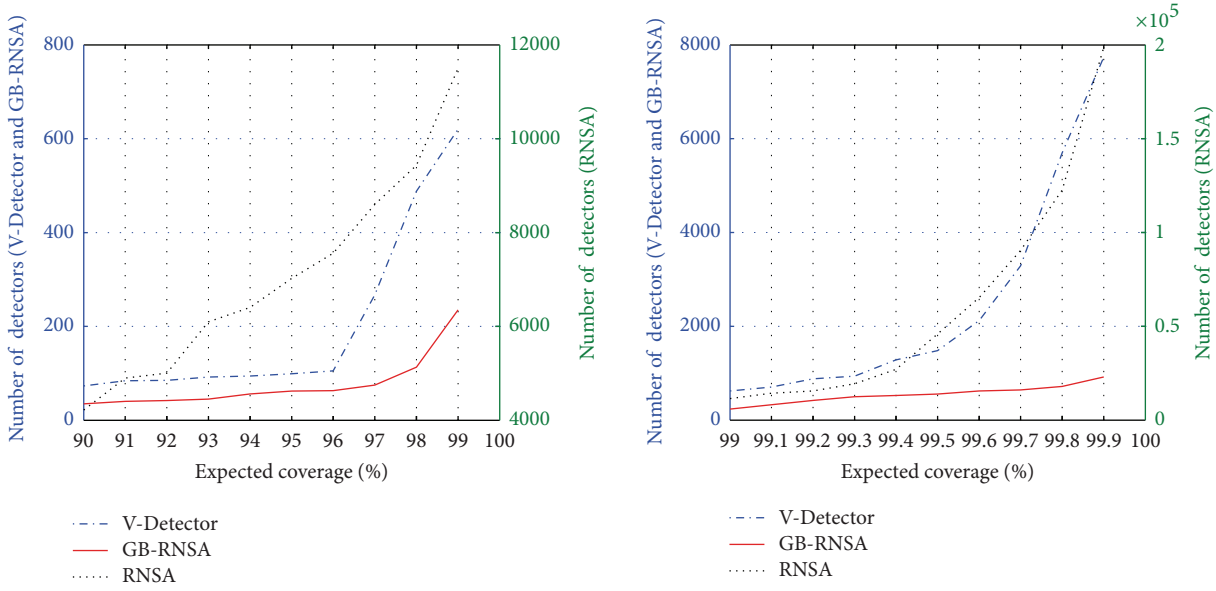


Figure 8: Comparisons of the numbers of detectors for RNSA, V-Detector, and GB-RNSA (dataset of Abalone is adopted; the radius of self antigen is 0.1).

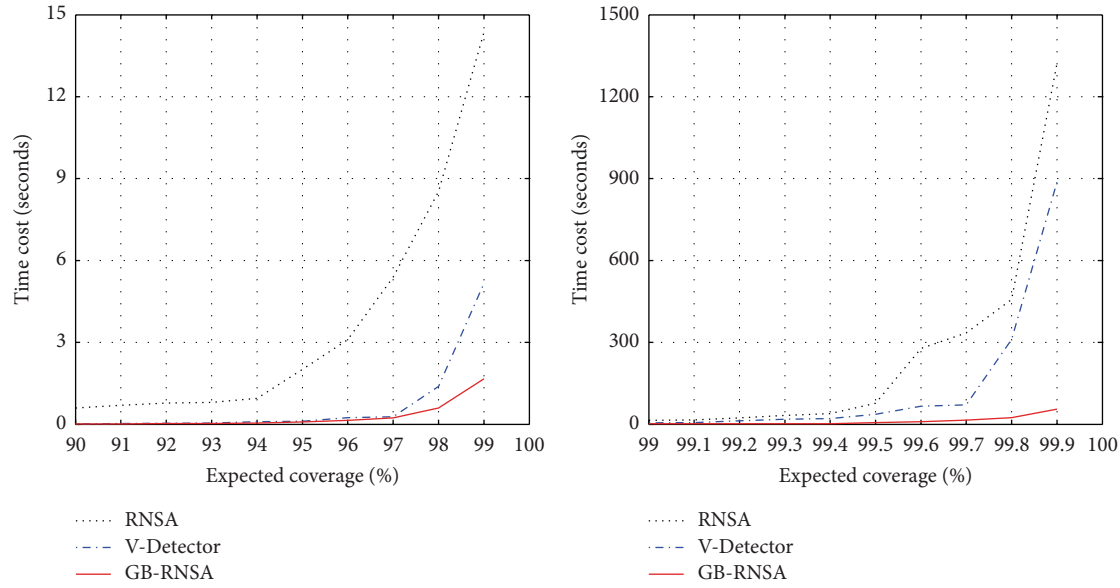


Figure 9: Comparisons of time costs of RNSA, V-Detector, and GB-RNSA (dataset of Haberman's Survival is adopted; the radius of self antigen is 0.1).

selected from the two-dimensional $[0, 1]$ space. The experiments were repeated 20 times and the average values were adopted. Experimental results are shown in Tables 2 and 3, where values within parenthesis are variances. Table 2 lists comparisons of detection rates and false alarm rates of GB-RNSA in the three datasets under the same expected coverage of 90%, the same training set $N_s = 300$, and different self radiuses. As can be seen, the algorithm has higher detection rate and false alarm rate under smaller self radius, while the algorithm has lower detection rate and false alarm rate under greater self radius. Table 3 lists comparisons of detection rates and false alarm rates of GB-RNSA in the three datasets under the same expected coverage of 90%, the same self radius $r_s = 0.05$ and different sizes of training set. The detection rate increases gradually and the false alarm rate decreases gradually while the size of the training set grows.

4.2. UCI Datasets. Three standard UCI data sets including Iris, Haberman's Survival and Abalone, are chosen to do the experiments, and experimental parameters are shown in Table 4. For the three data sets, self set and non-self set are chosen randomly, and records of training set and test set are

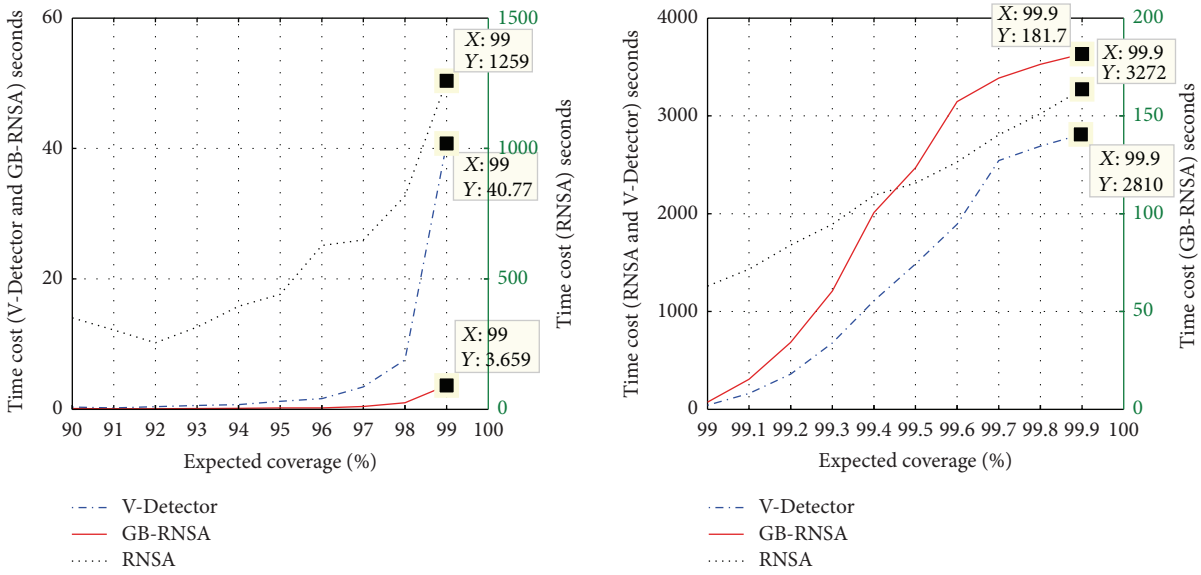


Figure 10: Comparisons of time costs of RNSA, V-Detector, and GB-RNSA (dataset of Iris is adopted; the radius of self antigen is 0.1).

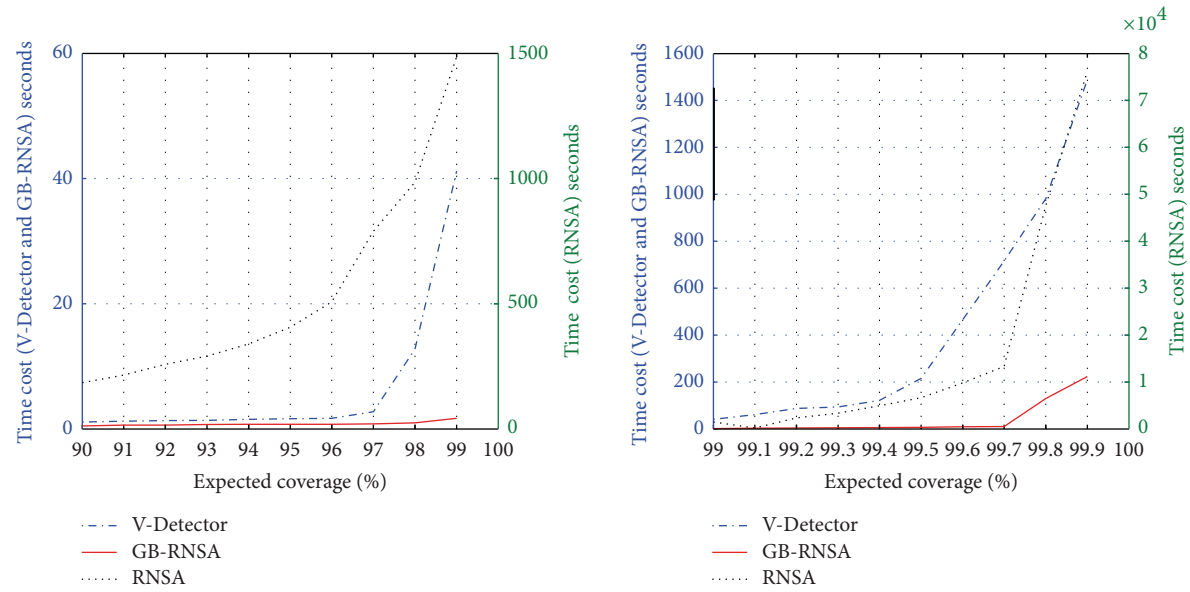


Figure 11: Comparisons of time costs of RNSA, V-Detector, and GB-RNSA (dataset of Abalone is adopted; the radius of self antigen is 0.1).

chosen randomly as well. The experiments were repeated 20 times and the average values were adopted.

4.2.1. Comparisons of the Number of Detectors. Figures 6, 7, and 8 show the number of mature detectors of RNSA, V-Detector, and GB-RNSA on the three data sets. Seen from the figures, with the increase of the expected coverage, the number of detectors which are needed to meet the coverage requirements for the three algorithms correspondingly increases. But the efficiency of GB-RNSA is significantly better than those of RNSA and V-Detector. For the data set of Iris, to achieve the expected coverage 99%, RNSA needs 13527 mature detectors, V-Detector needs 1432, and GB-RNSA needs 1166 which decreases about 91.4% and 18.6%, respectively. For the larger data set of Abalone, to achieve the expected coverage 99%, RNSA needs 11500 mature detectors, V-Detector needs 620, and GB-RNSA needs 235 which decreases about 98% and 62.1%, respectively. Thus, under the same expected coverage, different data dimensions, and different training sets, the number of mature detectors generated by GB-RNSA is significantly reduced compared with RNSA and V-Detector.

4.2.2. Comparisons of the Cost of Detectors' Generations. Figures 9, 10, and 11 show the time costs of detectors' generation of RNSA, V-Detector, and GB-RNSA on the

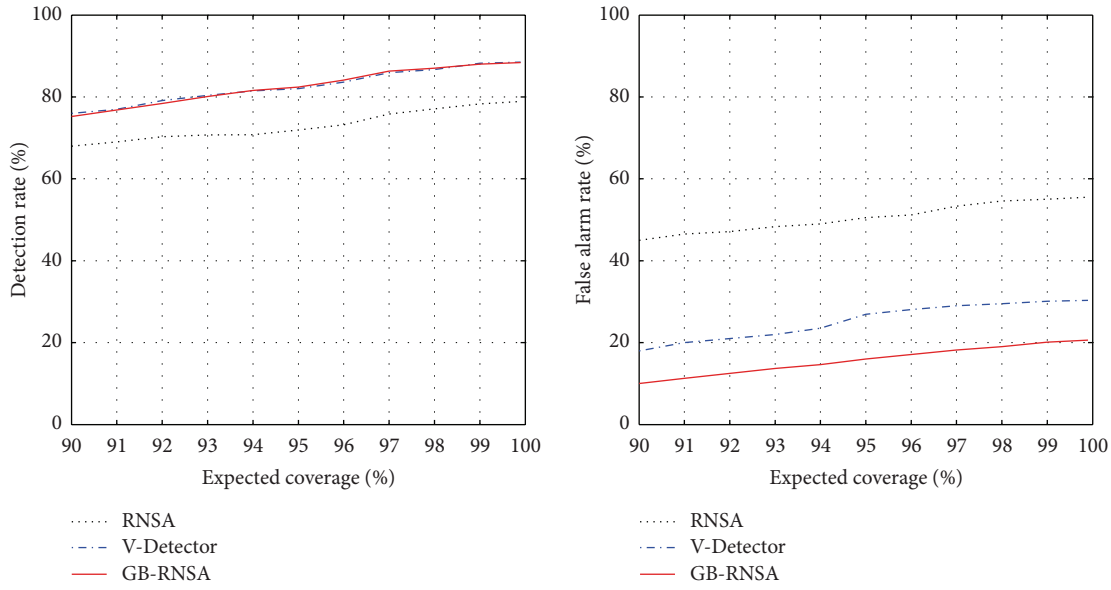


Figure 12: Comparisons of DR and FAR of RNSA, V-Detector, and GB-RNSA (dataset of Haberman's Survivalis is adopted; the radius of self antigen is 0.1).

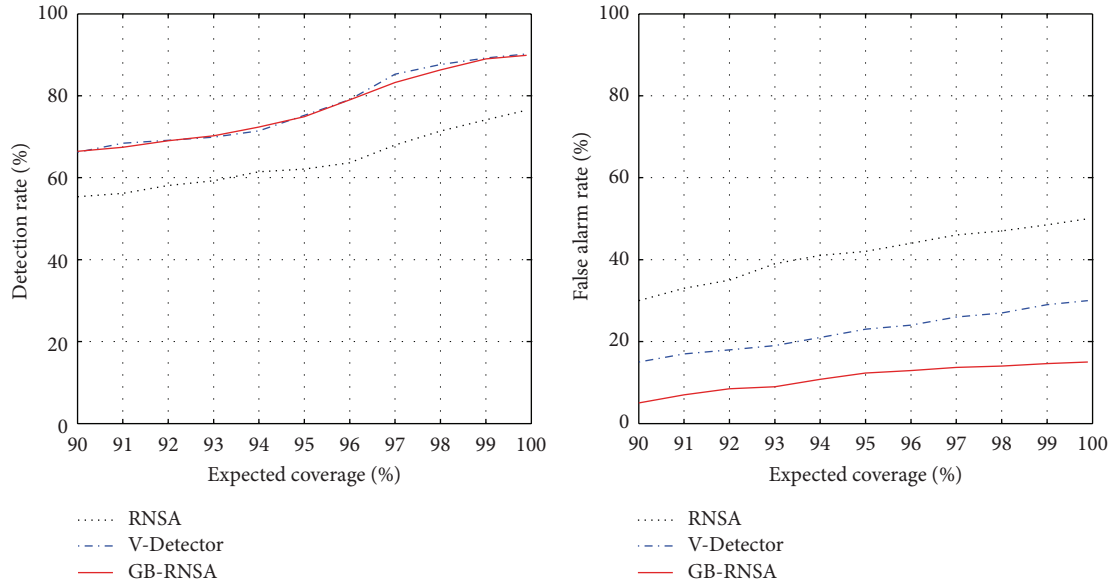


Figure 13: Comparisons of DR and FAR of RNSA, V-Detector, and GB-RNSA (dataset of Iris is adopted; the radius of self antigen is 0.1).

three data sets. As seen from the figures, with the increase of the expected coverage, the time cost of RNSA and V-Detector is in a sharp increase, while that of GB-RNSA is in a slow growth. For the data set of Iris, to achieve the expected coverage of 90%, the time cost of RNSA is 350.187 seconds, that of V-Detector is 0.347 seconds, and that of GB-RNSA is 0.1 seconds which decreases about 99.97% and 71.2%, respectively; when the expected coverage is 99%, the time cost of RNSA is 1259.047 seconds, that of V-Detector is 40.775 seconds, and that of GB-RNSA is 3.659 seconds which decreases about 99.7% and 91.0%, respectively. For the other two datasets, experimental results are similar. Thus, compared with RNSA and V-Detector, the effectiveness of detectors' generation of GB-RNSA is promoted.

4.2.3. Comparisons of Detection Rates and False Alarm Rates. Figures 12, 13, and 14 show the detection rates and false alarm rates of RNSA, V-Detector, and GB-RNSA on the three data sets. As seen from the figures, when the expected coverage is large than 90%, the detection rates of the three algorithms are similar, and that of RNSA is slightly lower, while the false alarm rate of GB-RNSA is obviously lower than those of RNSA and V-Detector. For the data set of Haberman's

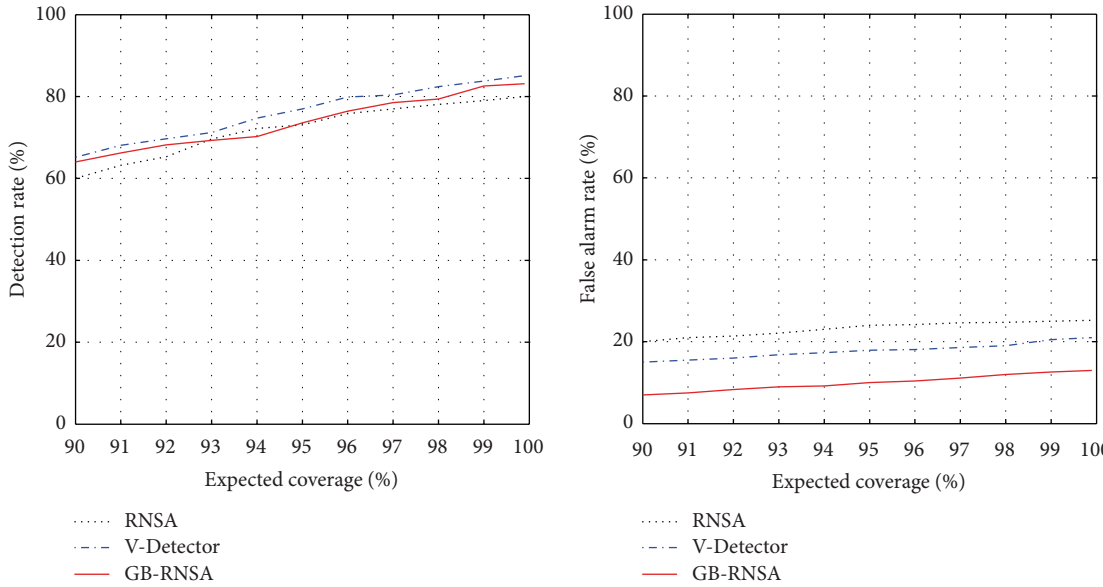


Figure 14: Comparisons of DR and FAR of RNSA, V-Detector, and GB-RNSA (dataset of Abalone is adopted; the radius of self antigen is 0.1).

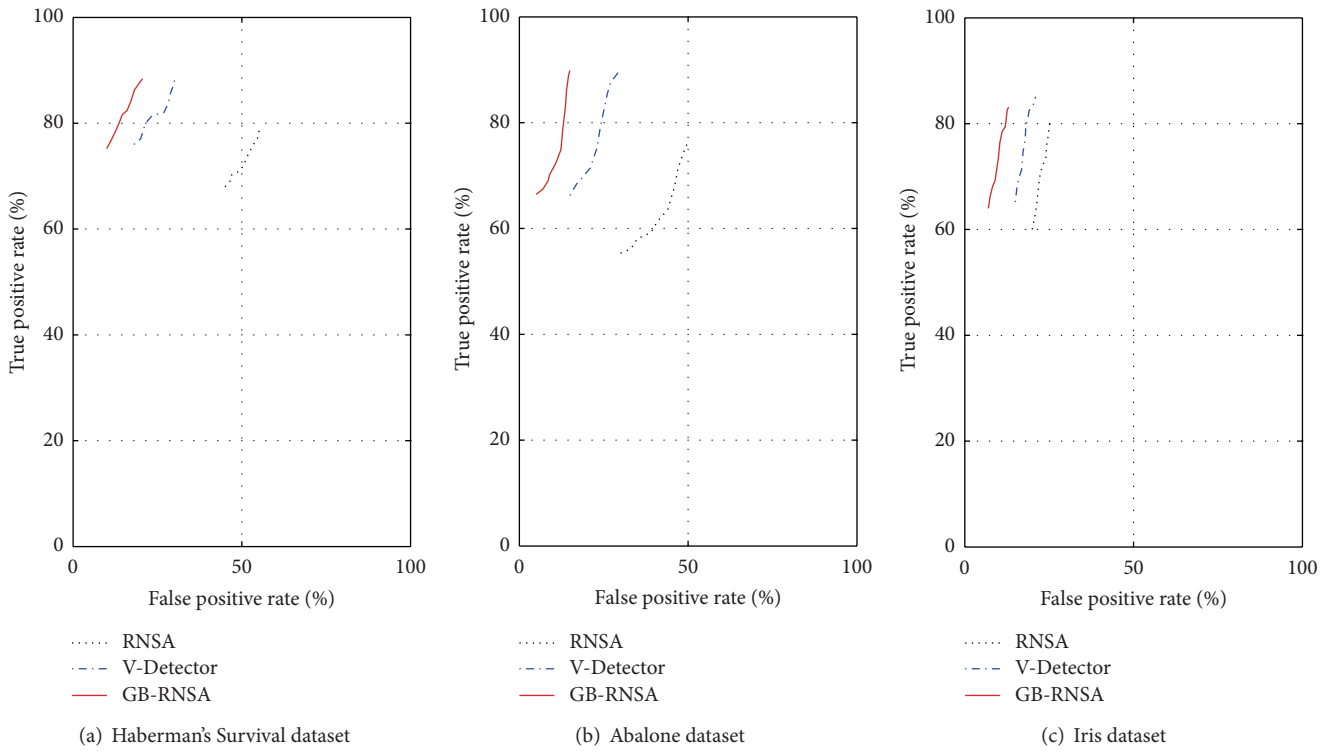


Figure 15: ROC curves of RNSA, V-Detector, and GB-RNSA.

Survival, when the expected coverage is 99%, the false alarm rate of RNSA is 55.2%, that of V-Detector is 30.1%, and that of GB-RNSA is 20.1% which decreases about 63.6% and 33.2%, respectively. For the data set of Abalone, when the expected coverage is 99%, the false alarm rate of RNSA is 25.1%, that of V-Detector is 20.5%, and that of GB-RNSA is 12.6% which decreases about 49.8% and 38.5%, respectively. Thus, under the same expected coverage, the false alarm rate of GB-RNSA is significantly lower compared with RNSA and V-Detector.

The ROC curve is a graphical method for the classification model using true positive rate and false positive rate. In NSAs, true positive rate is the detection rate and false positive rate is the false alarm rate. Figure 15 shows the ROC curves of RNSA, V-Detector, and GB-RNSA on the three data sets.

A good classification model should be as close as possible to the upper-left corner of the graphic. As seen from Figure 15, GB-RNSA is better than RNSA and V-Detector.

5. Conclusion

Too many detectors and high time complexity are the major problems of existing negative selection algorithms, which limit the practical applications of NSAs. There is also a problem of redundant coverage of non-self space for detectors in NSAs. A real-valued negative selection algorithm based on grid for anomaly detection GB-RNSA is proposed in this paper. The algorithm analyzes distributions of the self set in the real space and divides the space into grids by certain methods. The randomly generated candidate detector only needs to match selves who are in the grid where the detector is and in its neighbor grids. And before the candidate detector is added into the mature detector set, certain methods are adopted to reduce the duplication coverage. Theory analysis and experimental results demonstrate that GB-RNSA has better time efficiency and detector quality compared with classical negative selection algorithms and is an effective artificial immune algorithm to generate detectors for anomaly detection.

Acknowledgments

This work has been supported by the National Natural Science Foundation of China under Grant no. 61173159, the National Natural Science Foundation of China under Grant no. 60873246, and the Cultivation Fund of the Key Scientific and Technical Innovation Project, Ministry of Education of China, under Grant no. 708075.

References

[1] D. Dasgupta, S. Yu, and F. Nino, "Recent advances in artificial immune systems: models and applications," *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 1574–1587, 2011.

[2] P. Bretscher and M. Cohn, "A theory of self-nonself discrimination," *Science*, vol. 169, no. 3950, pp. 1042–1049, 1970.

[3] F. Burnet, *The Clonal Selection Theory of Acquired Immunity*, Vanderbilt University Press, Nashville, Tenn, USA, 1959.

[4] N. K. Jerne, "Towards a network theory of the immune system," *Annals of Immunology*, vol. 125, no. 1-2, pp. 373–389, 1974.

[5] P. Matzinger, "The danger model: a renewed sense of self," *Science*, vol. 296, no. 5566, pp. 301–305, 2002.

[6] M. L. Kapsenberg, "Dendritic-cell control of pathogen-driven T-cell polarization," *Nature Reviews Immunology*, vol. 3, no. 12, pp. 984–993, 2003.

[7] S. Forrest, L. Allen, A. S. Perelson, and R. Cherukuri, "Self-nonself discrimination in a computer," in *Proceedings of the IEEE Symposium on Research in Security and Privacy*, pp. 202–212, May 1994.

[8] T. Li, *Computer Immunology*, House of Electronics Industry, Beijing, China, 2004.

[9] T. Li, "Dynamic detection for computer virus based on immune system," *Science in China F*, vol. 51, no. 10, pp. 1475–1486, 2008.

[10] T. Li, "An immunity based network security risk estimation," *Science in China F*, vol. 48, no. 5, pp. 557–578, 2005.

[11] F. A. González and D. Dasgupta, "Anomaly detection using real-valued negative selection," *Genetic Programming and Evolvable Machines*, vol. 4, no. 4, pp. 383–403, 2003.

[12] Z. Ji, *Negative selection algorithms: from the thymus to V-detector [Ph.D. dissertation]*, University of Memphis, Memphis, Tenn, USA, 2006.

[13] Z. Ji and D. Dasgupta, "V-detector: an efficient negative selection algorithm with "probably adequate" detector coverage," *Information Science*, vol. 19, no. 9, pp. 1390–1406, 2009.

[14] X. Z. Gao, S. J. Ovaska, and X. Wang, "Genetic algorithms-based detector generation in negative selection algorithm," in *Proceedings of the IEEE Mountain Workshop on Adaptive and Learning Systems (SMCals '06)*, pp. 133–137, July 2006.

[15] X. Z. Gao, S. J. Ovaska, X. Wang, and M.-Y. Chow, "Clonal optimization of negative selection algorithm with applications in motor fault detection," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '06)*, pp. 5118–5123, Taipei, Taiwan, October 2006.

[16] J. M. Shapiro, G. B. Lament, and G. L. Peterson, "An evolutionary algorithm to generate hyper-ellipsoid detectors for negative selection," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '05)*, pp. 337–344, Washington, DC, USA, June 2005.

[17] M. Ostaszewski, F. Seredynski, and P. Bouvry, "Immune anomaly detection enhanced with evolutionary paradigms," in *Proceedings of the 8th Annual Genetic and Evolutionary Computation Conference (GECCO '06)*, pp. 119–126, Seattle, Wash, USA, July 2006.

[18] T. Stibor, P. Mohr, and J. Timmis, "Is negative selection appropriate for anomaly detection?" in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '05)*, pp. 569–576, IEEE Computer Society Press, June 2005.

[19] W. Chen, X. Liu, T. Li, Y. Shi, X. Zheng, and H. Zhao, "A negative selection algorithm based on hierarchical clustering of self set and its application in anomaly detection," *International Journal of Computational Intelligence Systems*, vol. 4, no. 4, pp. 410–419, 2011.

[20] "UCI Dataset," http://archive.ics.uci.edu/ml/datasets.

[21] G. Chang and J. Shi, *Mathematical Analysis Tutorial*, Higher Education Press, Beijing, China, 2003.

[22] F. Gonzalez, D. Dasgupta, and J. Gomez, "The effect of binary matching rules in negative selection," in *Proceedings of the Genetic and Evolutionary Computation (GECCO '03)*, pp. 196–206, Springer, Berlin, Germany, 2003.

[23] T. Stibor, J. Timmis, and C. Eckert, "On the appropriateness of negative selection defined over hamming shape-space as a network intrusion detection system," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '05)*, pp. 995–1002, September 2005.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 242941, 12 pages
http://dx.doi.org/10.1155/2013/242941

Research Article

Seismic Design Value Evaluation Based on Checking Records and Site Geological Conditions Using Artificial Neural Networks

Tienfuan Kerh,[1] Yutang Lin,[1] and Rob Saunders[2]

[1] *Department of Civil Engineering, National Pingtung University of Science and Technology, Pingtung 91207, Taiwan*
[2] *Faculty of Architecture, Design and Planning, University of Sydney, Sydney, NSW 2006, Australia*

Correspondence should be addressed to Tienfuan Kerh; tfkerh@gmail.com

Received 8 February 2013; Accepted 25 April 2013

Academic Editor: Fuding Xie

This study proposes an improved computational neural network model that uses three seismic parameters (i.e., local magnitude, epicentral distance, and epicenter depth) and two geological conditions (i.e., shear wave velocity and standard penetration test value) as the inputs for predicting peak ground acceleration—the key element for evaluating earthquake response. Initial comparison results show that a neural network model with three neurons in the hidden layer can achieve relatively better performance based on the evaluation index of correlation coefficient or mean square error. This study further develops a new weight-based neural network model for estimating peak ground acceleration at unchecked sites. Four locations identified to have higher estimated peak ground accelerations than that of the seismic design value in the 24 subdivision zones are investigated in Taiwan. Finally, this study develops a new equation for the relationship of horizontal peak ground acceleration and focal distance by the curve fitting method. This equation represents seismic characteristics in Taiwan region more reliably and reasonably. The results of this study provide an insight into this type of nonlinear problem, and the proposed method may be applicable to other areas of interest around the world.

1. Introduction

Seismic design values play an important role in constructing buildings to comply with regional safety standards that consider the effects of strong ground motions. Taiwan is an island located in the circum-Pacific seismic zone, sometimes called the Ring of Fire. Because earthquakes occur frequently in this area, this factor must be taken into account in structural analysis and design. After a few times of revisions and adjustments, the current building code in Taiwan classifies the entire island into two zones: the earthquake area coefficient of horizontal acceleration is 0.33 g for Zone A and 0.23 g for Zone B [1, 2]. These design values can be used to calculate earthquake force and should be examined as often as possible to determine their fit with actual conditions, either from a practical viewpoint or academic viewpoint.

There exist various types of earthquake problems; a typical case study for estimating peak ground acceleration (PGA) and a detailed review of recent efforts in predictions can be seen in the previous literatures [3, 4]. The present study focuses on the topic of using seismic recorded parameters and site soil conditions to evaluate the potential damage resulting from ground strong motions. The conventional methods of using seismic parameters to evaluate the potential damage of earthquakes are primarily based on vibration analysis and regression analysis. However, the first method often involves very tedious calculations, and the second method must assume a function in advance [5, 6]. Therefore, recently developed techniques in the field of computational intelligence, including neural networks and genetic algorithms, may be a better alternative for solving earthquake-related problems around the world because of their simplicity and effectiveness [7–18]. For more specific areas in Taiwan, the seismic key element, that is, PGA, can be estimated using neural network models trained on a series of historical seismic recorded data [19, 20]. An improved model that uses a combination of genetic algorithms and neural networks can also be found to be useful for solving the problem of checking the seismic design values [21, 22]. Previous studies have shown that

the seismic parameters of local magnitude (ML), epicentral distance (Di), and focal depth (De) in the learned model can achieve acceptable performance in estimating the PGA in various engineering projects and identifying potentially hazardous zones.

Regardless of whether the hypocenter is located under the sea or under the ground surface, seismic waves generally propagate through various strata to the ground surface, and their characteristics can be recorded by precision instruments installed in checking stations. Therefore, the geological conditions of site may have a significant effect on the ground motion caused by the earthquake. Previous studies dealing with this problem in several regions have shown that the seismic ground acceleration and response spectrum vary with the site soil conditions [23, 24]. In the case of predicting the PGA, the site geological conditions may be used as an input with the three basic seismic parameters (ML, Di, and De) in the neural network model. For example, the constant values 1, 3, and 5 representing rocky soil, stiff soil, and soft soil, respectively, can be used to develop a neural network model [25]. However, this model seems to perform poorly because the classification of site conditions is too rough and the input constants may be insensitive to the model. A better use of site conditions, including the thickness and mean frequency of shear waves, in the neural network model is more robust than classical models [26]. Studies on this topic have revealed that different parameters of site conditions in the input layer may influence the performance of the neural network model in predicting the PGA.

This study proposes a new set of input parameters in the neural network model for estimating the PGA for 86 checking stations spread across the island of Taiwan. Further to say is that three seismic parameters including local magnitude, epicentral distance, and focal depth collected from a series of historical checking records and two site soil test results including standard penetration test value (SPT-N) and shear wave velocity (V_s) are taken for training, validating, and testing the model. This study also develops a new weight-based neural network model with spatial relationship to estimate PGA at 24 unchecked sites, and the result may represent a new earthquake response at each of the subdivision zones. This study compares estimations with design values in the building code to identify potentially hazardous zones. Finally, this study develops an equation for linking the horizontal peak ground acceleration (PGA_H) and focal distance (D_f) in accordance with neural network estimates. The method adopted in this study and the obtained results may be useful in relevant engineering fields and might be applicable to other areas of interest around the world.

2. Research Area and Geological Condition

Based on a report from the Seismological Center of Central Weather Bureau, there are approximately 18000 strong ground motions per year in Taiwan and approximately 1000 of these strong ground motions can be felt by humans. According to the most recent report from the Central Geological Survey, there are 33 active faults in the Taiwan area, and these faults may create a place for releasing energy during an earthquake. A total of 99 recorded earthquakes have caused destructive results in the period from 1901 to 2009. This reveals the frequent occurrence of large-scale earthquakes on this island [27, 28]. Therefore, it is essential to check the effects of strong ground motions at construction sites to reduce the risk of future damage.

Most antiearthquake designs are based on the earthquake level and a recurrence period of 475 years, which is equivalent to approximately 10% of probability during 50 years of structural usage. In addition, if the design adopts a seismic isolation system, then over 2% of probability during 50 years of usage is considered in the building code. Therefore, the coefficient of horizontal spectral acceleration for a construction site design is determined from the above-mentioned potential damage. The analysis of potential damage must consider local magnitude, hypocenter, epicenter depth of past earthquakes, and activity of faults potential within approximately 200 km of the construction site. Because using the horizontal PGA in this potential damage analysis can become very complicated, a zone division is required to facilitate earthquake design work.

As indicated previously, the earthquake area coefficients of horizontal acceleration for Zone A and Zone B are 0.33 g and 0.23 g, respectively, where 1 g = 981 gal ($\mathrm{cm/s^2}$), for calculating earthquake force. These values can be used as a basis to check the present neural network estimation in 24 seismic subdivision zones for the whole island of Taiwan. Figure 1 shows a sketch of the present research area, where Zone A has 17 subdivision zones (A1–A17) and Zone B has seven subdivision zones (B1–B7). For each subdivision zone, seismic data sets from two to four checking stations around the zone recorded from the year 1994 to the year 2011 were used for analysis.

A typical earthquake record as seen in Table 1 includes several items, such as date and time, exact location in longitude and latitude, intensity, local magnitude, epicenter depth, epicentral distance, and PGA in different directions. However, the main seismic parameters for analysis in this study are local magnitude on the Richter scale, epicenter depth, epicentral distance, and PGA in vertical (V), North-South (N-S), and East-West (E-W) directions, respectively. Taiwan includes three major regions of geological conditions: (1) central mountain range region, (2) western foothill region, and (3) eastern coastal range region. From a plate tectonics viewpoint, the first region consists primarily of sedimentary rock; the second region consists primarily of sandstone and shale; the third region is a part of the island arc of the Philippine sea plate, which consists of igneous rock and sedimentary rock [29]. The western foothill range region is generally softer than the other two regions because of its geologically loose structure. Hence, ground motion in this region may be more sensitive to site effects and should be considered more carefully in engineering design.

Figure 2 shows a typical example of a stratum boring test result provided by the National Center for Research on Earthquake Engineering (NCREE) in Taiwan. The test

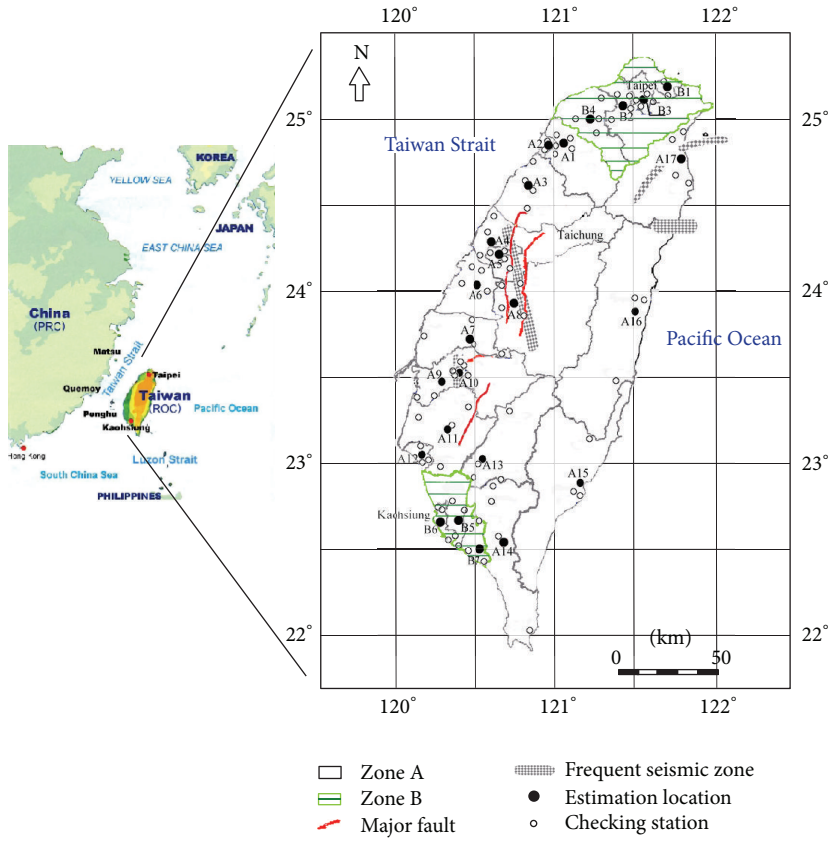


Figure 1: Sketch of the research area and seismic subdivision zone. (http://www.unc.edu/depts/diplomat/item/2010/0912/comm/norris_quemoymatsu.html).

result includes three parameters: the SPT-N value (number of times), V_s (S-wave, m/s), and V_p (P-wave, m/s). The present neural network model considers the standard penetration test value because it may be used to reflect the hardness of soil and the resistant of liquefaction. For a seismic body wave, the primary wave (sometimes referred to as the pressure wave) propagates very quickly and only lasts for a short time. Thus, it causes relatively insignificant structural damage and is not considered in this study. On the other hand, the shear wave, or secondary wave, propagates more slowly than the P-wave, and it may cause greater structural damage. Therefore, this study considers this factor in developing a neural network model.

3. Development and Performance of Neural Network Model

Neural network models have been applied to various engineering fields because they can be used to generate the required functions for parameter prediction and pattern recognition [30–33]. In this multilayered (input layer, hidden layer, and output layer) neural network, the output of each layer becomes the input of the next layer, and a specific learning law updates the weights of each layer connection in accordance with the errors from the network output. The equation for each layer may be written as

$$Y_j = \Phi\left(\sum W_{ij} X_i - \theta_j\right), \tag{1}$$

where Y_j is the output of neuron j, W_{ij} represents the connection weight from neuron i to neuron j, X_i is the input signal generated for neuron i, θ_j is the bias term associated with neuron j, and $\Phi(x) = 1/(1 + e^{-x})$ is the frequently used nonlinear activation function. More detailed descriptions of the algorithms and equations for neural networks can be found in the extensive literature on the subject, including the above cited references; thus, no further description will be given.

The performance of a neural network model can generally be evaluated by using the coefficient of correlation (R), defined as follows:

$$R = \frac{\sum_{i=1}^{m} (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{m} (x_i - \overline{x})^2 \sum_{i=1}^{m} (y_i - \overline{y})^2}}, \tag{2}$$

where x_i and $\overline{x}$ are the recorded value and its average value, respectively, y_i and $\overline{y}$ are the estimated value and its average value, respectively, and m denotes the number of data points in the analysis. In addition, an error evaluation function is required to calculate the difference between the actual record

Table 1: Typical seismic data sets at subdivision zone A7 (checking station CS25).

Year, Date and Time	Latitude (degree)	Longitude (degree)	Intensity	Magnitude (ML)	Depth (Di) (km)	Distance (De) (km)	PGA (V) (gal)	PGA (N-S) (gal)	PGA (E-W) (gal)
1996 0305 1452	23.93°	122.36°	2	6.4	6.0	192.4	2.0	6.3	6.4
1996 0305 1732	23.90°	122.30°	2	5.96	10.8	186.2	1.7	4.2	5.6
1996 0905 2342	22.00°	121.37°	2	7.07	14.8	218.2	2.8	7.4	6.3
1998 0717 0451	23.50°	120.66°	4	6.2	2.8	37.1	30.3	52.5	63.6
1999 0603 1611	24.40°	122.49°	2	6.18	61.7	215.2	4.6	7.5	6.7
1999 0920 1747	23.85°	120.82°	5	7.3	8.0	35.1	111.5	82.7	101.5
1999 0920 1751	24.09°	121.04°	2	5.97	6.2	66.2	5.0	7.5	7.3
1999 0920 1757	23.91°	121.04°	3	6.44	7.7	59.2	15.6	24.2	18.1
1999 0920 1803	23.80°	120.86°	5	6.6	9.8	39.0	60.8	68.0	97.4
1999 0920 1816	23.86°	121.04°	4	6.66	12.5	57.9	54.1	67.1	57.5
1999 0920 2146	23.58°	120.86°	4	6.59	8.6	44.9	51.0	50.2	47.4
1999 0921 0803	23.64°	120.63°	4	4.85	15.7	22.4	27.8	10.9	9.1
1999 0922 0049	23.76°	121.03°	4	6.2	17.4	56.5	25.8	26.1	35.9
1999 0925 2352	23.85°	121.00°	5	6.8	12.1	53.9	65.2	76.8	84.5
1999 1022 0218	23.52°	120.42°	5	6.4	16.6	30.8	104.9	167.7	87.9
1999 1022 0310	23.53°	120.43°	4	6	16.7	28.9	44.4	36.1	48.5
1999 1101 1753	23.36°	121.73°	3	6.9	31.3	135.9	9.7	18.4	16.4
2000 0610 1823	23.90°	121.11°	5	6.7	16.2	65.4	78.3	95.3	79.8
2000 0728 2028	23.41°	120.93°	3	6.1	7.3	62.6	5.0	12.1	10.5
2001 0613 1317	24.38°	122.61°	3	6.25	64.4	226.1	5.1	5.9	8.4
2001 0614 0235	24.42°	121.93°	3	6.3	17.3	163.0	3.8	7.2	9.0
2001 1218 0403	23.87°	122.65°	2	6.7	12.0	221.6	3.9	4.1	5.3
2002 0212 0327	23.74°	121.72°	3	6.2	30.0	127.0	9.4	19.3	18.9
2002 0331 0652	24.14°	122.19°	3	6.8	13.8	178.6	10.2	17.0	18.5
2002 0916 0003	25.10°	122.39°	2	6.8	175.7	242.0	2.1	6.7	6.3
2003 0610 0840	23.50°	121.70°	3	6.48	32.3	128.6	13.0	22.1	20.2
2004 0519 0704	22.71°	121.37°	3	6.03	27.1	150.2	4.4	7.9	8.4
2005 0121 1428	24.56°	122.53°	2	5.94	92.1	225.1	1.3	4.2	2.4
2006 0309 1207	23.64°	120.56°	5	5.09	9.9	18.1	79.74	102.80	132.70
2006 0401 1802	22.88°	121.08°	2	6.23	7.2	117.8	4.10	5.52	6.34
2006 0405 0330	24.49°	122.76°	2	5.8	99.5	244.3	0.98	1.84	3.78
2006 0416 0640	22.86°	121.30°	2	6.04	17.9	133.6	2.14	4.46	7.10
2006 0728 1540	23.97°	122.66°	2	6.02	28.0	222.8	1.36	2.58	4.06
2006 1226 2026	21.69°	120.56°	3	6.96	44.1	233.1	9.08	15.84	15.58
2006 1226 2034	21.97°	120.42°	4	6.99	50.2	201.7	31.32	28.30	18.70
2007 0125 1859	22.63°	122.03°	2	6.24	25.8	204.0	1.30	4.48	3.84
2007 0723 2140	23.72°	121.64°	2	5.77	38.63	118.27	6.34	7.02	7.64
2007 0809 0855	22.65°	121.08°	2	5.68	5.51	140.79	0.88	2.04	4.48
2007 0907 0151	24.28°	122.25°	3	6.63	54.01	188.53	9.34	15.96	20.06
2008 0305 0131	23.21°	120.70°	2	5.22	11.32	67.93	5.92	7.24	4.78
2008 1202 1116	23.34°	121.49°	2	5.68	31.67	113.57	2.66	4.18	4.7
2008 1208 0518	23.85°	122.20°	3	5.88	35.05	174.37	1.86	4.5	9.9
2009 0417 2037	23.92°	121.68°	2	5.33	43.43	122.56	2.26	3.72	7.22
2009 0720 0900	23.69°	120.96°	3	5.35	14.29	49.3	8.78	5.04	5.78
2009 0726 1410	23.43°	121.32°	2	5.38	12.52	93.73	2.02	3.66	2.98
2009 1004 0136	23.65°	121.58°	3	6.09	29.15	112.5	6.24	11.4	9.54
2010 1108 2101	23.21°	120.40°	3	5.16	17.46	65.05	7.14	9.76	13
2010 1121 2031	23.85°	121.69°	4	6.14	46.87	122.39	9.16	15.84	26.06
2011 0430 1635	24.65°	121.81°	2	5.81	75.02	164.74	1.36	2.14	3.94

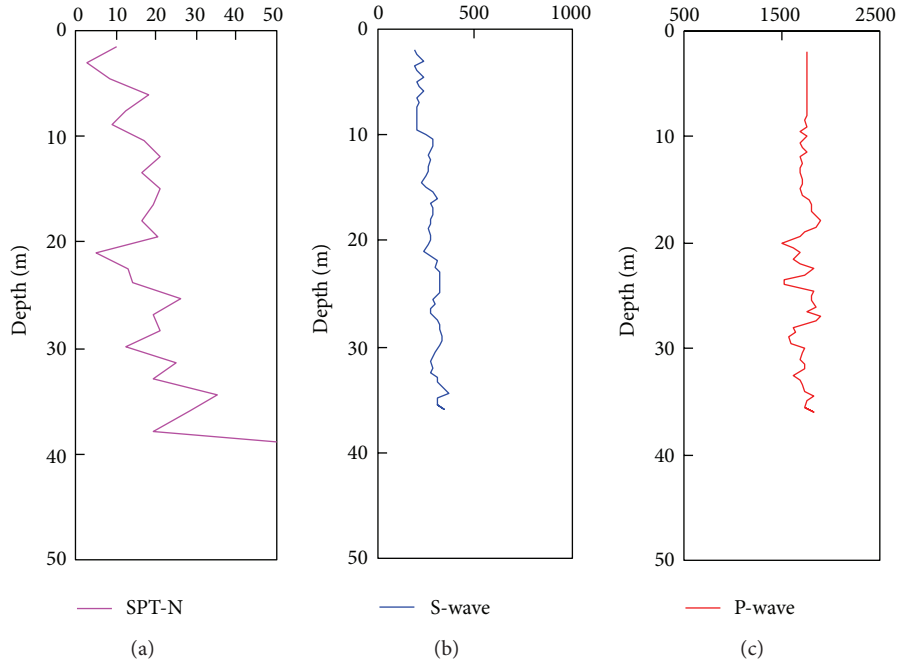


Figure 2: A typical example of geological test result (checking station CS25, A7).

Table 2: Performance (R^2) of NN models with different neurons in the hidden layer.

Models\neurons	1	2	3	4	5	6	7
NN1	0.770	0.736	0.843	0.820	0.852	0.733	0.790
NN2	0.873	0.853	**0.900**	0.854	0.759	0.713	0.668
NN3	0.873	0.834	**0.918**	0.835	0.812	0.745	0.707
NN4	0.821	0.764	0.811	0.777	0.789	0.753	0.743

Table 3: Comparison of NN models in different computation stages (R^2).

NN Model		NN2			NN3	
Direction	V	N-S	E-W	V	N-S	E-W
Training	0.905	0.905	0.901	0.903	0.911	0.893
Validation	0.898	0.931	0.909	0.873	0.894	0.867
Testing	0.938	0.947	0.942	0.916	0.921	0.928
Average		0.920			0.901	

values and neural network estimations. This study uses the root-mean-square error (RMSE), defined as

$$\text{RMSE} = \sqrt{\frac{\sum_n^N (T_n - Y_n)^2}{N}}, \tag{3}$$

where N is the number of learning cases, T_n is the target value for case n, and Y_n is the output value for case n. This study uses these equations to evaluate the performance of the proposed neural network model and check its effectiveness.

This study considers four neural network models of different input parameters with different neurons in the hidden layer. Figure 3 shows the structure of these models. The data sets of seismic parameters and soil test results require a normalization procedure before neural network computation. The data sets are then divided into three groups randomly, with 70% used for training, 20% used for validation, and 10% used to test the neural network models. To prevent performing too much work in computation for choosing the number of neurons in the hidden layer, this study initially takes three randomly subdivision zones to check the effect of neuron numbers in the hidden layer: northern part (Taipei city, B3), central part (Taichung city, A4), and southern part (Kaohsiung city, B5). Table 2 shows the averaged calculation results, indicating that using three neurons in the hidden layer can achieve relatively better coefficients of correlation in these comparison cases, particularly for NN2 and NN3 models. Though the result shown here is only for the chosen three stations, this should provide a basic check, and further details for all checking stations will be discussed later.

For error analysis, this study randomly chooses four checking stations from subdivision zones B3, A4, and B5. Figure 4 shows the convergent tendency in neural network computation, indicating that the root-mean-square errors in three directions are reasonable for these example cases. The errors ranged between 10^{-2} and 10^{-5} and should have a similar tendency for other checking stations. Thus, the present setup of 1000 epochs using the neural network toolbox in MATLAB should be sufficient to cover all checking stations and achieve acceptable accuracy.

Now by taking data sets from all checking stations, the computational result of NN2 and NN3 models, with three neurons in the hidden layer, is shown in Table 3. Training, validation, and testing stages show that the averaged square

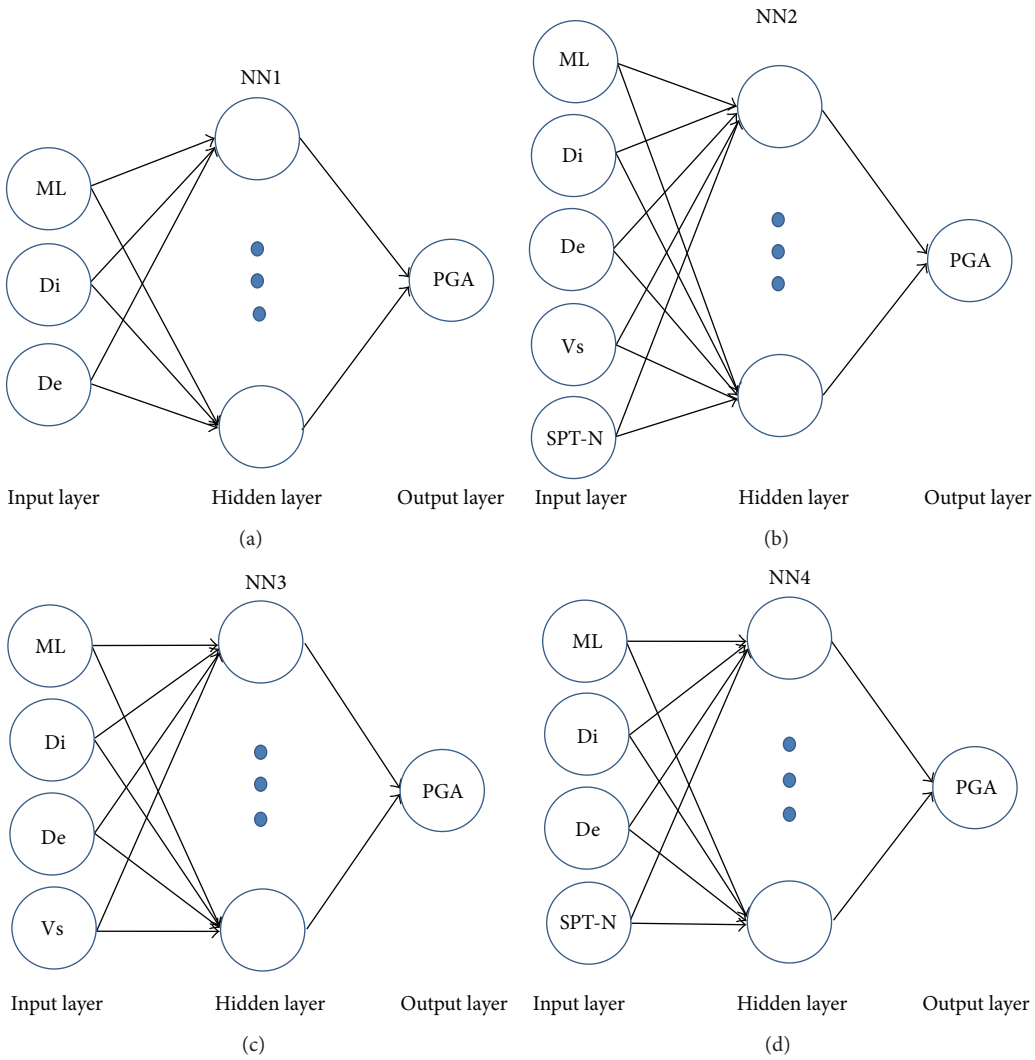


Figure 3: Four neural network models with various input parameters and neurons in the hidden layer.

of correlation coefficient of the NN2 model ($R^2 = 0.920$) is higher than that of the NN3 model ($R^2 = 0.901$). In other words, the NN2 model, which uses three seismic parameters (ML, Di, and De) and two soil test results (V_s, SPT-N) in the input layer, can obtain the best PGA estimation among the cases in this study. Therefore, this neural network model is employed for further PGA predictions in all 24 subdivision zones, and the following section discusses the calculation results.

4. Evaluation of Seismic Design Value in Subdivision Zone

The performance analysis above indicates that the neural network model NN2 with five inputting parameters (i.e., local magnitude, epicentral distance, epicenter depth, shear wave velocity, and standard penetration test value) offers reliable and generalizable results in predicting the PGA. To further check this model, Figure 5 shows the relationship between the actual seismic record and neural network estimation for all three directions and for all data sets from 86 checking stations. Note that a total of 3414 data points are plotted in the figure for all directions. The R^2 value ranges from 0.772 up to 0.8209, indicating a high correlation between the two data sets. The root-mean-square error is on the order of 10^{-2}, which is sufficiently small to demonstrate the ability of developing neural network. These results provide confidence for predicting the PGA in unchecked sites.

It is possible to interpolate peak ground acceleration from discrete array stations for generating a better shaking map after an earthquake [34]. In this study, calculating the PGA in the 24 subdivision zones requires a spatial relationship to determine a new location to represent each subdivision zone. This can be done by using coordinates for checking stations near each of the subdivision zones. A straightforward method of calculating the PGA in each new site is to distribute neural network estimations from nearby checking stations. A weighting factor is assigned to each checking station in accordance with the distance between two locations. The

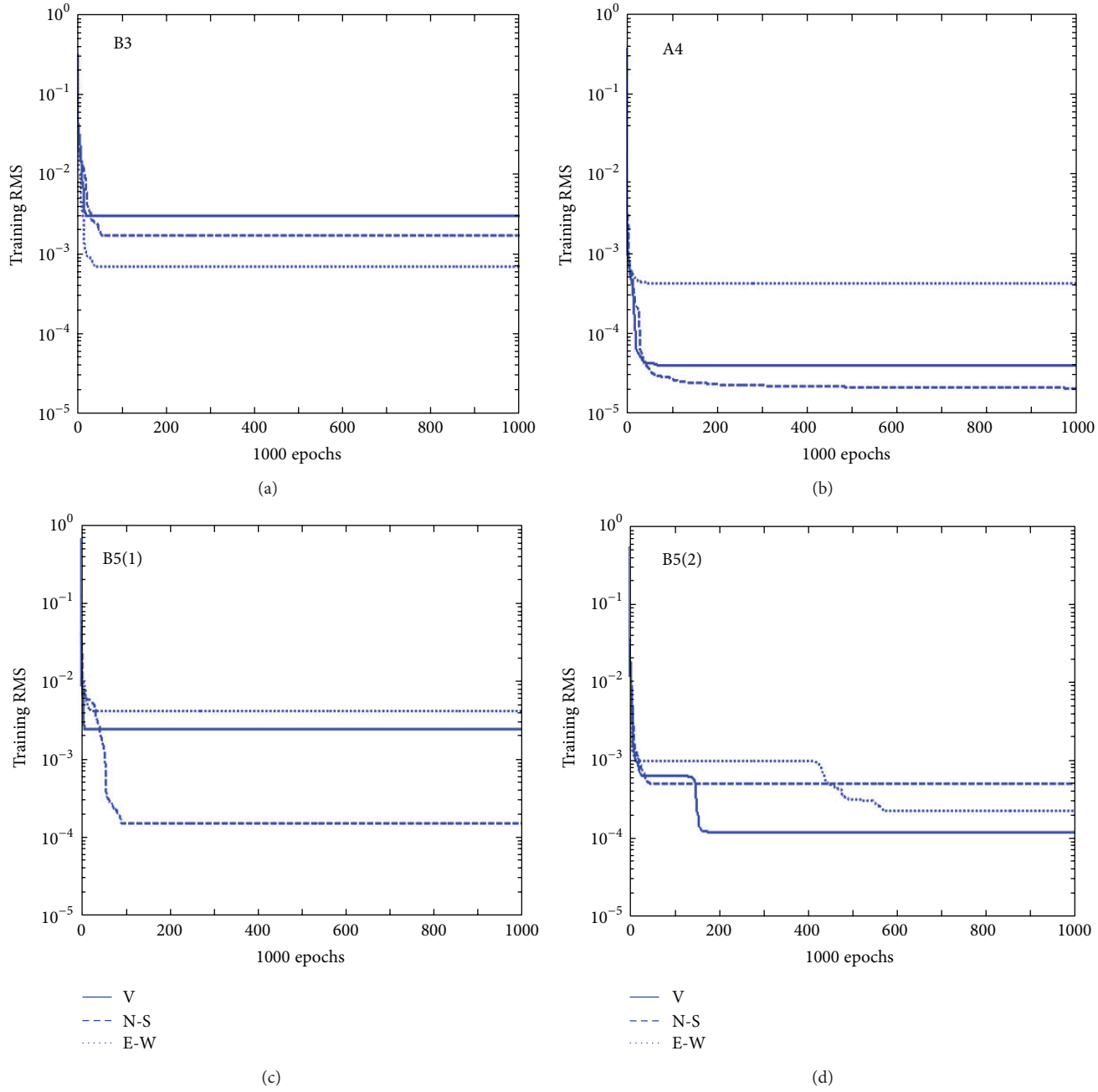


Figure 4: Examples of error convergent tendency in neural computing (RMSE versus epochs).

weight (W_i) of each checking station to the unchecked site can be defined as follows:

$$W_i = \frac{\left(\sum_{j=1}^{n} d_j\right)/d_i}{\sum_{k=1}^{n}\left[\left(\sum_{j=1}^{n} d_j\right)/d_k\right]}; \quad i = 1, 2, 3, \ldots, n, \qquad (4)$$

where d_i, d_j, and d_k are the distances between the unchecked site and known checking stations. The estimation result for the new location can be obtained after summing the neural network estimation for all checking stations around this new location. This simple method is denoted as "Model 1" in this study.

Alternatively, a better way to estimate the PGA at an unchecked site is to take a new set of the seismic data (same local magnitude and epicenter depth, but new epicentral distance for each of the seismic records) and a new set of geological conditions (weight-based soil test result) from known checking stations nearby. Then, insert the data set in a neural network model developed for each known checking station. By summing the results with weighting factors in accordance with the distances between the unchecked sites to the known stations, the final estimation is obtained for the unchecked site. This method is denoted as "Model 2."

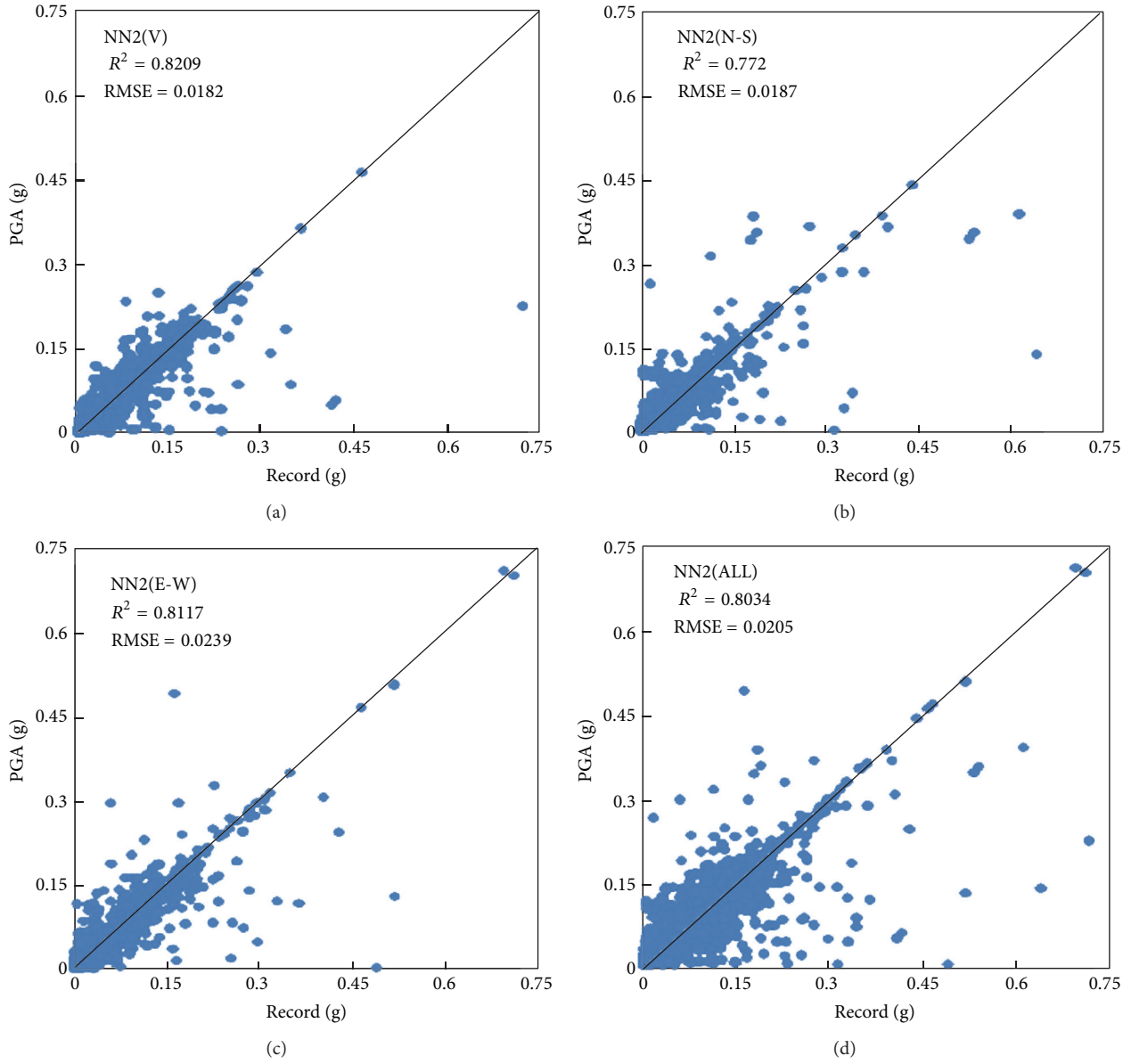


Figure 5: Relationship between actual seismic record and neural network estimation.

The descriptions may be written as the following equation [35, 36]:

$$\mathrm{NN}_{\mathrm{ucs}} = \sum_{i=1}^{n} (\mathrm{NN2}_i)\, W_i, \tag{5}$$

where $\mathrm{NN}_{\mathrm{ucs}}$ is the final PGA estimation for the unchecked site, $\mathrm{NN2}_i$ is the estimation using preferred neural network model as discussed in the previous section for each checking station, n is the number of checking stations, and W_i is the same as defined in (4).

Figure 6 shows the PGA prediction for all 24 subdivision zones in different directions for both models. The vertical PGA is smaller than the average of the other two directions. These calculation results do not differ significantly between the two models, except at subdivision zones A8, A13, and A15, and particularly in vertical direction. The main difference between Model 1 and Model 2 is that Model 1 uses the estimation results from nearby checking stations directly, whereas Model 2 considers a new epicentral distance to obtain the PGA result for each subdivision zone. Therefore, the epicentral distance may be varied by subdivision zone area, graphic condition, and mean location of checking stations. This can cause somewhat different PGA predictions in the two models. In general, Model 2 is more reasonable and preferable because it has a spatial relationship to the proposed neural network model.

To check reliability of the above estimation result, Figure 7 shows a comparison of the neural network-predicted PGA and the result of available microtremor measurements

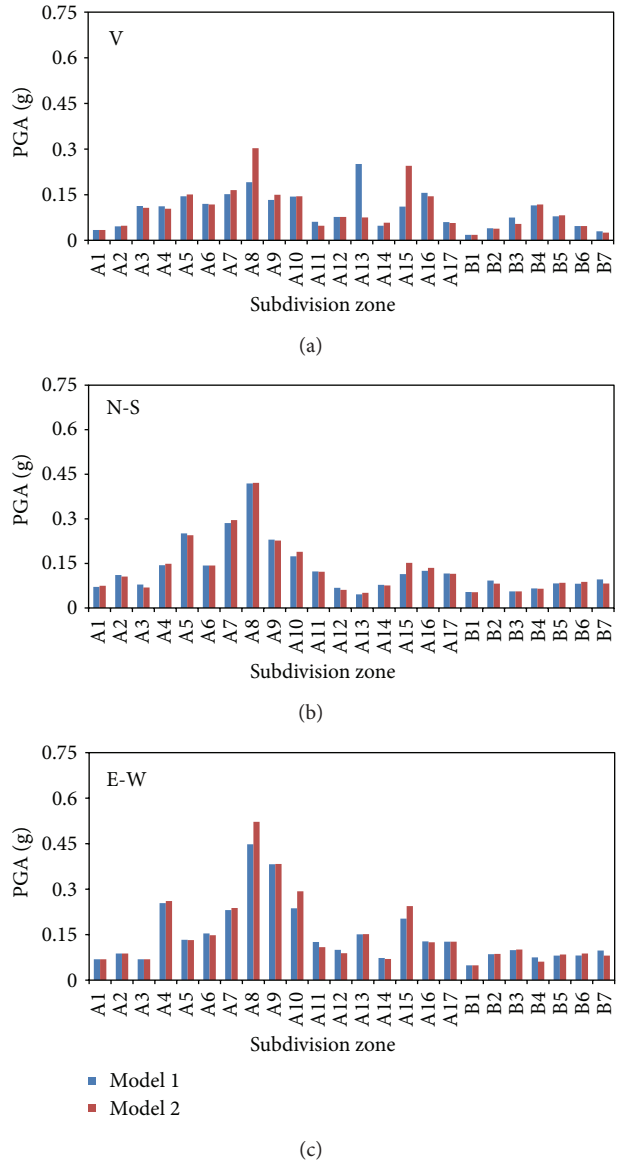


Figure 6: PGA predictions in the 24 seismic subdivision zones from two models.

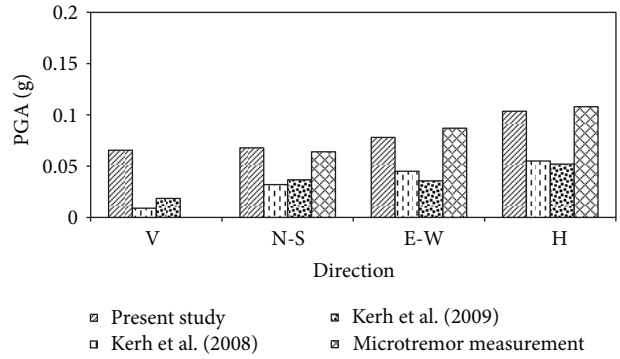


Figure 7: Predicted PGA results versus PGA results from microtremor measurements.

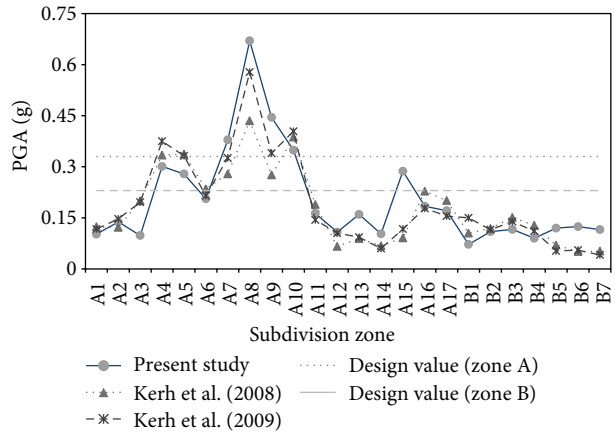
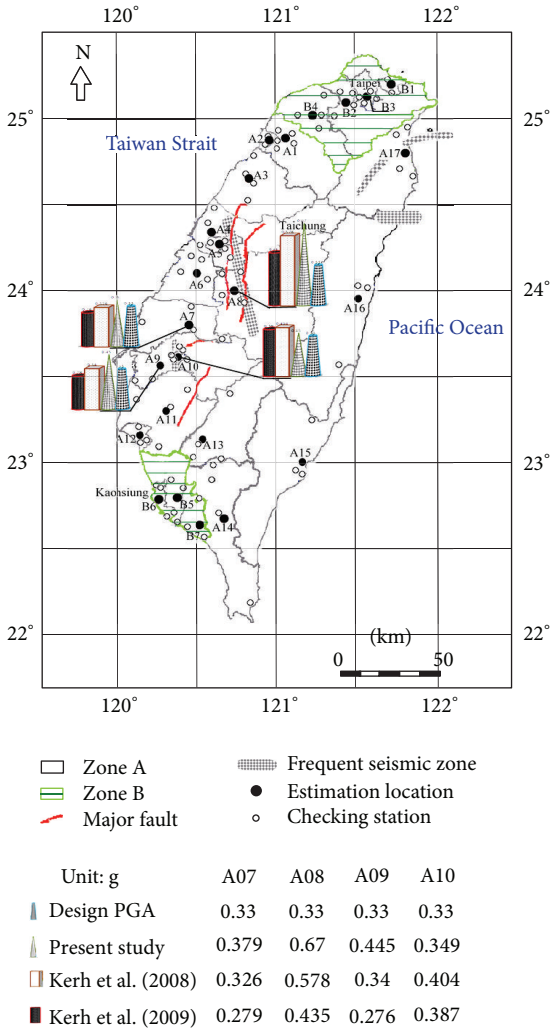


Figure 8: Comparison of predicted horizontal PGA and seismic design values.

[37]. Note that measurement result in the vertical direction is not available from the previous literature. The proposed neural network model, which considers seismic parameters and site geological conditions, achieves better prediction results than previous studies. This may be because the present study uses more updated seismic records to develop the neural network model. The present study also uses soil test results as the input parameters, which may be more related to onsite microtremor measurements. Thus, the results of this study can increase the confidence of predicting the PGA at unchecked sites.

Figure 8 shows the horizontal PGA calculated from N-S and E-W directions for each subdivision zone. This figure shows data for four locations: Yunlin county (A7), Nantou county (A8), Chiayi county (A9), and Chiayi city (A10). These locations exhibit a higher neural network estimation than that of the seismic design value in Zone A (0.33 g). These locations are somewhat different from previous studies. To help display the results more clearly, Figure 9 shows that these four identified potential hazardous subdivision zones are located in the Central and Southern parts of Taiwan. The predictions suggest that these areas deserve more study to prevent unnecessary loss because of unpredictable strong ground motions. For Zone B, the neural network PGA predictions in the seven subdivision zones all comply with design standard (0.23 g); that is, no predicted PGA exceeds the design value.

This study uses checking stations and soil test results taken from the same places. In addition, more recent earthquake records (up to 2011) are included to develop the proposed neural network model. Therefore, the results obtained in this study should be more reliable than those of the previous literatures. Now, by taking all neural network estimations for each of the 24 subdivision zones, and by defining the distance between hypocenter of an earthquake to the checking station as the focal distance (which represents two important earthquake parameters, i.e., the focal depth and the epicentral distance). Besides, a local magnitude of earthquake may be directly related to the element of peak



Unit: g	A07	A08	A09	A10
Design PGA	0.33	0.33	0.33	0.33
Present study	0.379	0.67	0.445	0.349
Kerh et al. (2008)	0.326	0.578	0.34	0.404
Kerh et al. (2009)	0.279	0.435	0.276	0.387

FIGURE 9: Location of identified potential hazardous subdivision zones.

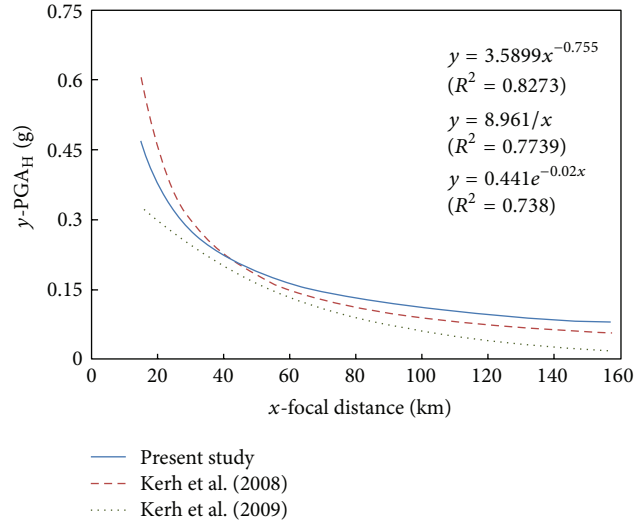


FIGURE 10: Curve fitting model for horizontal PGA and focal distance.

ground acceleration. Hence, a derived result with one single variable for prediction is possible and shown in Figure 10. From the relationship between horizontal PGA and focal distance for all subdivision zones, this study develops the equation $\mathrm{PGA}_H = 3.5899\mathrm{D}_f{}^{-0.755}$ with a high square value of correlation coefficient $R^2 = 0.8273$ using a curve fitting method. This mathematical equation is more reliable than those in previous studies and can be used to represent seismic characteristics in Taiwan region more reasonably.

5. Conclusion

Previous studies have shown that using three seismic parameters (i.e., local magnitude on Richter scale, epicentral distance, and epicenter depth) in the input layer of a neural network model can efficiently predict PGA, which is the key parameter for evaluating earthquake response in a construction site. However, geological conditions may have an influence on earthquake wave propagation, causing significant variation in the level of structural damage. Therefore, it is worthwhile to include suitable soil test results as inputs when developing a neural network model.

In addition to these three seismic parameters, this study adopts two soil test results (i.e., shear wave velocity and standard penetration test value) to develop a neural network model for 86 checking stations across the island of Taiwan. Results show that the model with three neurons in the hidden layer achieved relatively better performance based on the correlation of coefficient and the mean square error. This study also develops a simple distributing method and a weight-based neural network model to predict the PGA in 24 subdivision zones.

These results show that four locations have higher PGAs than that of the seismic design value and thus require more caution as potentially hazardous areas. This study uses a curve fitting method to develop a mathematical equation $\mathrm{PGA}_H = 3.5899\mathrm{D}_f{}^{-0.755}$ with a sufficiently high square value of correlation coefficient $R^2 = 0.8273$. This equation might represent seismic characteristics in Taiwan region more reliably and reasonably than previous similar equations. The geological conditions of an unchecked site might not be suitable for characterizing nearby checking stations. However, the method presented in this study provides a good way to model this type of nonlinear seismic problem and might be applicable to other areas of interest around the world.

Acknowledgments

The authors gratefully acknowledge the Central Weather Bureau Seismological Center and the National Center for Research on Earthquake Engineering of Taiwan for providing seismic records and geological surveys, respectively. The financial support of the National Science Council under the Project no. NSC101-2221-020-030 is also greatly appreciated.

In addition, the editing of the work with improving of English by Ryan Wallace is also acknowledged.

References

[1] Central Weather Bureau, "What is the zone division standard of anti-earthquake for general building in Taiwan?" Hundred Questions of Earthquake, 2005, http://scman.cwb.gov.tw/eqv5/eq100/100/058.htm.

[2] Construction and Planning Agency, "Revision of building anti-earthquake design code and explanation," Ministry of the Interior, 2006, http://www.cpami.gov.tw/web/index.php?option=com_content&task=view&id=976&Itemid=95.

[3] S. E. Hough, T. Taniguchi, and J. R. Altidor, "Estimation of peak ground acceleration from horizontal rigid body displacement: a case study in Port-au-Prince, Haiti," *Bulletin of the Seismological Society of America*, vol. 102, no. 6, pp. 2704–2713, 2012.

[4] H. Adeli and A. Panakkat, "A probabilistic neural network for earthquake magnitude prediction," *Neural Networks*, vol. 22, no. 7, pp. 1018–1024, 2009.

[5] K. V. Yuen, *Bayesian Methods for Structural Dynamics and Civil Engineering*, John Wiley & Sons, 2010.

[6] K. V. Yuen and H. Q. Mu, "Peak ground acceleration estimation by linear and nonlinear models with reduced order Monte Carlo simulation," *Computer-Aided Civil and Infrastructure Engineering*, vol. 26, no. 1, pp. 30–47, 2011.

[7] E. I. Alves, "Earthquake forecasting using neural networks: results and future work," *Nonlinear Dynamics*, vol. 44, no. 1–4, pp. 341–349, 2006.

[8] G. Ghodrati Amiri and A. Bagheri, "Application of wavelet multiresolution analysis and artificial intelligence for generation of artificial earthquake accelerograms," *Structural Engineering and Mechanics*, vol. 28, no. 2, pp. 153–166, 2008.

[9] C. R. Arjun and A. Kumar, "Neural network estimation of duration of strong ground motion using Japanese earthquake records," *Soil Dynamics and Earthquake Engineering*, vol. 31, no. 7, pp. 866–872, 2011.

[10] M. H. Baziar and A. Ghorbani, "Evaluation of lateral spreading using artificial neural networks," *Soil Dynamics and Earthquake Engineering*, vol. 25, no. 1, pp. 1–9, 2005.

[11] H. Dai and C. MacBeth, "Application of back-propagation neural networks to identification of seismic arrival types," *Physics of the Earth and Planetary Interiors*, vol. 101, no. 3-4, pp. 177–188, 1997.

[12] S. R. García, M. P. Romo, and J. M. Mayoral, "Estimation of peak ground accelerations for Mexican subduction zone earthquakes using neural networks," *Geofisica Internacional*, vol. 46, no. 1, pp. 51–63, 2007.

[13] S. C. Lee and S. W. Han, "Neural-network-based models for generating artificial earthquakes and response spectra," *Computers and Structures*, vol. 80, no. 20-21, pp. 1627–1638, 2002.

[14] C. C. J. Lin and J. Ghaboussi, "Generating multiple spectrum compatible accelerograms using stochastic neural networks," *Earthquake Engineering and Structural Dynamics*, vol. 30, no. 7, pp. 1021–1042, 2001.

[15] A. Panakkat and H. Adeli, "Neural network models for earthquake magnitude prediction using multiple seismicity indicators," *International Journal of Neural Systems*, vol. 17, no. 1, pp. 13–33, 2007.

[16] A. Panakkat and H. Adeli, "Recent efforts in earthquake prediction (1990-2007)," *Natural Hazards Review*, vol. 9, no. 2, pp. 70–80, 2008.

[17] A. Panakkat and H. Adeli, "Recurrent neural network for approximate earthquake time and location prediction using multiple seismicity indicators," *Computer-Aided Civil and Infrastructure Engineering*, vol. 24, no. 4, pp. 280–292, 2009.

[18] G. A. Tselentis and L. Vladutu, "An attempt to model the relationship between MMI attenuation and engineering ground-motion parameters using artificial neural networks and genetic algorithms," *Natural Hazards and Earth System Science*, vol. 10, no. 12, pp. 2527–2537, 2010.

[19] T. Kerh, J. S. Lai, D. Gunaratnam, and R. Saunders, "Evaluation of seismic design values in the Taiwan building code by using artificial neural network," *Computer Modeling in Engineering and Sciences*, vol. 26, no. 1, pp. 1–12, 2008.

[20] T. Kerh and D. Chu, "Neural networks approach and microtremor measurements in estimating peak ground acceleration due to strong motion," *Advances in Engineering Software*, vol. 33, no. 11-12, pp. 733–742, 2002.

[21] T. Kerh, D. Gunaratnam, and Y. Chan, "Neural computing with genetic algorithm in evaluating potentially hazardous metropolitan areas result from earthquake," *Neural Computing and Applications*, vol. 19, no. 4, pp. 521–529, 2010.

[22] T. Kerh, Y. Chan, and D. Gunaratnam, "Treatment and assessment of nonlinear seismic data by a genetic algorithm based neural network model," *International Journal of Nonlinear Sciences and Numerical Simulation*, vol. 10, no. 1, pp. 45–56, 2009.

[23] C. T. Huang, "A study on the earthquake potential damage and evaluation method—earthquake spectral research of considering regional site effect," Project of National Science Council NSC 90-2625-Z-011-001, 2002.

[24] G. L. Wun, W. Y. Gien, and Y. W. Chang, "Strong ground motion site effect in Taiwan area," Earthquake Technology Report MOTC-CWB-93-E-09, Central Weather Bureau, 2004.

[25] K. Günaydın and A. Günaydın, "Peak ground acceleration prediction by artificial neural networks for northwestern Turkey," *Mathematical Problems in Engineering*, vol. 2008, Article ID 919420, 20 pages, 2008.

[26] B. Derras, "Peak ground acceleration prediction using artificial neural networks approach: application to the Kik-Net data," *International Review of Civil Engineering*, vol. 1, no. 3, pp. 243–252, 2010.

[27] Central Geological Survey, "Taiwan fault distribution map," 2011, Ministry of Economic Affairs, http://fault.moeacgs.gov.tw/TaiwanFaults/Default.aspx?LFun=2.

[28] H. Tsai and G. F. Yang, *Faults and Earthquakes of Taiwan*, Walkers Cultural Enterprise Ltd, Taipei, Taiwan, 2004.

[29] Encyclopedia of Taiwan, "Geology," 2012, http://taiwanpedia.culture.tw/web/index.

[30] K. Kuźniar, E. Maciag, and Z. Waszczyszyn, "Computation of response spectra from mining tremors using neural networks," *Soil Dynamics and Earthquake Engineering*, vol. 25, no. 4, pp. 331–339, 2005.

[31] Y. Lu, "Underground blast induced ground shock and its modelling using artificial neural network," *Computers and Geotechnics*, vol. 32, no. 3, pp. 164–178, 2005.

[32] S. Mandal, S. Rao, and D. H. Raju, "Ocean wave parameters estimation using backpropagation neural networks," *Marine Structures*, vol. 18, no. 3, pp. 301–318, 2005.

[33] F. Sarghini, G. Felice, and S. Santini, "Neural networks based subgrid scale modeling in large eddy simulations," *Computers and Fluids*, vol. 32, no. 1, pp. 97–108, 2003.

[34] E. Harmandar, E. Cakti, and M. Erdik, "A method for spatial estimation of peak ground acceleration in dense arrays," *Geophysical Journal International*, vol. 191, no. 3, pp. 1272–1284, 2012.

[35] T. Kerh, C. H. Huang, and D. Gunaratnam, "Neural network approach for analyzing seismic data to identify potentially hazardous bridges," *Mathematical Problems in Engineering*, vol. 2011, Article ID 464353, 15 pages, 2011.

[36] T. Kerh, T. Ku, and D. Gunaratnam, "Comparative evaluations of the seismic key parameter by artificial neural network model and ambient vibration survey," *Disaster Advances*, vol. 4, no. 2, pp. 5–12, 2011.

[37] T. Kerh and S. B. Ting, "Neural network estimation of ground peak acceleration at stations along Taiwan high-speed rail system," *Engineering Applications of Artificial Intelligence*, vol. 18, no. 7, pp. 857–866, 2005.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 923036, 13 pages
http://dx.doi.org/10.1155/2013/923036

Research Article

Mathematical Model Based on BP Neural Network Algorithm for the Deflection Identification of Storage Tank and Calibration of Tank Capacity Chart

Caihong Li,[1] Yali Yuan,[1] Lulu Song,[2] Yunjian Tan,[1] and Guochen Wang[3]

[1] *School of Information Science and Engineering, Lanzhou University, Lanzhou, Gansu 730000, China*
[2] *College of Tourism, Hainan University, Haikou, Hainan 570228, China*
[3] *Hefei Rongshida Sanyo Electric Co. Ltd., Hefei, Anhui 230061, China*

Correspondence should be addressed to Yali Yuan; yuanyali_first@foxmail.com

Received 28 February 2013; Accepted 22 April 2013

Academic Editor: Fuding Xie

The tank capacity chart calibration problem of two oil tanks with deflection was studied, one of which is an elliptical cylinder storage tank with two truncated ends and another is a cylinder storage tank with two spherical crowns. Firstly, the function relation between oil reserve and oil height based on the integral method was precisely deduced, when the storage tank has longitudinal inclination but has no deflection. Secondly, the nonlinear optimization model which has both longitudinal inclination parameter α and lateral deflection parameter β was constructed, using cut-complement method and approximate treatment method. Then the deflection tank capacity chart calibration with a 10 cm oil level height interval was worked out. Lastly, the tank capacity chart was corrected by BP neural network algorithm and got proportional error of theoretical and experimental measurements ranges from 0% to 0.00015%. Experimental results demonstrated that the proposed method has better performance in terms of tank capacity chart calibration accuracy compared with other existing approaches and has a strongly practical significance.

1. Introduction

Tanks for storing oil have been in existence for almost one hundred years. Many of the underground storage tanks are horizontal tanks; they are mainly divided into square, cylindrical, and elliptic cylindrical and their roofs can be divided into flat top, conic top, ball lacunarity, and so on [1, 2]. In this paper, cylindical storage tank with two spherical crowns was primarily discussed, which is more practical. The oil reserve measurement of storage tank is a challenging problem, especially after a period of time, due to foundation deformation and other reasons; oil storage tanks tend to be vertical or horizontal displacement, resulting in inaccurate tank volume tables.

Although some install the automated measurement system, the measurement accuracy is not high. And the price of the imported high-precision liquid level instrument is too high. Therefore, on the basis of the situation and the development trend of the current domestic and foreign oil tank liquid level measurement technology, developing a kind of liquid level measurement technology which is suitable for China's national conditions is very important [3]. People usually use flow meter and oil level gauge to measure input or output oil, oil height of the tank, and other data and to get the changeable relation of oil height and oil volume, by means of the precalibration tank capacity table (the corresponding relationship between oil height and oil volume of storage tank), so as to determine whether to add oil or not [4–8].

Since the early 1870s, some scholars have already tried to solve this problem [9, 10]. In particular after 2010s, many researchers have made much study as to how to improve the accuracy of calibration of tank capacity chart and proposed many improved methods. Most of them adopted pure integral and infinitesimal method to handle tank issues [11–13] while

some of them used method of the minimum squares [14, 15]. However, for the methods of error correction, few of the papers use methods to correct the result of the calibration of tank capacity chart [16]. And few papers have been presented on handling tank issues by BP neural network [17]. Nevertheless, many issues and problems about calibration have been addressed and resolved and got a good result when using BP neural network [18–20].

The error between theoretical oil reserve and actual oil reserve results from two main reasons. One of the reasons is the irregular geometry of the actual storage tank and another is the volatilization of the oil, the thickness, and the capillary absorption phenomenon of storage tank, which leads to a certain deviation between the theoretical oil reserve and actual ones. As the rule of this kind deviation is relatively fuzzy, in order to further reduce error and improve the accuracy of the calibration, the BP neural network, a method with self-learning ability, is adopted in this paper to revise the calibration value [21].

Artificial neural networks (ANNS) are powerful tools for prediction of nonlinearities. These mathematical models comprise individual processing units called neurons that resemble neural activity [22]. After the first simple neural network was developed by McCulloch and Pitts in 1943 [23], many types of ANN have been proposed. BP neural network simulates the human nervous system structure and the neural network model with multilayer perceptron is the most mature, widely used model among ANN.

Currently the error back-propagation (BP) neural network is the most widely used, which consists of three layers, namely, input, hidden, and output layers. One artificial neuron is simple, but a lot of artificial neurons can compose complicated neural network which can achieve highly nonlinear mapping relation between the input and output through the interaction of artificial neurons and realize the information processing and storage [24]. Due to its highly parallel structure, high-speed self-leaning ability, self-adaptable processing ability, arbitrary function mapping ability, powerful pattern classification, and pattern recognition capabilities for modeling complex nonlinear systems [25], BP neural network algorithm studies show promising results in calibration and is used in this study.

The paper is mainly organized into four sections. Section 2 describes the model establishment and solution of small elliptic storage tank with deflection identification and calibration of tank capacity chart. It is divided into three parts as follows: Section 2.1 mathematical models of the relation between oil reserve and oil height of the small nondeflection elliptic storage tank, Section 2.2 model 2 for tank capacity chart calibration problem of small elliptic storage tank at an inclination angle α of 4.1°, and Section 2.3 correction model of calibration based upon BP neural network. Section 3 describes the establishment and solution of the model with deflection identification and calibration of tank capacity chart of actual storage tank. It also includes three parts: Section 3.1 model 3 of actual storage tank with longitudinal inclination and lateral deflection, Section 3.2 the determination of the deflection parameter, and Section 3.3 model solving of actual storage tank and calibration of storage tank chart. Finally, some concluding remarks are drawn in Section 4.

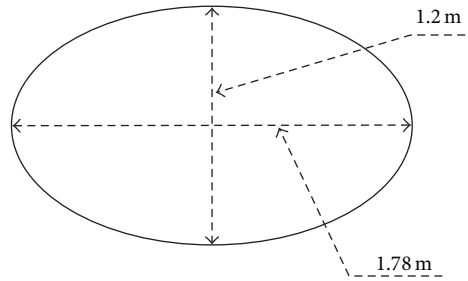


Figure 1: Cross-section schematic of small elliptic storage tank.

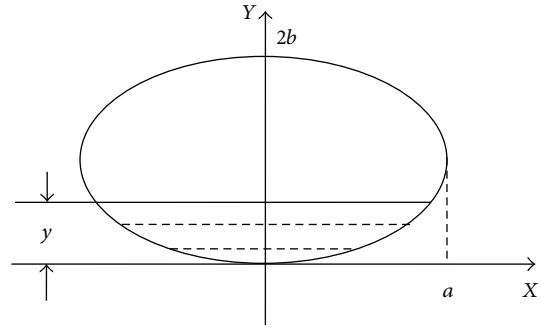


Figure 2: Cross-section diagram of the small elliptical tank without deflection.

2. Model Establishment and Solution of Small Elliptic Storage Tank with Deflection Identification and Calibration of Tank Capacity Chart

2.1. Mathematical Models of the Relation between Oil Reserve and Oil Height of the Small Nondeflection Elliptic Storage Tank. According to the cross-section diagram of the small elliptical storage tank as shown in Figure 1, we set up a coordinate system as shown in Figure 2. Make the profile nadir of the storage tank for origin, the high for y shaft, and the basal level tangent for x shaft.

The cross-section oil surface area of the small elliptical tank can be calculated according to the application of definite integration:

$$s_1(h_1) = 2\int_0^{h_1} x\,dy. \tag{1}$$

According to the elliptic equation $(x^2/a^2) + (y-b)^2/b^2 = 1$, substitute $x = a\sqrt{1-(y-b)^2/b^2}$ into formula (1) and integral, which can obtain the following:

$$s_1(h_1) = \frac{ab}{2}\pi - a\sqrt{k_h} + \frac{a}{b}h_1\sqrt{k_h} + ab\arcsin\left(-1+\frac{h_1}{b}\right), \tag{2}$$

where k_h is $-h_1(h_1-2b)$.

Table 1: Testing data and result of model one.

Site 1					Site 2				
OH/mm	AOR/L	TOR/L	IOR/L	IER	OH/mm	AOR/L	TOR/L	IOR/L	IER
159.02	312.00	322.88	313.46	0.47%	486.89	1512.00	1564.74	1511.08	0.06%
176.14	362.00	374.63	362.90	0.25%	498.95	1562.00	1616.49	1561.20	0.05%
192.59	412.00	426.36	412.41	0.10%	510.97	1612.00	1668.24	1611.33	0.04%
208.50	462.00	478.13	462.03	0.01%	522.95	1662.00	1719.98	1661.46	0.03%
223.93	512.00	529.85	511.67	0.06%	534.90	1712.00	1771.73	1711.59	0.02%
238.97	562.00	3581.61	561.39	0.11%	546.82	1762.00	1823.46	1761.71	0.02%
253.66	612.00	633.35	611.16	0.14%	558.72	1812.00	1875.19	1811.84	0.01%
268.04	662.00	685.08	660.95	0.16%	570.61	1862.00	1926.95	1862.00	0.00%
282.16	712.00	736.85	710.81	0.17%	582.48	1912.00	1978.68	1912.12	0.01%
296.03	762.00	788.58	760.67	0.17%	594.35	1962.00	2030.43	1962.27	0.01%
309.69	812.00	840.33	810.58	0.18%	606.22	2012.00	2082.20	2012.44	0.02%
323.15	862.00	892.06	860.49	0.18%	618.09	2062.00	2133.95	2062.59	0.03%
336.44	912.00	943.80	910.44	0.17%	629.96	2112.00	2185.67	2112.71	0.03%
349.57	962.00	995.54	960.41	0.17%	641.85	2162.00	2237.43	2162.86	0.04%
362.56	1012.00	1047.30	1010.41	0.16%	653.75	2212.00	2289.16	2212.99	0.04%
375.42	1062.00	1099.05	1060.43	0.15%	665.67	2262.00	2340.89	2263.10	0.05%
388.16	1112.00	1150.81	1110.47	0.14%	677.63	2312.00	2392.67	2313.27	0.06%
400.79	1162.00	1202.55	1160.51	0.13%	678.54	2315.83	2396.61	2317.09	0.05%
413.32	1212.00	1254.29	1210.56	0.12%	690.53	2365.83	2448.37	2367.23	0.06%
425.76	1262.00	1306.03	1260.62	0.11%	690.82	2367.06	2449.62	2368.45	0.06%
438.12	1312.00	1357.77	1310.69	0.10%	702.85	2417.06	2501.40	2418.60	0.06%
450.40	1362.00	1409.49	1360.75	0.09%	714.91	2467.06	2553.11	2468.69	0.07%
462.62	1412.00	1461.24	1410.85	0.08%	727.03	2517.06	2604.88	2518.82	0.07%
474.78	1462.00	1512.98	1460.95	0.07%	739.19	2567.06	2656.59	2568.89	0.07%

Where OH, AOR, TOR, IOR, and IER are, respectively, oil height, actual oil reserve, theoretical oil reserve, improved oil reserve, and improved error ratio.

We get theoretical oil reserve of small elliptical storage tank according to cylinder volume formula:

$$v_1(h_1) = l\left(\frac{ab}{2}\pi - a\sqrt{s_1} + \frac{a}{b}h_1\sqrt{s_1} + ab\arcsin\left(-1 + \frac{h_1}{b}\right)\right), \tag{3}$$

where $s_1 = -h_1(h_1 - 2b)$, l is the length of the storage tank cylinder, and h_1 is oil height of the tank capacity chart.

According to the data provided by the topic A of 2010 National Mathematical Contest in Modeling (Table 1) [26], we substitute the known data into formula (3) and then compare the theory oil reserve with the actual oil reserve. It is obvious that the proportional error is so large, as high as 3.4% ~ 3.5%, that it is necessary to take an error analysis.

2.1.1. Error Analysis. With the increase of liquid level, the part of the pipe submerged in the oil is increasing, which makes the theoretical data larger than the actual data. According to the actual situation, the capacity of the pipe in the oil and the probe will take a linear change. Fit the two groups of data, namely, the theoretical and actual oil reserve difference values and the height of the liquid. The results are shown in the Figure 3.

It turns out to be that the above two groups of data meet the linear relationship, and the curve fitting goes to $R^2 = 0.9967$ from the diagram. It also obtains the capacity of the pipe in the oil and the probe which is named Δv_1:

$$\Delta v_1 = 1.3493h_1 - 12.031. \tag{4}$$

2.1.2. Model One (Improved Model 1). The relationship between the capacity and the height of the oil in the tank without deflection can be acquired through formula (3), (4)

$$\bar{v}_1 = l\left(\frac{ab}{2}\pi - a\sqrt{-h_1(h_1 - 2b)} + \frac{a}{b}h_1\sqrt{-h_1(h_1 - 2b)} + ab\arcsin\left(-1 + \frac{h_1}{b}\right)\right) - \Delta v_1, \tag{5}$$

where l is the length of storage tank cylinder and h_1 is oil height of the tank capacity chart.

Substituting the oil height into formula (5) can obtain the improved oil reserve and the error ratio is within 0.47%. It means that the precision has been improved by more than 10 times compared with the original model. The specific testing data are shown in Table 1.

2.2. Model 2 for Tank Capacity Chart Calibration Problem of Small Elliptic Storage Tank at an Inclination Angle α of 4.1°. When the inclination angle α equals 4.1°, take left inclination for example as shown in Figure 4.

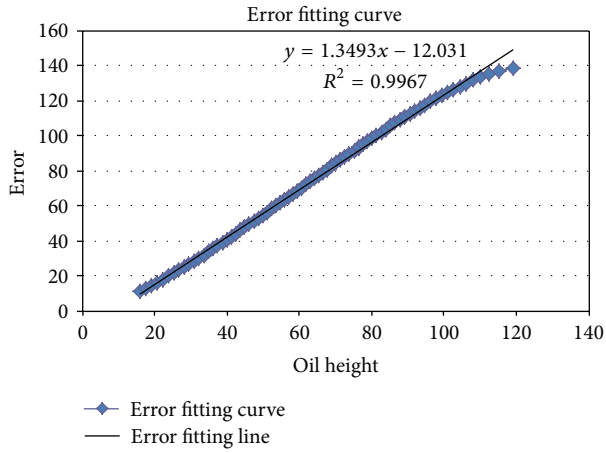


Figure 3: The fitting result of the difference between the theoretical and actual oil reserve and the height of the liquid.

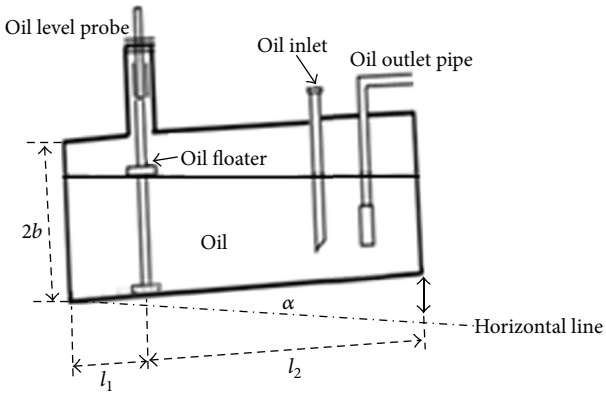


Figure 4: Facade schematic of small elliptic storage tank.

Considering the relation of mutative oil reserve and oil height, this problem can be divided into three conditions to discuss as shown in Figure 5.

Make the lower left quarter of the storage tank for origin, the length of storage tank for x shaft, and the high for y shaft. Then the coordinate system can be built as shown in Figure 6.

Based upon Figures 5 and 6, (1) there is little oil in storage tank that is; oil level is under line AB. Now, $0 \le h_2 < l_2 \tan\alpha$.

(2) There is moderate oil in storage tank; that is, oil level should be between line CD and AB. Now, $l_2 \tan\alpha \le h_2 < 2b - l_1 \tan\alpha$.

(3) there is much oil in storage tank, that is; oil level should be over line CD. Now, $2b - l_1 \tan\alpha \le h_2 \le 2b$.

For above three situations, we build model 2 according to the relation of oil height and oil reserve. The detailed solving is below.

First of all, establish equation of the liquid level. Obviously, the slope of this line is $-\tan\alpha$ and

$$k = \frac{h_2 - b}{l_1} = -\tan\alpha, \tag{6}$$

where l_1 is the length of OC.

The other equation is obtained as follows:

$$b = h_2 + l_1 \tan\alpha. \tag{7}$$

So, the relation between oil height y and horizontal ordinate x is defined as follows:

$$y = (l_1 - x)\tan\alpha + h_2. \tag{8}$$

Make differential on both sides of the function at the same time which can obtain the following:

$$dx = -\cot\alpha\, dy. \tag{9}$$

The theoretical oil reserve of storage tank at longitudinal angle α of 4.1 degrees can be acquired through the stereoscopic volume formula [27] with known parallel cross-section area,

$$v_2 = \int_0^{245} A(x)\, dx; \tag{10}$$

that is,

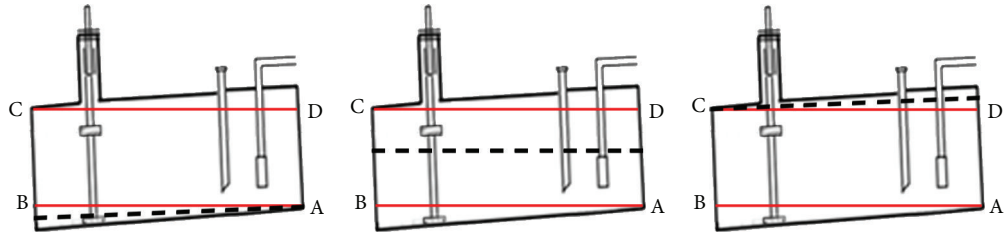
$$v_2 = \int_0^{l} \left(\frac{ab}{2}\pi - a\sqrt{p_1} + \frac{a}{b} y \sqrt{p_1} + ab \arcsin\left(-1 + \frac{y}{b}\right)\right) dx, \tag{11}$$

where $p_1 = -y(y-2b)$, $A(x)$ is the parallel cross-section area when inclined angle with deflection of α is 4.1 degrees, l is the length of storage tank cylinder, and the value of y is $(l_1 - x)\tan\alpha + h_2$.

As was discussed above, the relation model between oil height h_2 and oil reserve of storage tank can be obtained, as shown in the following model:

$$V_2 = \begin{cases} 10^{-3} \displaystyle\int_0^{h_2 + l_1 \tan\alpha} \cot\alpha \left[\frac{ab}{2}\pi - a\sqrt{-y(y-2b)} + \frac{a}{b} y \sqrt{-y(y-2b)} + ab \arcsin\left(-1 + \frac{y}{b}\right)\right] dy & 0 \le h_2 < l_2 \tan\alpha, \\ 10^{-3} \displaystyle\int_{h_2 - l_2 \tan\alpha}^{h_2 + l_1 \tan\alpha} \cot\alpha \left[\frac{ab}{2}\pi - a\sqrt{-y(y-2b)} + \frac{a}{b} y \sqrt{-y(y-2b)} + ab \arcsin\left(-1 + \frac{y}{b}\right)\right] dy & l_2 \tan\alpha \le h_2 < 2b - l_1 \tan\alpha, \\ v_1(2b) - 10^{-3} \displaystyle\int_0^{h_2 - l_2 \tan\alpha} \cot\alpha \left[\frac{ab}{2}\pi - a\sqrt{-y(y-2b)} + \frac{a}{b} y \sqrt{-y(y-2b)} + ab \arcsin\left(-1 + \frac{y}{b}\right)\right] dy & 2b - l_1 \tan\alpha \le h_2 \le 2b. \end{cases} \tag{12}$$

According to the data provided by 2010 National Mathematical Contest in Modeling the title of A [26] (Table 1), model 2 can be tested by the inclined oil-taking data. What is more, the displayed oil reserve of oil height between



(a) Little oil in storage tank (b) Moderate oil in storage tank (c) Much oil in storage tank

Figure 5: Different oil reserve profiles.

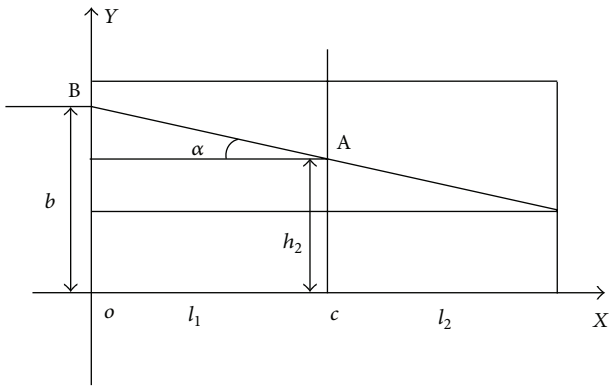


Figure 6: Facade schematic of small elliptic storage tank with longitudinal inclination.

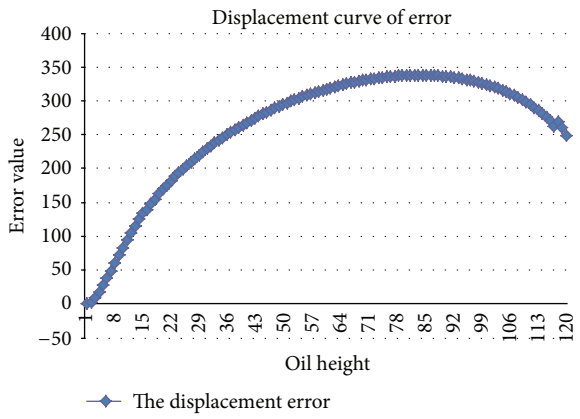


Figure 7: Calibrated error curve graph with inclined angle of 4.1 degrees.

0 and 120 cm accordingly and theoretical oil reserve of inclined deflection can be calculated. The chart can be generated as shown in Figure 7 using the calibrated error value and oil height.

The original tank capacity chart can no longer reflect the real oil capacity when the tank inclines. As shown in Figure 7, when the oil height is more than 90 cm, error should be smaller with the increasing oil height.

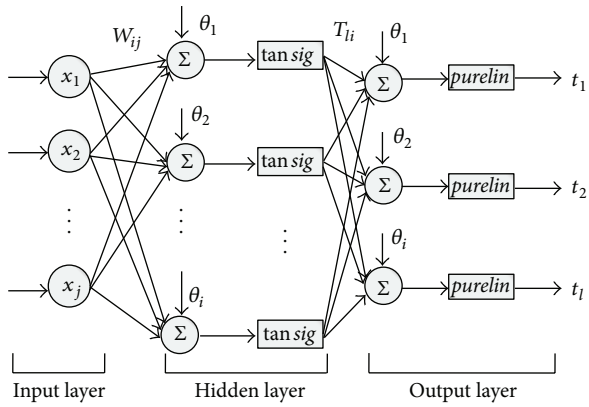


Figure 8: The topological structure of the BP network model.

2.3. Correction Model of Calibration Based upon BP Neural Network. BP neural network is a nonlinear adaptive dynamic system consisting of many parallel neurons with learning ability, memory ability, calculation ability, and intelligent processing ability [28]. Commonly, a typical BP neural network model is a full-connected neural network including input layer, hidden layer, and output layer [29, 30]. Each layer has multiple neurons, and the nodes between two adjacent layers connect in single direction. It has been proved by Kolmogorov's theorem, a neural network theory theorem, that the fully studied three-layer BP network can approach any function [28].

Some researchers also claim that networks with a single hidden layer can approximate any continuous function to any desired accuracy and are enough for most forecasting problems [31–33].

In this study, a three-layer neural network is applied in calibration of storage tank chart modeling. What's more, the network training is actually an unconstrained nonlinear minimization problem, and the nonlinear model is used in this study. Therefore, it can achieve better effect to process residual correction of this model by BP neural network.

The input node, hidden node, and output node are hypothesized as x_j, y_i and O_l, respectively. The connection weight between the input node and the hidden node is w_{ij}, while the connection weight between the hidden node and the output node is T_{li}. Giving the maximum iterating times and error precision, Figure 8 is the topological structure of the BP neural network model.

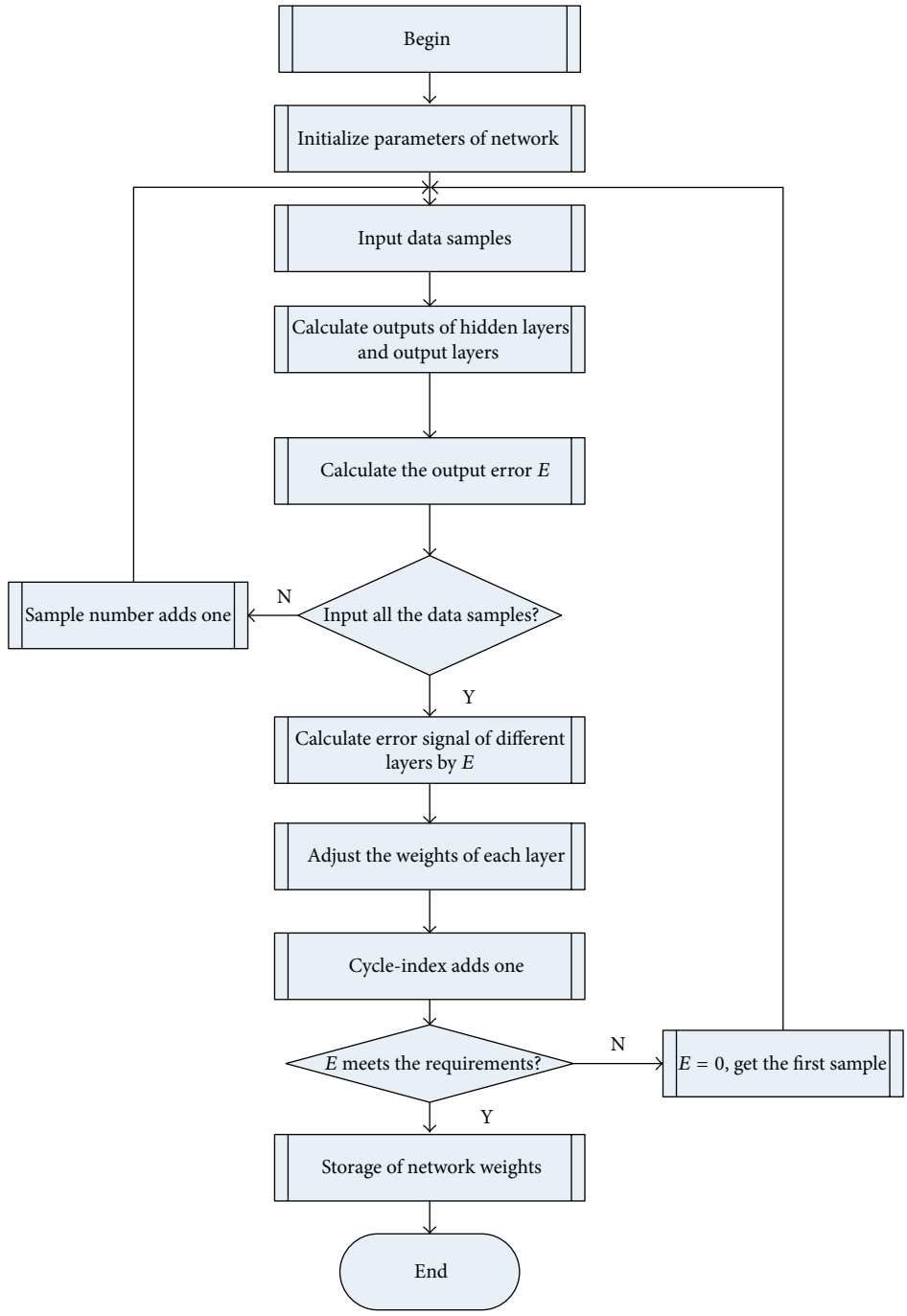


Figure 9: Flowchart of back-propagation (BP) neural networks algorithm.

Various steps of the BP training procedure are described in Figure 9.

According to Figures 8 and 9, suppose that the expected output value of the output node is t_l; then the BP network model adopts a learning algorithm for training as follows.

Firstly, Initialize by giving random number between −1 and 1 to the connection weights w_{ij}, T_{li} and threshold values θ_i, θ_l, choosing a mode and giving network to x_j, t_l.

Secondly, the output of the hidden note is $y_i = f_1(\sum_j w_{ij} x_j - \theta_i)$.

The output of the output note is $O_l = f_2(\sum_i T_{li} y_i - \theta_l)$.

Thirdly, calculate new connection weights and threshold values. The correction value of connection between hidden and output node is defined as is $T_{li}(k) = T_{li}(k) + \eta\delta_l y_i$.

The correction of the threshold values is $\theta_l(k+1) = \theta_l(k) + \eta\delta_l$.

Table 2: The oil reserve of oil-out and analysis of percentage error with BP neural network statistical table.

Site 1					Site 2				
OH/mm	AOR/L	TOR/L	FENN/L	PEC	OH/mm	AOR/L	TOR/L	FENN/L	PEC
1020.65	3464.74	3405.10	59.61	0.00%	715.32	2214.74	2222.10	10.08	0.12%
1007.73	3414.74	3361.80	53.15	0.01%	705.43	2164.74	2180.70	11.76	0.19%
994.32	3364.74	3315.90	48.64	0.01%	693.52	2114.74	2130.60	13.97	0.09%
980.96	3314.74	3269.40	45.20	0.00%	682.5	2064.74	2084.30	16.09	0.17%
967.10	3264.74	3220.10	42.05	0.08%	671.02	2014.74	2036.00	18.31	0.15%
956.01	3214.74	3180.10	39.59	0.15%	658.68	1964.74	1984.00	20.60	0.07%
941.54	3164.74	3127.20	36.34	0.04%	647.74	1914.74	1938.00	22.50	0.04%
929.69	3114.74	3083.20	33.60	0.07%	635.76	1864.74	1887.60	24.34	0.08%
916.44	3064.74	3033.40	30.45	0.03%	624.61	1814.74	1840.70	25.80	0.01%
904.14	3014.74	2986.60	27.46	0.02%	612.53	1764.74	1790.00	27.08	0.10%
891.9	2964.74	2939.50	24.47	0.03%	600.69	1714.74	1740.30	27.99	0.14%
879.23	2914.74	2890.30	21.39	0.10%	589.40	1664.74	1693.10	28.55	0.01%
868.99	2864.74	2850.20	18.96	0.15%	577.00	1614.74	1641.30	28.82	0.14%
855.13	2814.74	2795.50	15.81	0.12%	564.58	1564.74	1589.60	28.76	0.25%
844.02	2764.74	2751.20	13.45	0.00%	554.33	1514.74	1547.10	28.48	0.26%
831.64	2714.74	2701.60	11.07	0.08%	540.76	1464.74	1491.00	27.83	0.11%
820.47	2664.74	2656.50	9.20	0.04%	528.65	1414.74	1441.20	27.02	0.04%
808.16	2614.74	2606.50	7.50	0.03%	517.19	1364.74	1394.20	26.09	0.25%
796.00	2564.74	2556.90	6.24	0.06%	504.87	1314.74	1344.10	24.97	0.33%
785.04	2514.74	2511.90	5.51	0.11%	490.78	1264.74	1287.00	23.55	0.10%
773.07	2464.74	2462.60	5.18	0.12%	478.06	1214.74	1235.90	22.20	0.09%
762.09	2414.74	2417.20	5.30	0.12%	465.97	1164.74	1187.70	20.88	0.18%
750.81	2364.74	2370.30	5.86	0.01%	452.40	1114.74	1134.00	19.39	0.01%
739.42	2314.74	2322.90	6.84	0.06%	439.98	1064.74	1085.30	18.04	0.24%
727.09	2264.74	2271.40	8.32	0.07%	425.83	1014.74	1030.30	16.55	0.10%

Where OH, AOR, TOR, FENN, and PEC are, respectively, oil height, actual oil reserve, theoretical oil reserve, fitting error of neural network, and proportional error with correction.

The correction value of connection between input and hidden node is defined as: $w_{ij}(k+1) = w_{ij}(k) + \eta' \theta_i' x_j$.

The correction of the threshold values: $\theta_i(k+1) = \theta_i(k) + \eta' \delta_i'$.

Where η and η' reflect learning efficiency, $\delta_l = (t_l - O_l) \cdot O_l \cdot (1 - O_l)$, $\delta_i' = y_i(1 - y_i) \sum_l \delta_l T_{li}$.

Lastly, select the next input mode and return to step (2). Keep training until the error precision of the network settings meets the requirements. Then finish the training. Thus the BP neural network model is established.

Calibration correction: take the oil-out level height data of the small longitudinal tilting elliptical tank as input data while take the D-value between the theoretical oil reserve and the actual measurement of oil as output data. Construct a BP neural network model with single input, single output and hidden layer with three-node by matlab 2010.

Then train the model with inspecting data of the oil-in level height data and the practical measurement oil reserve. The training results are shown in Figures 10 and 11.

As mentioned above, it turns out to be that the accuracy of the results of the correction BP neural network model is very high. Part of the results can be seen in Table 2.

As shown in Table 2, the proportional error of theoretical value and experimental measurement value with BP neural network correction ranges from 0.00% to 0.38%. Error reduces a lot more than before. Using the correction model, the calibration of tank capacity chart value can be calculated with the internal of oil height for 1 cm after the deflection of storage tank as shown in Table 3.

3. Establishment and Solution of the Model with Deflection Identification and Calibration of Tank Capacity Chart of Actual Storage Tank

3.1. Model 3 of Actual Storage Tank with Longitudinal Inclination and Lateral Deflection

3.1.1. Model Establishment of Actual Storage Tank with Longitudinal Inclination. The graph in Figure 12 clearly shows that

$$v_3 = v_C + v_L + v_R, \tag{13}$$

$$y_1 = h + l_1 \tan\alpha,$$

$$y_2 = h - \left(\frac{l}{2} + l_2\right) \tan\alpha. \tag{14}$$

Table 3: Calibration results of tank capacity chart of the small deflected elliptic storage tank after the correction of BP neural network.

Site 1		Site 2		Site 3		Site 4	
OH/cm	TOR/L	OH/cm	TOR/L	OH/cm	TOR/L	OH/cm	TOR/L
0	1.03	31	615.49	62	1847.65	93	3118.02
1	3.42	32	650.15	63	1888.48	94	3157.49
2	6.55	33	685.36	64	1929.13	95	3196.50
3	10.57	34	721.10	65	1969.62	96	3235.08
4	15.59	35	757.36	66	2009.97	97	3273.22
5	21.67	36	794.10	67	2050.20	98	3310.96
6	28.92	37	831.32	68	2090.35	99	3348.43
7	37.39	38	868.99	69	2130.48	100	3385.85
8	47.17	39	907.08	70	2170.61	101	3423.58
9	58.30	40	945.60	71	2210.80	102	3462.14
10	70.85	41	984.50	72	2251.08	103	3502.02
11	84.86	42	1023.78	73	2291.50	104	3543.18
12	100.40	43	1063.41	74	2332.10	105	3584.59
13	117.51	44	1103.38	75	2372.89	106	3624.60
14	136.25	45	1143.66	76	2413.88	107	3661.99
15	159.60	46	1184.22	77	2455.08	108	3696.57
16	180.85	47	1225.04	78	2496.49	109	3728.74
17	203.42	48	1266.10	79	2538.10	110	3759.03
18	227.17	49	1307.36	80	2579.86	111	3787.81
19	252.01	50	1348.80	81	2621.76	112	3815.32
20	277.86	51	1390.38	82	2663.74	113	3841.66
21	304.66	52	1432.06	83	2705.77	114	3866.82
22	332.36	53	1473.84	84	2747.80	115	3890.82
23	360.90	54	1515.64	85	2789.78	116	3913.55
24	390.25	55	1557.46	86	2831.66	117	3934.90
25	420.38	56	1599.25	87	2873.39	118	3943.58
26	451.23	57	1640.96	88	2914.91	119	3962.31
27	482.79	58	1682.60	89	2956.19	120	3979.40
28	515.02	59	1724.10	90	2997.17		
29	547.90	60	1765.45	91	3037.83		
30	581.40	61	1806.64	92	3078.12		

Where OH is oil height and TOR is theoretical oil reserve.

In the case of no deflection, the formula in references [34] can be cited; namely,

$$v_C = \frac{a}{b} l \left[q_h + b^2 \arcsin\left(\frac{h}{b} - 1\right) + \frac{1}{2}\pi b^2 \right], \quad (15)$$

where $q_h = (h - b)\sqrt{h(2b - h)}$,

$$v_S = \frac{\pi ac}{2b^2} \left[b^2 (h - b) - \frac{1}{3}(h - b)^3 + \frac{2}{3} b^3 \right], \quad (16)$$

where c reflects sagittal of the storage tank.

Calculate the oil capacity in the tank with longitudinal angel of α by integration, as shown in Figure 12. As both sides of the storage tank are irregular solid, it is difficult to calculate accurately. But, the angel α is very tiny according to the fact that both longitudinal angle and lateral deflection angle are small angles, so cut-complement method can be adopted to make an approximate disposal. The extra volume of left approximately equals the insufficient volume of the right; that is:

$$\Delta v_L - \Delta v_R \longrightarrow 0. \quad (17)$$

Hence, the relation between oil reserve of storage tank and oil height can be defined as follows:

$$v_3 = v_C + v_{S|h=y_1} + v_{S|h=y_2}. \quad (18)$$

The calculations of v_C are as shown in Figure 13.

The boundary of the cylinder's longisection is rectangular. As shown in Figure 13, firstly, draw a line perpendicular to the base through the base's midpoint and the line intersects with the metal line. Secondly, draw a parallel line to the base through the above point of intersection. Then the parallel line, metal line, and two boundaries form two triangles named v' and v'', both of which are right-angled triangles with equal vertical angles and horizontal right-angle side.

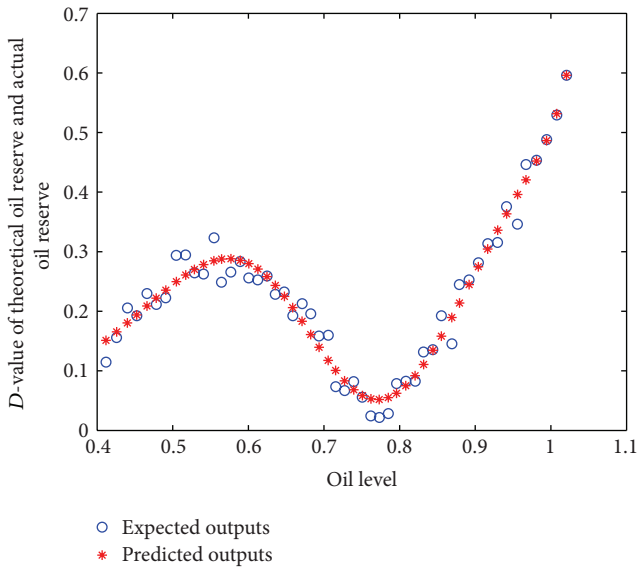


FIGURE 10: The outputs of the network prediction.

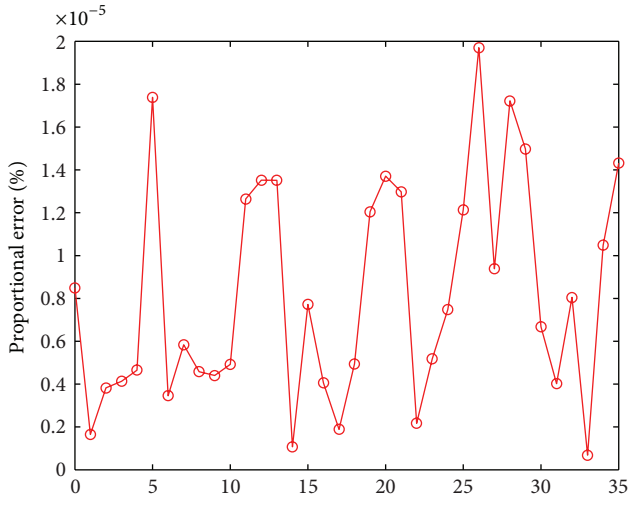


FIGURE 11: The relative error percentage of the BP neural network prediction.

So the two triangles are congruent. Apparently, their areas are also equal. According to the ZuYuan principle, their corresponding volumes in the cylinder are also equal; that is, $v' = v''$. Therefore, it can be acquired through the cut-complement method as follows:

$$v_C = \frac{a}{b} l \left[q_y + b^2 \arcsin\left(\frac{y_C}{b} - 1 \right) + \frac{1}{2}\pi b^2 \right], \quad (19)$$

where q_y is $(y_C - b)\sqrt{y_C(2b - y_C)}$.

Suppose that the metal line's slope equals k; then

$$k = \frac{y_C - h_3}{l_2} = -\tan\alpha \Longrightarrow y_C = h_3 - l_2 \tan\alpha. \quad (20)$$

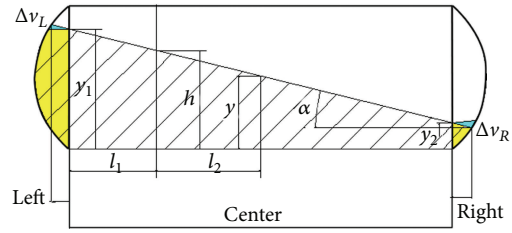


FIGURE 12: Facade schematic of actual elliptic storage tank with angle α of longitudinal inclination.

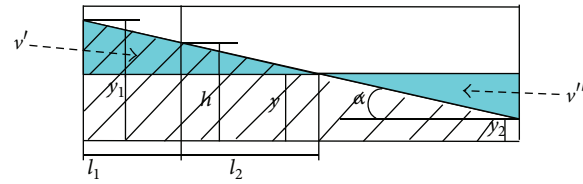


FIGURE 13: Facade schematic of cylinder with inclined angle of α.

Substituting formula (20) into formula (19),

$$v_C = \frac{a}{b} l \left[(h_3 - l_2 \tan\alpha - b)\sqrt{q_a} + b^2 \arcsin\left(\frac{h_3 - l_2 \tan\alpha}{b} - 1 \right) + \frac{1}{2}\pi b^2 \right], \quad (21)$$

where q_a is $(h_3 - l_2 \tan\alpha)(2b - (h_3 - l_2 \tan\alpha))$.

The spherical crown's metal line of both ends parallels the undersurface of the cylindrical, after the approximate disposal, which is equivalent to the case of no deflection. From formula (16), the calculation of v_s can be defined as follows:

$$v_{S|h=y_1} = \frac{\pi ac}{2b^2}\left[b^2 (y_1 - b) - \frac{1}{3}(y_1 - b)^3 + \frac{2}{3}b^3 \right],$$
$$v_{S|h=y_2} = \frac{\pi ac}{2b^2}\left[b^2 (y_2 - b) - \frac{1}{3}(y_2 - b)^3 + \frac{2}{3}b^3 \right]. \quad (22)$$

Substitute formula (14) into formula (22); thus

$$v_L = \frac{\pi ac}{2b^2}\left[b^2 t_b - \frac{1}{3}t_b^3 + \frac{2}{3}b^3 \right], \quad (23)$$

where $t_b = h_3 + l_1 \tan\alpha - b$.

$$v_R = \frac{\pi ac}{2b^2}\left[b^2 t_l - \frac{1}{3}t_l^3 + \frac{2}{3}b^3 \right], \quad (24)$$

Table 4: The results of calibration of storage tank with deflection of actual storage tank without correction by BP neural network.

Site 1		Site 2		Site 3		Site 4	
OH/cm	TOR/L	OH/cm	TOR/L	OH/cm	TOR/L	OH/cm	TOR/L
0	3.58	80	11827.88	160	33484.74	240	54967.27
10	140.31	90	14271.04	170	36340.02	250	57230.13
20	885.88	100	16823.93	180	39177.91	260	59331.74
30	2084.66	110	19468.14	190	41983.45	270	61244.02
40	3601.50	120	22186.49	200	44741.33	280	62931.80
50	5372.10	130	24962.73	210	47435.73	290	64346.91
60	7353.89	140	27781.26	220	50050.13	300	65410.85
70	9514.61	150	30626.89	230	52566.96		

Where OH is oil height and TOR is theoretical oil reserve.

where t_l is $h_3 - (l/2 + l_2)\tan\alpha - b$.

$$\begin{aligned} v_3 &= v_C + v_L + v_R \\ &= \frac{a}{b} l \Bigg[(h_3 - l_2 \tan\alpha - b) \\ &\quad \times \sqrt{(h_3 - l_2\tan\alpha)(2b - (h_3 - l_2\tan\alpha))} \\ &\quad + b^2 \arcsin\left(\frac{h_3 - l_2\tan\alpha}{b} - 1\right) + \frac{1}{2}\pi b^2 \Bigg] \\ &\quad + \frac{\pi ac}{2b^2}\left[b^2 t_2 - \frac{1}{3}t_2^3 + \frac{2}{3}b^3\right] \\ &\quad + \frac{\pi ac}{2b^2}\left[b^2 t_1 - \frac{1}{3}t_1^3 + \frac{2}{3}b^3\right], \end{aligned} \tag{25}$$

where $t_1 = h_3 - (l/2 + l_2)\tan\alpha - b$ and $t_2 = h_3 + l_1\tan\alpha - b$.

3.1.2. Establishment of the Model with Deflection Identification and Calibration of Tank Capacity Chart of Actual Storage Tank. As shown in Figure 14

$$\begin{aligned} h_4 &= r + t, \\ t &= \frac{s}{\cos\beta}, \\ s &= h_3 - r. \end{aligned} \tag{26}$$

Hence, the h is defined by

$$h = (h_4 - r)\cos\beta + r. \tag{27}$$

Substitute formula (27) into formula (25). Then, the theoretical model about α, β, h of oil reserve with longitudinal inclination and lateral deflection of storage tank is obtained below:

$$\begin{aligned} v_4 &= \frac{a}{b} l \Bigg[(h - l_2\tan\alpha - b) \\ &\quad \times \sqrt{(h - l_2\tan\alpha)(2b - (h - l_2\tan\alpha))} \\ &\quad + b^2 \arcsin\left(\frac{h - l_2\tan\alpha}{b} - 1\right) + \frac{1}{2}\pi b^2 \Bigg] \\ &\quad + \frac{\pi ac}{2b^2}\Bigg[b^2(h + l_1\tan\alpha - b) \\ &\qquad - \frac{1}{3}(h + l_1\tan\alpha - b)^3 + \frac{2}{3}b^3\Bigg] \\ &\quad + \frac{\pi ac}{2b^2}\left[b^2 r_a - \frac{1}{3}r_a^3 + \frac{2}{3}b^3\right], \end{aligned} \tag{28}$$

where $h = (h_4 - r)\cos\beta + r$, l is the length of storage tank cylinder, r_a is $h - (l/2 + l_2)\tan\alpha - b$, and h is the oil height of the calibration of storage tank chart.

3.2. The Determination of the Deflection Parameter. As was discussed in Section 3.1.2 and Figure 7 of model 2, the error nearby $h_4 = 150$ is micro, even without error. For this reason, equations can be established by selecting three contiguous groups of data near $h_4 = 150$ to ascertain the deflection parameters, α and β. The equations can be defined by

$$\begin{aligned} &v_4(151.073, \alpha, \beta) - v_4(150.765, \alpha, \beta) = 86.76, \\ &v_4(150.765, \alpha, \beta) - v_4(150.106, \alpha, \beta) = 187.61. \end{aligned} \tag{29}$$

Substitute the formula (28) into formula (29). Then, solve them by the software of Mathematica [35] and Matlab [36] using quasi-Newton iterative algorithm and the result can be got as follows:

$$\alpha = 2.3592^0, \qquad \beta = 3.80127^0. \tag{30}$$

The angles are very tiny which is realistic.

3.3. Model Solving of Actual Storage Tank and Calibration of Storage Tank Chart. According to the data provided by 2010 National Mathematical Contest in Modeling the title of A [26] (Table 2), we substitute α, β and collected oil height of actual storage tank into model 3. The calibration of tank capacity chart value can be calculated with the internal of oil height for 10 cm after the deflection of storage tank, as shown in Table 4.

Error can be controlled under 2%, when testing model 3 by actual collected data of storage tank. But it is still large for the volume of this tank. Similarly, in order to further reduce error and improve the accuracy of the calibration, model 3 also uses the BP neural network which is a method with self-learning ability to revise the calibration value.

Table 5: The statistics of oil reserve of actual storage tank with oil-out and the percentage of error.

Site 1					
OH/cm	AOR/L	TOR/L	ATDV	FENN/L	PEC
2014.29	46552.67	45130.62	1422.05	1422.10	0.015%
2003.74	46275.12	44843.34	1431.78	1431.80	0.011%
1995.29	46052.18	44612.73	1439.45	1439.50	0.006%
1989.53	45899.88	44455.27	1444.61	1444.60	0.004%
1985.62	45796.35	44348.27	1448.08	1448.10	0.003%
1979.33	45629.56	44175.95	1453.61	1453.60	0.002%
1972.51	45448.37	43988.83	1459.54	1459.50	0.001%
1969.31	45363.23	43900.94	1462.29	1462.30	0.000%
1961.41	45152.73	43683.70	1469.03	1469.00	0.004%
1957.15	45039.03	43566.41	1472.62	1472.60	0.005%
1951.30	44882.67	43405.18	1477.49	1477.50	0.003%
1943.47	44673.02	43189.07	1483.95	1483.90	0.006%
1938.96	44552.06	43064.44	1487.62	1487.60	0.006%
1934.05	44420.22	42928.64	1491.58	1491.60	0.006%
1925.88	44200.47	42702.39	1498.08	1498.10	0.005%
1921.23	44075.20	42573.46	1501.74	1501.70	0.005%
1910.98	43798.57	42288.87	1509.70	1509.70	0.009%
1899.16	43478.71	41960.06	1518.65	1518.60	0.006%
1889.86	43226.43	41700.88	1525.55	1525.50	0.006%
1884.42	43078.62	41549.09	1529.53	1529.50	0.006%
1876.58	42865.28	41330.10	1535.18	1535.20	0.005%
1873.50	42781.36	41243.99	1537.37	1537.40	0.004%
1862.44	42479.59	40934.45	1545.14	1545.10	0.005%
1851.64	42184.24	40631.69	1552.55	1552.50	0.005%
1841.46	41905.25	40345.88	1559.37	1559.40	0.003%
1836.85	41778.73	40216.32	1562.41	1562.40	0.003%
1828.91	41560.56	39992.99	1567.57	1567.60	0.002%
1826.68	41499.23	39930.22	1569.01	1569.00	0.003%
1820.43	41327.19	39754.21	1572.98	1573.00	0.001%
1815.31	41186.12	39609.91	1576.21	1576.20	0.003%
1811.69	41086.29	39507.84	1578.45	1578.40	0.000%
1807.90	40981.71	39400.92	1580.79	1580.80	0.001%
1801.67	40809.66	39225.07	1584.59	1584.60	0.001%
1798.73	40728.40	39142.04	1586.36	1586.40	0.000%
1790.75	40507.64	38916.53	1591.11	1591.10	0.000%
1784.04	40321.79	38726.76	1595.03	1595.00	0.001
1775.08	40073.31	38473.15	1600.16	1600.20	0.003

Where OH, AOR, TOR, ATDV, FENN, and PEC are, respectively, oil height, actual oil reserve, theoretical oil reserve, D-value of theoretical and actual oil reserve, fitting error of neural network, and proportional error with correction.

By taking the oil-out level height data of the actual storage tank as input data and the D-value between the theoretical oil reserve and the actual measurement of oil as output data, the whole network reflects the function mapping relation between input node and output node.

Then, train the network, so it can have a certain ability of association and prediction for this kind of problem.

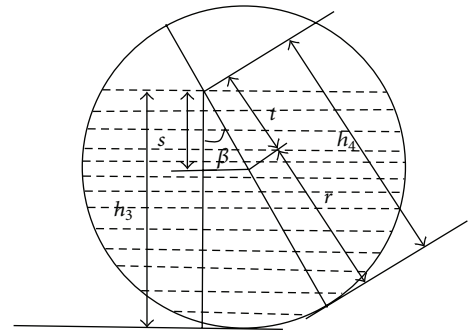


Figure 14: Cross-section schematic of lateral deflection.

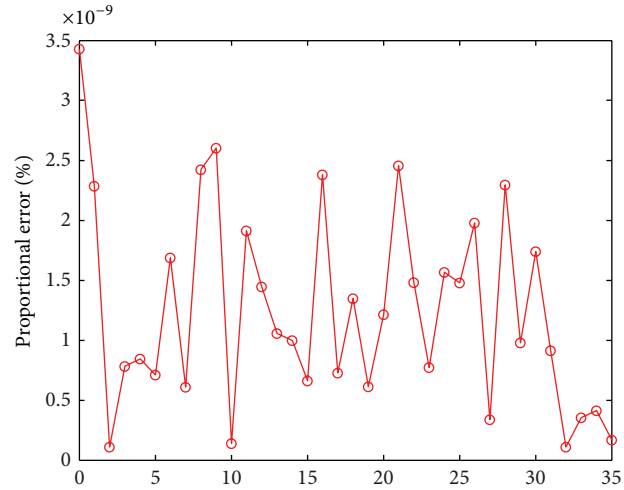


Figure 15: The relative error percentage of the BP neural network prediction.

Calibration correction: randomly take 150 groups of the oil-out level height data of the actual storage tank as input data; corresponding, take same groups of the D-value between the theoretical oil reserve and the actual measurement of oil as output data, while 50 groups of data are taken as testing data. A BP neural network model can be constructed and trained with single input and single output and three-node hidden layer by matlab 2010. The training result is as shown in Figure 15.

As discussed above, it turns out to be that the accuracy of the results of the correction BP neural network model is very high. Part of the results can be seen in Table 5.

As shown in Table 5, the proportional error of theoretical value and experimental measurement value with BP neural network correction ranges from 0.00000% to 0.00015%. Error is micro. Using the correction model, the calibration of tank capacity chart value can be calculated with the internal of oil height for 10 cm after the deflection of storage tank as shown in Table 6.

4. Conclusions

In this paper, using geometrical relationship of the storage tank, the integral models are established from simple to

Table 6: The results of calibration of storage tank with deflection of actual storage tank with correction by BP neural network.

Site 1		Site 2		Site 3		Site 4	
OH/cm	TOR/L	OH/cm	TOR/L	OH/cm	TOR/L	OH/cm	TOR/L
0	264.71	80	13394.67	160	35163.23	240	55909.19
10	705.23	90	15892.60	170	37978.85	250	58008.41
20	1703.45	100	18487.94	180	40764.00	260	59932.14
30	3105.15	110	21163.01	190	43501.00	270	61654.34
40	4783.19	120	23900.97	200	46176.56	280	63142.53
50	6682.45	130	26685.62	210	48772.32	290	64351.76
60	8767.99	140	29501.21	220	51271.70	300	65207.10
70	11012.99	150	32332.21	230	53656.90		

Where OH is oil height and TOR is theoretical oil reserve.

complicated, which makes the models simple and easy to understand. Taking into account many possible oil level conditions and giving the common theoretical relation between the oil reserve and oil height with the known deflection parameters, it has strong universality and is easy to popularize. Cut-complement algorithm is designed to construct this model, according to the special inclined angle. And the nonlinear equations are effectively solved by quasi-Newton iterative method. A novel method is applied to calibrate the storage tank chart which combines the advantages of the polynomial fitting method and BP neural network. Models are tested by the known data and the improved models are got by polynomial fitting method. Based upon fuzzification of system measurement error, BP neural network is proposed to correct results. Quasi-Newton iterative algorithm is used to calculate deflection parameters $\alpha = 2.3592^{\circ}$, $\beta = 3.80127^{\circ}$ by Mathematica, Matlab software. However, when oil in the storage tank is approximately full or there is very little oil in it, it is unable to get the accurate calibration method, so more research efforts should be devoted to validating these issues. Developing better models of solving these problems is the next step we will undertake.

Acknowledgments

This work was supported by the Natural Science Foundation of P. R. of China (90912003, 61073193), the Key Science and Technology Foundation of Gansu Province (1102FKDA010), Natural Science Foundation of Gansu Province (1107RJZA 188), and the Fundamental Research Funds for the Central Universities (lzujbky-2012-47, lzujbky-2012-48).

References

[1] J. Sun, "The discussion of horizontal tank volume about the problematic point of the verification and calculation," *Petroleum Products Application Research*, vol. 18, no. 5, pp. 20–24, 2000.

[2] Z. Li, "The calculation of horizontal storage tank volume with elliptic cylinder type," *Mathematics in Practice and Theory*, vol. 2, pp. 17–26, 1997.

[3] Y. Ji, S. Song, and Y. Tu, "The current situation and development tendency of liquid level measurement technology of storage tank," *Petroleum Engineering Construction*, vol. 32, no. 4, pp. 1–4, 2006.

[4] C. Li, T. Liang, W. zhou, and J. Lu, "The function relation of remain liquid volume and oil height of storage tank," *Petro-Chemical Equipment*, no. 6, pp. 25–27, 2001.

[5] C. Li, T. Liang, and M. Jin, "The relation of liquid volume and oil height of storage tank with deformation section," *Petro-Chemical Equipment*, no. 1, pp. 21–22, 2003.

[6] G. Si, "Calculate the liquid volume by liquid height of tank with deformation section," *Process Equipment & Piping*, no. 2, pp. 63–64, 2000.

[7] B. Gao and X. Su, "The volume calculation of different liquid height of horizontal tank with various end enclosure," *Petro-Chemical Equipment*, no. 4, pp. 1–7, 1999.

[8] C. Fu, "The volume calculation of inclined storage tank," *Journal of Heilongjiang Bayi Agricultural University*, no. 2, pp. 43–52, 1981.

[9] S. Sivaraman and W. A. Thorpe, "Measurement of tank-bottom deformation reduces volume errors," *Oil and Gas Journal*, vol. 84, no. 44, pp. 69–71, 1986.

[10] F. Kelly, "Shore tank measurement," *Quarterly Journal of Technical Papers*, vol. 1, pp. 59–62, 1987.

[11] Q. Ai, J. Huang, X. zhang, and M. Zhang, "Model of the identification of oil tank's position and the calibration of tank capacity table," *Light Industry Design*, vol. 4, 2011.

[12] S. Si, F. Hua, X. Tian, and J. Wu, "The study of the relation of any height and volume in Erect Spherical cap body," *Automation & Instrumentation*, vol. 154, pp. 15–16, 2011.

[13] Z. Li, "Study on tank capacity of tilted oil tank in elliptic cross-section," *Engineering & Test*, vol. 51, no. 2, pp. 38–40, 2011.

[14] Y. Chang, D. Zhou, N. Ma, and Y. Lei, "The problem of deflection identification of horizontal storage tank and calibration of tank capacity chart," *Oil & Gas Storage and Transportation*, vol. 31, no. 2, pp. 109–113, 2012.

[15] J. Dou, Y. Mei, Z. Chen, and L. Wang, "Model of the identification of oil tank's position and the calibration of tank capacity table," *Pure and Applied Mathematics*, vol. 27, no. 6, pp. 829–840, 2011.

[16] Y. Ai, "The study of capacity table calibration of storage tank with deflection," *Business Affection*, vol. 41, pp. 170–170, 2012.

[17] S. Ou, J. Wang, and S. Han, "Model of the identification of oil tank's position and the calibration of tank capacity table," *China Petroleum and Chemical Standard and Quality*, vol. 31, no. 4, pp. 25–26, 2011.

[18] Z. Wang, Y. Li, and R. F. Shen, "Correction of soil parameters in calculation of embankment settlement using a BP network back-analysis model," *Engineering Geology*, vol. 91, pp. 168–177, 2007.

[19] S. Tian, Y. Zhao, H. Wei, and Z. Wang, "Nonlinear correction of sensors based on neural network model," *Optics and Precision Engineering*, vol. 14, no. 5, pp. 896–902, 2006.

[20] W. He, J. H. Lan, Y. X. Yin, and Z. H. Zhang, "The neural network method for non-linear correction of the thermal resistance transducer," *Journal of Physics*, vol. 48, no. 1, pp. 207–211, 2006.

[21] W. Liu, *MATLAB Program Design and Application*, China Higher Education Press, Beijing, China, 2002.

[22] B. H. M. Sadeghi, "BP-neural network predictor model for plastic injection molding process," *Journal of Materials Processing Technology*, vol. 103, no. 3, pp. 411–416, 2000.

[23] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The Bulletin of Mathematical Biophysics*, vol. 5, no. 4, pp. 115–133, 1943.

[24] L. Sun and Y. Li, "Review of on-line defects detection technique for above ground storage tank floor monitoring," in *Proceedings of the 8th World Congress on Intelligent Control and Automation (WCICA '10)*, vol. 8, pp. 4178–4181, July 2010.

[25] T. Hu, P. Yuan, and J. Din, "Applications of artificial neural network to hydrology and water resources," *Advances in Water Science*, vol. 11, pp. 76–81, 1995.

[26] China Society for Industrial and Applied Mathematics, *2010 National Mathematical Contest in Modeling the Title of a [DB/OL]*, China Society for Industrial and Applied Mathematics, Beijing, China, 2010.

[27] Department of pplied Mathematics of Tongji university, *Advanced Mathematics (New Paris Interiors)*, Higher Education Press, Beijing, China, 2006.

[28] D. Zhang, *MATLAB Neural Network Application Design*, Mechanical Industry Press, Beijing, China, 2009.

[29] ASCE Task Committee on Application of Artificial Neural Networks in Hydrology, "Artificial neural networks in hydrology. I: preliminary concepts," *Journal of Hydrologic Engineering*, vol. 5, no. 2, pp. 115–123, 2000.

[30] ASCE Task Committee on Application of Artificial Neural Networks in Hydrology, "Artificial neural networks in hydrology. II: hydrologic applications," *Journal of Hydrologic Engineering*, vol. 5, no. 2, pp. 124–137, 2000.

[31] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals, and Systems*, vol. 2, no. 4, pp. 303–314, 1989.

[32] R. Hecht-Nielsen, *Neurocomputing*, Addison-Wesley, Menlo Park, Calif, USA, 1990.

[33] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359–366, 1989.

[34] J. Guan and H. Zhao, "Practical methods of oil volume calibration of horizontal storage tank," *Metrology & Measurement Technique*, vol. 31, no. 3, pp. 21–36, 2004.

[35] M. Yang and J. Lin, *Mathematica Base and Mathematical Software*, Dalian University of Technology Press, Dalian, China, 2007.

[36] Science and Technology Products Research Center of Feisi, *Neural Network Theory and the Implementation of Matlab7*, pp. 99–108, Electronic Industry Press, Beijing, China, 2005.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 208964, 9 pages
http://dx.doi.org/10.1155/2013/208964

Research Article

A New Strategy for Short-Term Load Forecasting

Yi Yang,[1] Jie Wu,[2] Yanhua Chen,[1] and Caihong Li[1]

[1] *School of Information Science and Engineering, Lanzhou University, Lanzhou, Gansu 730000, China*
[2] *School of Mathematics and Statistics, Lanzhou University, Lanzhou 730000, China*

Correspondence should be addressed to Jie Wu; wuj19870903@126.com

Received 28 February 2013; Accepted 22 April 2013

Academic Editor: Fuding Xie

Electricity is a special energy which is hard to store, so the electricity demand forecasting remains an important problem. Accurate short-term load forecasting (STLF) plays a vital role in power systems because it is the essential part of power system planning and operation, and it is also fundamental in many applications. Considering that an individual forecasting model usually cannot work very well for STLF, a hybrid model based on the seasonal ARIMA model and BP neural network is presented in this paper to improve the forecasting accuracy. Firstly the seasonal ARIMA model is adopted to forecast the electric load demand day ahead; then, by using the residual load demand series obtained in this forecasting process as the original series, the follow-up residual series is forecasted by BP neural network; finally, by summing up the forecasted residual series and the forecasted load demand series got by seasonal ARIMA model, the final load demand forecasting series is obtained. Case studies show that the new strategy is quite useful to improve the accuracy of STLF.

1. Introduction

Load forecasting has always been an essential and important topic for power systems, especially the STLF, which is fundamental in many applications such as providing economic generation, system security, and management and planning [1]. Basic operation functions such as unit commitment, economic dispatch, fuel scheduling, and unit maintenance can be performed more efficiently with an accurate forecasting [2]. However, load forecasting is a difficult task as the load at a given hour is dependent not only on the load at the previous hour but also on the load at the same hour on the previous day and on the load at the same hour on the day with the same denomination in the previous week. The STLF is also difficult to handle due to the nonlinear and random-like behaviors of system load, weather conditions, and variations of social and economic environments, and so forth [3]. So how, to improve the forecasting accuracy is still a difficult and critical problem.

During the past years, a wide variety of techniques have been developed for STLF to improve the forecasting accuracy. For example, in [4], a hybrid fuzzy modeling method by employing the orthogonal least squares method to create the fuzzy model and a constrained optimization algorithm to perform the parameter learning for STLF was presented. Another fuzzy modeling technique was also used for STLF in [5]. Yang and Stenzel proposed a new regression tree method for STLF in [6]; both increment and nonincrement trees were built according to the historical data to provide the data space partition and input variable selections then support vector machine was employed to the samples of regression tree nodes for further fine regression; results of different tree nodes were integrated through weighted average method to obtain the comprehensive forecasting result. Based on state space and Kalman filter approach, a novel time-varying weather and load model for solving the STLF problem was proposed in [7], where time-varying state space model was used to model the load demand on hourly basis while Kalman filter was used recursively to estimate the optimal load forecast parameters for each hour of the day. Considering that STLF was always affected by a variety of nonlinear factors, a mapping function was defined for each factor to identify the nonlinearity in [8]. Several other typical approaches for STLF can be found in [9–12].

The seasonal ARIMA model is frequently employed to forecast data with seasonal item. For instance, Choi et al. [13] used a hybrid SARIMA wavelet transform method for sales

forecasting. Egrioglu et al. [14] proposed a hybrid approach based on SARIMA and partial high order bivariate fuzzy time series forecasting model and applied the hybrid model to two real seasonal time series. Besides, Chen and Wang [15] developed a hybrid SARIMA and support vector machines in forecasting the production values of the machinery industry in Taiwan. Considering that the load demand series always contain seasonal item, the seasonal ARIMA model is adopted in this paper.

The BP neural network is a kind of typical feed forward network, through the network structure positive transfer method; using the training function reverse revision network weight matrix and threshold, the BP neural network completes samples training model of the structure and then uses the built training model to complete the treatment of the sample to be measured [16]. The BP neural network model is applied to a wide field of forecasting. As Ke et al. [17] used the genetic algorithm-BP neural network to forecast the electricity power industry loan, Li et al. [18] adopted the BP neural network to the prediction of the mechanical properties of porous NiTi shape memory alloy prepared by thermal explosion reaction. In addition, the BP neural network can also be used for evaluation and classification: Li and Chen [19] utilized the BP neural network algorithm to study the sustainable development evaluation of highway construction project. Bao and Ren showed the wetland landscape classification based on the BP neural network in Dalinor Lake area in [20]. For the BP neural network can approximate the underlying function of the curves to any arbitrary degree of accuracy, this model is also employed to constitute the hybrid model of this paper.

Both ARIMA and BP neural network models have achieved successes in their own linear or nonlinear domains. Though a large number of models have been used to load demand forecasting, more techniques for STLF should be sought to further improve the predictive capability. For this purpose, a hybrid model of combining the seasonal ARIMA model and BP neural network is proposed in this paper. Firstly, the seasonal ARIMA model is adopted to forecast the load demand day ahead then BP neural network is used to forecast the residual series. Finally, by summing up the forecasted residual series and the forecasted load demand, the final load demand is obtained.

The remainder of this paper is organized as follows. Section 2 introduces the combined forecasting model theory. In Section 3, seasonal ARIMA model and BP neural network are presented. In Section 4, a case study of forecasting electricity load of South Australia (SA) State of Australia is demonstrated. Section 5 concludes this paper.

2. The Combined Forecasting Model

The combined forecasting theory states that if there exist M kinds of forecasting models for solving a certain forecasting problem, with properly selected weight coefficients, several forecasting methods' results can be added up. Assume that y_t $(t = 1, 2, \ldots, L)$ is the actual time series data, L is the number of sample points, $\widehat{y}_{it}$ $(i = 1, 2, \ldots, M, t = 1, 2, \ldots, L)$ is the weight coefficient for the ith forecasting model, the mathematical model of the combined forecasting model can be expressed as follows:

$$y_t = \sum_{i=1}^{M} \omega_i \left(y_{it} + e_{it}\right), \quad t = 1, 2, \ldots, L \tag{1}$$

$$\widehat{y}_t = \sum_{i=1}^{M} \widehat{\omega}_i \widehat{y}_{it}, \quad t = 1, 2, \ldots, L, \tag{2}$$

where $\widehat{\omega}_i$ is the estimated value for ω_i and $\widehat{y}_t$ is the combined forecasting value.

Determination of the weight coefficients for each individual model is the key step in a construction of a combined forecasting model. This can be achieved by solving an optimization problem which minimizes the absolute error summation for the combined model. This optimization problem can be expressed as follows:

$$\begin{aligned} \text{Min} \quad & \sum_{t=1}^{L} \left|y_t - \widehat{y}_t\right|, \\ \text{St} \quad & \sum_{i=1}^{M} \omega_i = 1, \\ & 0 \le \omega_i \le 1, \quad i = 1, 2, \ldots, M. \end{aligned} \tag{3}$$

The optimization process can be terminated provided that the predefined absolute error summation is reached or the maximum iteration number is reached.

3. The Hybrid Model

3.1. Review of the Seasonal ARIMA Model. Seasonal ARIMA is an extension of autoregressive integrated moving average (ARIMA), which is one of the most common models in time series forecasting analysis. They originated from autoregressive (AR) model which was firstly proposed by Yule in 1972, moving average (MA) model which was firstly proposed by Walker in 1931, and AR and MA combination model autoregressive integrated moving average model (ARMA). Only in sequence where circumstances are stable, ARMA model is effective, but SARIMA and ARIMA do not have such restrictions. Generally speaking, it is assumed that the time series $\{x_t \mid t = 1, 2, \ldots, k\}$ has mean zero. A nonseasonal ARIMA model of order (p, d, q) (denoted by ARIMA (p, d, q)) representing the time series can be expressed as follows:

$$\begin{aligned} x_t &= \phi_1 x_{t-1} + \phi_2 x_{t-2} + \cdots + \phi_p x_{t-p} + \varepsilon_1 \\ &\quad - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \cdots - \theta_q \varepsilon_{t-q}, \\ \phi(B) \nabla^d x_t &= \theta(B) \varepsilon_t, \end{aligned} \tag{4}$$

where x_t and ε_t are the actual value and random error at time t, respectively, ϕ_t and θ_t are the coefficients, p is the order of autoregressive, q is the order of moving average polynomials, B denotes the backward shift operator, $\nabla^d = (1 - B)^d$, d

is the order of regular differences and $\phi(B)$ and $\theta(B)$ are, respectively, defined as follows

$$\begin{aligned}\phi(B) &= 1 - \phi_1 B - \phi_2 B^2 - \cdots - \phi_p \\ B^p \theta(B) &= 1 - \theta_1 B - \theta_2 B^2 - \cdots - \theta_q B^q.\end{aligned} \tag{5}$$

Random errors, ε_t, are assumed to be independently and identically distributed with a mean of zero and a constant variance of σ^2, and the roots of $\phi(x) = 0$ and $\theta(x) = 0$ all lie outside the unit circle [21].

Equation (1) entails several important special cases of the ARIMA family of models. If $q = 0$, then (1) becomes an AR model for order p. When $p = 0$, the model reduces to an MA model of order q. One central task of ARIMA model building is to determine the appropriate model order (p, q). Similarly, a seasonal model $(p, d, q)(P, D, Q)_s$ can be written as follows (using the second expression):

$$\begin{aligned}&\varphi_p(B)\,\Phi_p\left(B^s\right)(1-B)^d\left(1-B^s\right)^D X_t \\ &\quad = \theta_q(B)\,\Theta_Q\left(B^s\right)\varepsilon_t,\end{aligned} \tag{6}$$

where p and q are the nonseasonal ARMA orders, d is the number of trend differences, X_t is the observation at time t, B is the back-shift operator, ε_t is the residual (an error term at t time period), $\varphi_p(B)$, $\theta_q(B)$ are polynomials in B of order p and q, respectively, P and Q are the seasonal ARMA orders, D is the number of seasonal differences, s is the seasonal period, and $\Phi_P(B^s)$, $\Theta_Q(B^s)$ are polynomials in B^s of order P and Q, respectively [22].

The SARIMA model formulation includes four steps [23]:

(i) Identification of the SARIMA $(p, d, q)(P, D, Q)_s$ structure: use autocorrelation function (ACF) and partial autocorrelation function (PACF) to develop the rough function.

(ii) Estimation of the unknown parameters.

(iii) Use of goodness-of-fit tests on the estimated residuals.

(iv) Forecast future outcomes based on the known data.

The steps of this modeling are identification, estimation, availability tests, and forecasting. In the following, we will specifically introduce the four steps. Identification, the appropriate models are determined from all possible models in this stage. The step of identification consists of determining appropriate AR, MA, or ARMA processes and the order of AR, MA, and/or ARMA models. Estimation, in this step parameters are estimated by using ordinary least squares (OLS) and sometimes nonlinear estimation methods. Estimated parameters of AR and MA processes included in ARIMA model should analyze whether they are stationary and invertible or not, respectively. Availability tests, in this stage, it is determined that the estimated ARIMA models are harmonized or not by diagnostic checking. On the other hand, estimated ARIMA model should have to be carried out the assumption that the processes of AR and MA have to be in the unit circle and the assumption of normality. Forecasting, estimated ARIMA models which keep assumptions as expressed above are used in forecasting in this stage.

3.2. Brief Introduction to the Back Propagation (BP) Neural Network. Artificial neural networks (ANN) is a typical kind of intelligent learning algorithm; it is widely used in some practical application, such as pattern classification, function approximation, optimization, forecast, and automation control [24, 25]. In this section, we will introduce the standard multilayer feed-forward neural network (FNN). FNN is a multilayer perception neural network; it is relative to the single perception neural network that can only solve linear separable classification problem. In order to increase the classification ability of the network, the only method is to use the multilayer network. Because the hidden layer neurons are introduced in the multilayer neural network, neural network has better classification and memory ability, so the corresponding learning algorithm becames the focus of research. In 1986, Rumelhart put forward the BP algorithm which solves the learning problems of the multilayer neural network layer implied in the hidden connection weights and gives a complete mathematical deduction. Because BP algorithm overcomes the drawback of the simple perception cannot solve XOR and some other problems, BP algorithm became the main multilayer perception learning algorithm, an important mode of neural network, and widely used.

The BP, one of the most popular techniques in the field of NN, is a kind of supervised learning neural network, the principle behind which involves using the steepest gradient descent method to reach any small approximation. The learning process consists of two parts: forward propagation and back-propagation. When facing the forward propagation, after implicit unit layer processing, information from the input layer to the output layer, the state of each layer neuron affects only the state of next layer neuron. If it is not a desired output in the output layer, then transferred to a back-propagation, the error signal returns along the original neurons connected channel [26]. In the return process, change the neuron connection weights in each layer; this process is iterative, and finally makes the error signal to the permitted range. From that we can see that, in the multilayer feed forward network, there are two signals in circulation: (1) working signal: after the input signal is applied to the working signal, it propagates forward until the actual output signal is produced in the output side, it is the function of inputs and weights. (2) Error signal: the error is the difference between the actual network output and the due output; it propagates back from the output terminal layer by layer [27].

There are three layers contained in BP: input layer, hidden layer, and output layer. Two nodes of each adjacent layer are directly connected, which is called a link. Each link has a weighted value presenting the relational degree between two nodes. Assume that there are n input neurons, m hidden neurons, and one output neuron, the relationship between the output (y_t) and the inputs $(y_{t-1}, y_{t-2}, \ldots, y_{t-n})$ have the following mathematical representation:

$$y_t = \alpha_0 + \sum_{j=1}^{m} \alpha_j g\left(\beta_{0j} + \sum_{i=1}^{n} \beta_{ij} y_{t-i}\right) + \varepsilon_t. \tag{7}$$

We can infer a training process described by the following equations to update these weighted values, which can be divided into two steps.

(i) Hidden layer stage: the outputs of all neurons in the hidden layer are calculated by the following steps:

$$\begin{aligned} \text{net}_j &= \sum_{i=0}^{n} v_{ij} x_i, \quad j = 1, 2, \ldots, m, \\ y_j &= f_H\left(\text{net}_j\right), \quad j = 1, 2, \ldots, m. \end{aligned} \tag{8}$$

Here net_j is the activation value of the jth node, y_j is the output of the hidden layer, and f_H is called the activation function of a node, usually a sigmoid function as follows:

$$f_H(x) = \frac{1}{1 + \exp(-x)}. \tag{9}$$

(ii) Output stage: the outputs of all neurons in the output layer are given as follows:

$$O = f_o\left(\sum_{j=0}^{m} \omega_{jk} y_j\right). \tag{10}$$

Here f_o is the activation function, usually a line function. All weights are assigned with random values initially and are modified by the delta rule according to the learning samples traditionally [28].

Hence, the BP model of (1) in fact performs a nonlinear functional mapping from the past observations $(y_{t-1}, y_{t-2}, \ldots, y_{t-n})$ to the future value y_t that is, $y_t = f(y_{t-1}, y_{t-2}, \ldots, y_{t-n}, w) + \varepsilon_t$, where w is a vector of all parameters and f is a function determined by the network structure and connection weights. Thus, the neural network is equivalent to a nonlinear autoregressive model. Note that expression (7) implies one output node in the output layer which is typically used for one-step-ahead forecasting.

4. Simulation Results

The electric load demand data used for the simulation are sampled from South Australia (SA) State of Australia at half an hour rate, so for one day, 48 load demand data are included. Figure 1 provides the load demand of SA from June 2, 2007 to July 14, 2007.

From Figure 1, it can be found that there exists significant similarity in load demand on the same day of each week; in other words seasonal components exist in load demand on the same day of each week. So, the seasonal ARIMA model will be greatly helpful to forecast the load demand day ahead using the historical load demand on the same day several weeks ago. Using the data on June 2, June 9, and June 16 of 2007, the electric load demand on June 23 is forecasted. Then the same way, that is, using the load demand data on the same day of the three sequential weeks to forecast the load demand on the same day of the adjacent week, is adopted to

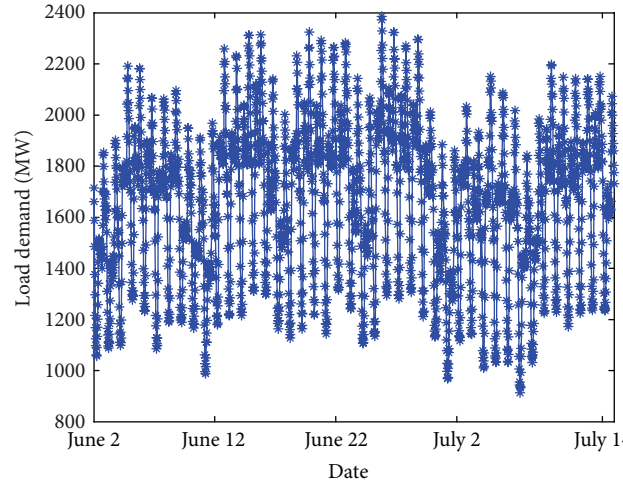


Figure 1: Load demand of SA from June 2, 2007 to July 14, 2007.

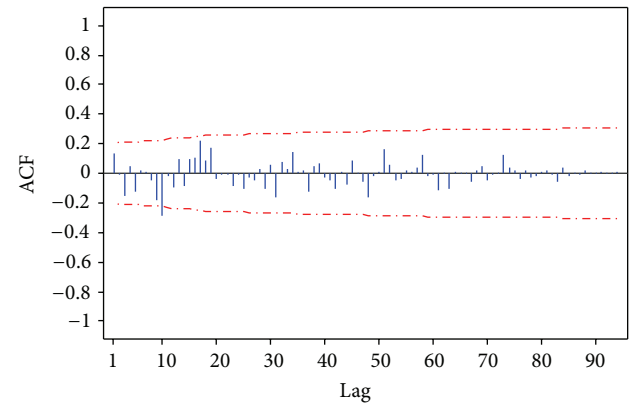


Figure 2: ACF figure in forecasting the load demand on June 23 by seasonal ARIMA model.

forecast the load demand on June 30, July 7, and July 14. Before the forecasting, values of the parameters should be estimated, obviously, $s = 48$, other parameters can be estimated by the ACF and PACF figures; values of parameters in forecasting load demand on June 23, June 30, July 7, and July 14 are listed in Table 1. In addition, as an example, ACF and PACF figures in forecasting load demand on June 23 by seasonal ARIMA model are shown in Figures 2 and 3, respectively.

By applying the estimated parameters shown in Table 1 to load demand forecasting, load demand results on June 23, June 30, July 7, and July 14 can be obtained by the seasonal ARIMA models. Forecasted load demand results are shown in Figure 4.

Using these forecasted load demand values, residual errors of load demand series on June 23, June 30, and July 7 will be obtained, as presented in Figure 5. Regarding the residual series as the original data series, the residual series on July 14 can be forecasted. From Figure 4 it can be observed that no significant variation trend can be found in the residual error series; therefore, the BP neural network, which can approximate the underlying function of the curves to any

Table 1: Parameters in seasonal ARIMA Model.

Parameters	Forecasting load demand on June 23	Forecasting load demand on June 30	Forecasting load demand on July 7	Forecasting load demand on July 14
p	1	2	1	1
d	1	1	1	1
q	1	1	2	1
P	0	1	0	1
D	1	1	1	1
Q	1	1	0	1

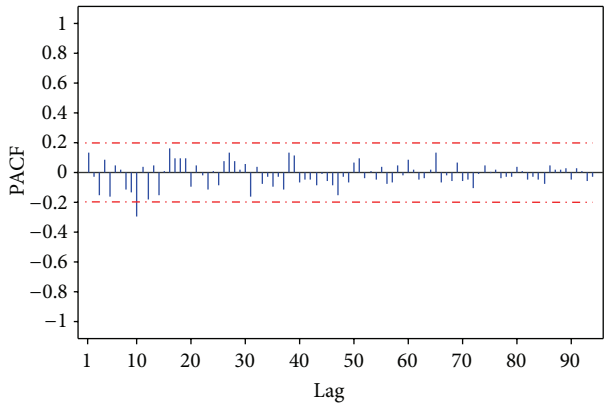


Figure 3: PACF figure in forecasting the load demand on June 23 by seasonal ARIMA model.

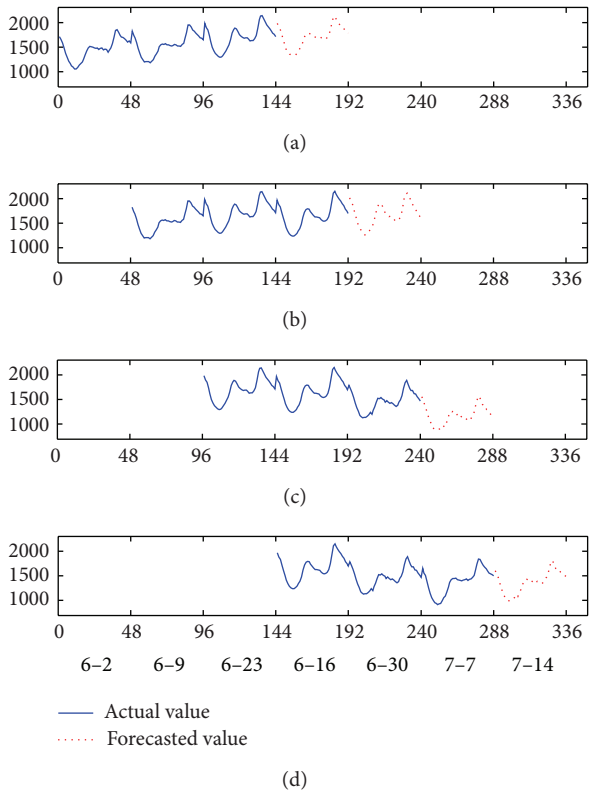


Figure 4: Forecasted load demand values by seasonal ARIMA models.

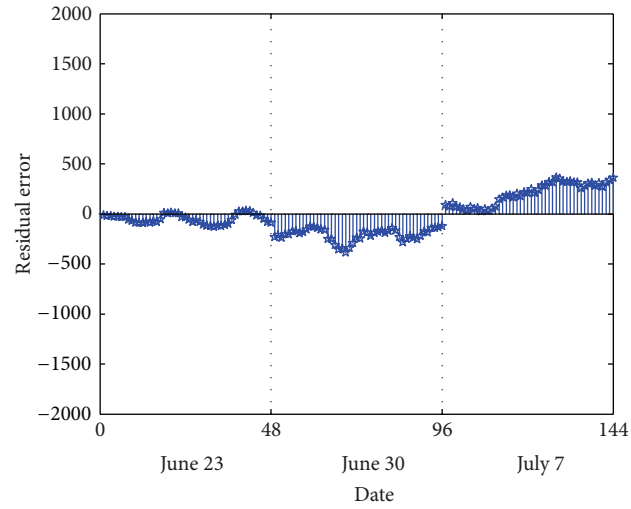


Figure 5: Residual error of the load demand forecasted by the seasonal ARIMA models.

arbitrary degree of accuracy, is employed to forecast the residual error series on July 14. In constructing the BP neural network, one of the most important tasks is training. When for training, the number of nodes in input layer is set as 2, which represent the load demand residual data on June 23 and June 30 at time t, and the corresponding 1-element output will be the residual data on July 7 at the same time, so there are total 48 samples for training. Except for determining the number of nodes in the input layer and output layer, the number of neurons in the hidden layer should also be given to construct the network. For the number of neurons in the hidden layer, we will adopt Hecht-Nelson's method [29], which is determined as follows:

$$h = 2 * i + 1, \tag{11}$$

where i is the number of inputs. So the node number in the hidden layer is 5. The structure of the BP neural network is shown in Figure 6.

Once the training data and the number of neurons in each layer have been determined, the training process can be conducted. Figure 7 shows the variation of the training error with the epoch number of the BP neural network, where the maximal epoch is 1000.

Then the forecasting can be implemented by the trained network. When for forecasting, the residual load demand

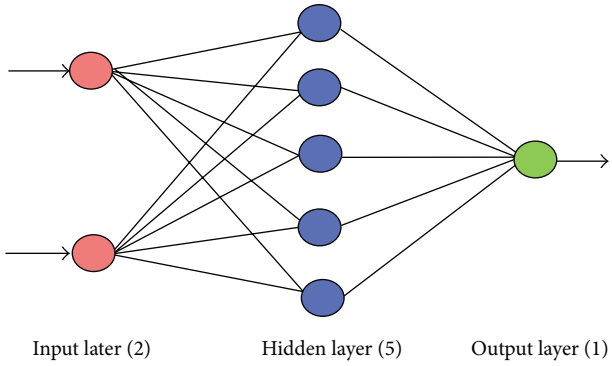


Figure 6: The architecture of the BP Neural network.

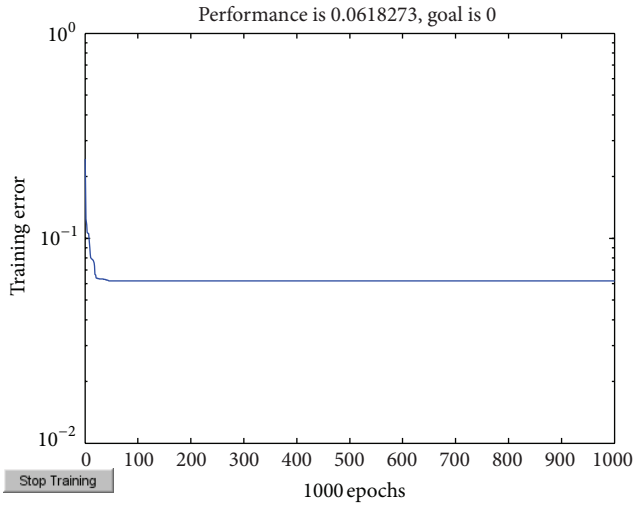


Figure 7: Variation of the training error with the epoch number of the BP neural network.

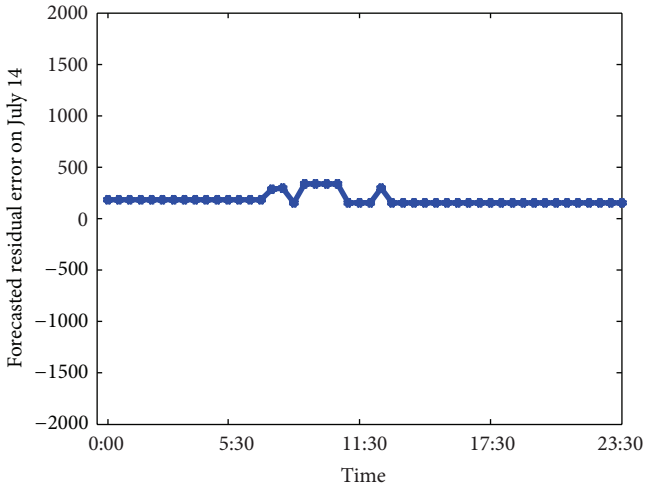


Figure 8: Residual error on July 14 forecasted by the BP neural network.

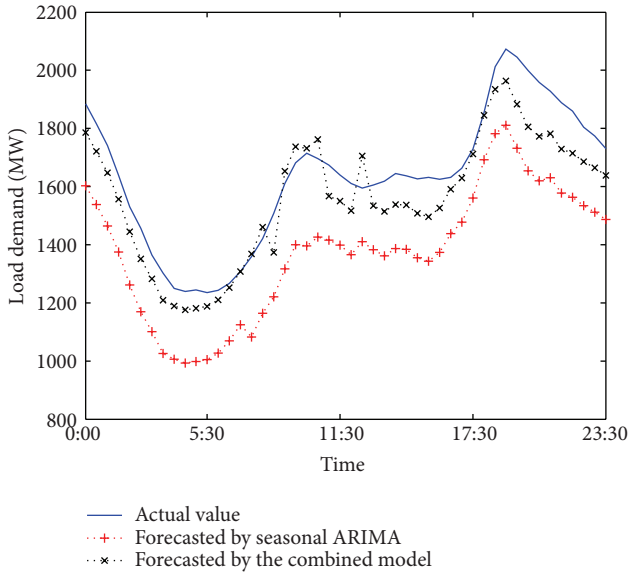


Figure 9: Actual and forecasted load demand values.

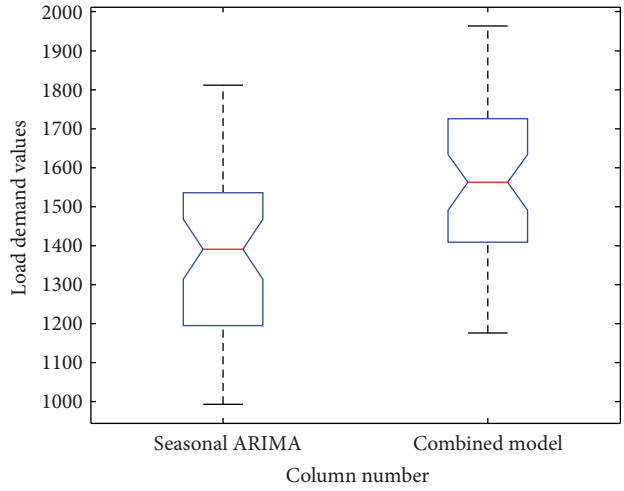


Figure 10: Box graphs of the forecasted load demand by the two models.

data on June 30 and July 7 at time t are used for inputs, with which the same time's load demand on July 14 can be forecasted. Forecasting results are plotted in Figure 8.

Finally, by summing up this forecasted residual series to the forecasted load demand obtained by seasonal ARIMA model, the final load demand can be got, which is shown in Figure 9.

Figure 10 produces whisker plot with two boxes which have lines at the lower, median, and upper quartile values of the load demand forecasted by the single seasonal ARIMA model and the combined model. It can be observed that each of the boxes includes a notch in the position of the median value.

In order to evaluate the performance of the new forecasting strategy, two error measure criteria, that is, the root mean square error (RMSE) and the mean absolute percentage error

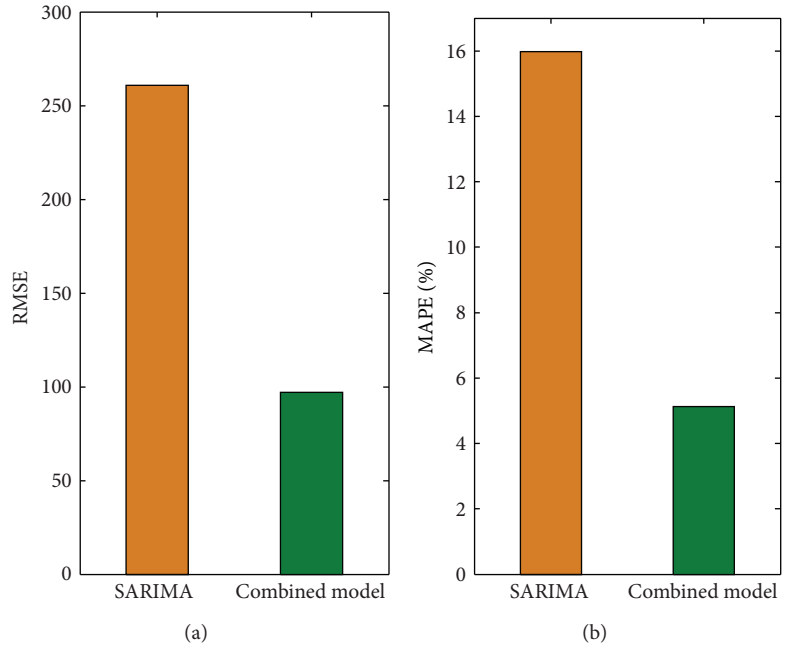


Figure 11: Bar figure of the RMSE and MAPE values.

Table 2: Comparison of RMSE and MAPE.

Models	RMSE	MAPE (%)
Individual seasonal ARIMA model	260.7376	15.98
Combined model	97.1366	5.13

(MAPE), are used; the forecasting effect is better when the loss function value is smaller. The two error measure criteria are expressed as follows:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i - \widehat{x}_i\right)^2},$$
$$\text{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{x_i - \widehat{x}_i}{x_i}\right| \times 100\%, \tag{12}$$

where x_i and $\widehat{x}_i$ represent the actual and the forecasted load demand at time i, and the value of N in our simulation is 48. Values of RMSE and MAPE obtained by individual seasonal ARIMA model and by the hybrid model based on seasonal ARIMA and BP neural network are listed in Table 2, and the corresponding bar figure are presented in Figure 11.

From Table 2 and Figure 11, it can be seen that the value of RMSE varies from 260.7376 in the individual seasonal ARIMA model to 97.1366 in the combined model, while MAPE is reduced from 15.98% to 5.13%. Therefore, the combined model improves the load forecasting accuracy as compared to the individual seasonal ARIMA model.

The performance of the individual seasonal ARIMA model and the combined model in forecasting the load demand is also evaluated by the mean comparison; the comparison result is shown in Figure 12, where group 1, group 2, and group 3 represent the actual load demand, the load demand forecasted by the individual seasonal ARIMA model and the load demand forecasted by the combined model respectively.

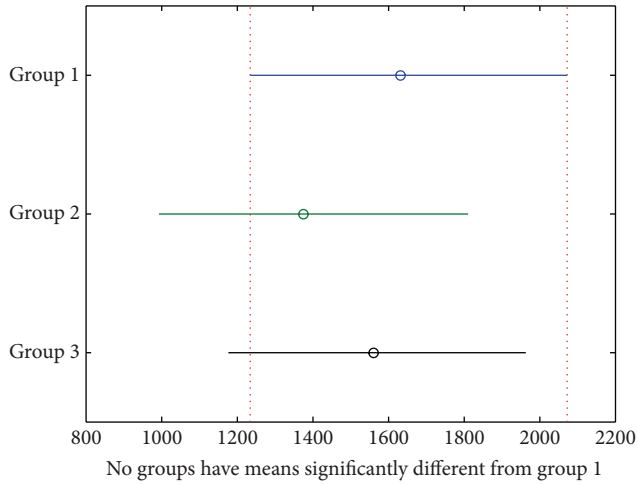


Figure 12: Mean multiple comparisons figure.

As shown, no groups have means significantly different from group 1, that is, there is no significant difference between the means of the actual load demand and the load demand forecasted by the individual seasonal ARIMA model, as well as the means between the actual load demand and the load demand forecasted by the combined model. However, group 3 occupies more common part with the actual load demand variation range than group 2; thus, the combined model performs better than the individual seasonal ARIMA model

in load demand forecasting; that is, the combined model improves the load demand forecasting accuracy as compared to the individual seasonal ARIMA model.

5. Conclusions

Different from usual combined forecasting models, a new strategy for STLF of using combined models is presented in this paper. As many sequences has periodic in real life, so these similar to the SARIMA model which can dig out the periodicity contained in the data is often used to predict and model the time series which have periodic. Secondly, this paper proposed by using the error sequence which SARIMA model predicted to predict the residual series of one day in the future, and by adding the residual series to the load value which got by the BP on the same day to improve the accuracy of the model. But the load prediction value residuals which were obtained by the SARIMA model do not have the same tendency or regularity; therefore, the choice of subsequent residual sequence prediction method should be careful. Considering that the neural network has a good effect for fitting of nonlinear function, this paper uses neural network model which can perfectly reflect the nonlinear relation between the input and output element to predict the subsequent residual sequence and did not use the regression or other model which has clear requirements for the form of the data; this further improves the accuracy of the prediction residuals. Furthermore, according to the characteristics of the model, this paper constructed a validity criterion which can measure the effectiveness of the model. At last, by using this combination method to the electricity load demand forecasting of South Australia, it appears that this combination method has a good effect in improving the prediction precision, because it is relative to the error in 15.98% which was predicted by a single SARIMA model, a hybrid model based on SARIMA, and neural network reduces the load predict error to 5.13%, and the validity criterion increases from 0.8402 to 0.9487. Simulation results demonstrate that the new strategy for STLF is effective in getting satisfying improvement of forecasting accuracy.

Acknowledgments

This work was supported by the Natural Science Foundation of P. R. of China (90912003, 61073193), the Key Science and Technology Foundation of Gansu Province (1102FKDA010), Natural Science Foundation of Gansu Province (1107RJZA188), and the Fundamental Research Funds for the Central Universities (lzujbky-2012-47, lzujbky-2012-48).

References

[1] H. M. Al-Hamadi and S. A. Soliman, "Short-term electric load forecasting based on Kalman filtering algorithm with moving window weather and load model," *Electric Power Systems Research*, vol. 68, no. 1, pp. 47–59, 2004.

[2] T. Senjyu, P. Mandal, K. Uezato, and T. Funabashi, "Next day load curve forecasting using hybrid correction method," *IEEE Transactions on Power Systems*, vol. 20, no. 1, pp. 102–109, 2005.

[3] B. Wang, N. L. Tai, H. Q. Zhai, J. Ye, J. D. Zhu, and L. B. Qi, "A new ARMAX model based on evolutionary algorithm and particle swarm optimization for short-term load forecasting," *Electric Power Systems Research*, vol. 78, no. 10, pp. 1679–1685, 2008.

[4] P. A. Mastorocostas, J. B. Theocharis, S. J. Kiartzis, and A. G. Bakirtzis, "A hybrid fuzzy modeling method for short-term load forecasting," *Mathematics and Computers in Simulation*, vol. 51, no. 3-4, pp. 221–232, 2000.

[5] S. E. Papadakis, J. B. Theocharis, and A. G. Bakirtzis, "A load curve based fuzzy modeling technique for short-term load forecasting," *Fuzzy Sets and Systems*, vol. 135, no. 2, pp. 279–303, 2003.

[6] J. F. Yang and J. Stenzel, "Short-term load forecasting with increment regression tree," *Electric Power Systems Research*, vol. 76, no. 9-10, pp. 880–888, 2006.

[7] H. M. Al-Hamadi and S. A. Soliman, "Short-term electric load forecasting based on Kalman filtering algorithm with moving window weather and load model," *Electric Power Systems Research*, vol. 68, no. 1, pp. 47–59, 2004.

[8] C. Q. Kang, X. Cheng, Q. Xia, Y. H. Huang, and F. Gao, "Novel approach considering load-relative factors in short-term load forecasting," *Electric Power Systems Research*, vol. 70, no. 2, pp. 99–107, 2004.

[9] S. Fan, L. N. Chen, and W. J. Lee, "Machine learning based switching model for electricity load forecasting," *Energy Conversion and Management*, vol. 49, no. 6, pp. 1331–1344, 2008.

[10] D. Srinivasan, "Evolving artificial neural networks for short term load forecasting," *Neurocomputing*, vol. 23, no. 1–3, pp. 265–276, 1998.

[11] J. F. Chen, W. M. Wang, and C. M. Huang, "Analysis of an adaptive time-series autoregressive moving-average (ARMA) model for short-term load forecasting," *Electric Power Systems Research*, vol. 34, no. 3, pp. 187–196, 1995.

[12] L. F. Amaral, R. C. Souza, and M. Stevenson, "A smooth transition periodic autoregressive (STPAR) model for short-term load forecasting," *International Journal of Forecasting*, vol. 24, no. 4, pp. 603–615, 2008.

[13] T. M. Choi, Y. Yu, and K. F. Au, "A hybrid SARIMA wavelet transform method for sales forecasting," *Decision Support Systems*, vol. 51, no. 1, pp. 130–140, 2011.

[14] E. Egrioglu, C. H. Aladag, U. Yolcu, M. A. Basaran, and V. R. Uslu, "A new hybrid approach based on SARIMA and partial high order bivariate fuzzy time series forecasting model," *Expert Systems with Applications*, vol. 36, no. 4, pp. 7424–7434, 2009.

[15] K. Y. Chen and C. H. Wang, "A hybrid SARIMA and support vector machines in forecasting the production values of the machinery industry in Taiwan," *Expert Systems with Applications*, vol. 32, no. 1, pp. 254–264, 2007.

[16] H. Liu, V. Chandrasekar, and G. Xu, "An adaptive neural network scheme for radar rainfall estimation form WSR-88D observations," *Journal of Applied Meteorology*, vol. 40, no. 11, pp. 2038–2050, 2001.

[17] L. Ke, G. Wenyan, S. Xiaoliu, and T. Zhongfu, "Research on the forecast model of electricity power industry loan based on GA-BP neural network," *Energy Procedia*, vol. 14, pp. 1918–1924, 2012.

[18] Q. Li, J. Y. Yu, B. C. Mu, and X. D. Sun, "BP neural network prediction of the mechanical properties of porous NiTi shape memory alloy prepared by thermal explosion reaction," *Materials Science and Engineering A*, vol. 419, no. 1-2, pp. 214–217, 2006.

[19] M. Li and W. Chen, "Application of BP neural network algorithm in sustainable development of highway construction projects," *Physics Procedia*, vol. 25, pp. 1212–1217, 2012.

[20] Y. H. Bao and J. Ren, "Wetland landscape classification based on the BP neural network in DaLinor lake area," *Procedia Environmental Sciences*, vol. 10, pp. 2360–2366, 2011.

[21] F. M. Tseng, H. C. Yu, and G. H. Tzeng, "Combining neural network model with seasonal time series ARIMA model," *Technological Forecasting and Social Change*, vol. 69, no. 1, pp. 71–87, 2002.

[22] S. L. Lai and W. L. Lu, "Impact analysis of September 11 on air travel demand in the USA," *Journal of Air Transport Management*, vol. 11, no. 6, pp. 455–458, 2005.

[23] F. M. Tseng and G. H. Tzeng, "A fuzzy seasonal ARIMA model for forecasting," *Fuzzy Sets and Systems*, vol. 126, no. 3, pp. 367–376, 2002.

[24] M. A. Mohandes, S. Rehman, and T. O. Halawani, "A neural networks approach for wind speed prediction," *Renewable Energy*, vol. 13, no. 3, pp. 345–354, 1998.

[25] L. Yu, S. Wang, and K. K. Lai, "Forecasting crude oil price with an EMD-based neural network ensemble learning paradigm," *Energy Economics*, vol. 30, no. 5, pp. 2623–2635, 2008.

[26] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[27] H. S. Hippert, C. E. Pedreira, and R. C. Souza, "Neural networks for short-term load forecasting: a review and evaluation," *IEEE Transactions on Power Systems*, vol. 16, no. 1, pp. 44–55, 2001.

[28] Y. D. Zhang and L. N. Wu, "Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network," *Expert Systems with Applications*, vol. 36, no. 5, pp. 8849–8854, 2009.

[29] C. X. J. Feng, C. G. Abhirami, A. E. Smith, and Z. G. S. Yu, "Practical guidelines for developing BP neural network models of measurement uncertainty data," *Journal of Manufacturing Systems*, vol. 25, no. 4, pp. 239–250, 2006.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 652083, 9 pages
http://dx.doi.org/10.1155/2013/652083

Research Article

Model for the Assessment of Seawater Environmental Quality Based on Multiobjective Variable Fuzzy Set Theory

Lina Ke and Huicheng Zhou

School of Hydraulic Engineering, Dalian University of Technology, Dalian 116024, China

Correspondence should be addressed to Lina Ke; kekesunny@163.com

Received 17 February 2013; Accepted 23 April 2013

Academic Editor: Yong Zhang

With the rapid development of marine economy industry, the activities for exploring and exploiting the marine resources are increasing, and there are more and more marine construction projects, which contribute to the growing trend of eutrophication and frequent occurrence of red tide. Thus, seawater quality has become the topic which the people generally cared about. The seawater quality evaluation could be considered as an analysis process which combined the evaluation indexes with certainty and evaluation factors with uncertainty and its changes. This paper built a model for the assessment of seawater environmental quality based on the multiobjective variable fuzzy set theory (VFEM). The Qingdao marine dumping site in China is taken as an evaluation example. Through the quantitative research of water-quality data from 2004 to 2008, the model is more reliable than other traditional methods, in which uncertainty and ambiguity of the seawater quality evaluation are considered, and trade the stable results as the final results of seawater quality evaluation, which effectively solved the impact of the fuzzy boundary of evaluation standard and monitoring error, is more suitable for evaluation of a multi-index, multilevel, and nonlinear marine environment system and has been proved to be an effective tool for seawater quality evaluation.

1. Introduction

Since the 1950s, marine environmental quality has been studied by domestic and foreign scholars in detail, and many methods for seawater quality evaluation are available, including the single factor index method [1–3], the fuzzy comprehensive evaluation method [4–6], the BP neural network method [7, 8], the grey clustering method [9], and the support vector machine (SVM) [10], among others. Each of these methods has their own advantages and disadvantages. Seawater quality assessment combines certain evaluation indices and criteria with uncertain evaluation factors and is a complicated process coupling the effects of multiple factors and their content changes. The assessment indices are often variable with fuzzy uncertainty. Commonly, traditional methods of water quality assessment treat the evaluation standard and a reference as a point [11, 12], and hence the application of these methods have some limitations. In recent years, application of fuzzy comprehensive evaluation becomes more and more popular in real cases [13–17], for solving the limitation problem in a classical mathematical model that describes uncertainty with *either-or* only. In fuzzy sets theory, we use *this and that* to describe the problems in uncertainty [18], we could solve the problem of fuzzy boundary effectively and monitor errors affecting the evaluation results in environment evaluation. However, the fuzzy comprehensive evaluation method has some uncertainties, and the model is difficult to perform self-adjustment and self-verification. Therefore, to evaluate seawater environment quality scientifically with feasibility, we put forward a new model for assessment of seawater environment quality based on variable fuzzy recognition model and applied it in the assessment of seawater quality status of the Qingdao marine dumping site in China from 2004 to 2008. The results demonstrate that with the method, we can reasonably determine the relative membership degree and the relative membership function of evaluation indices in all levels or intervals of applicable standard and assess the grade of water quality more

realistically and reasonably, which would be important as a new concept and reference for improving the performance of seawater assessment in China and, potentially, in the world.

2. Materials and Methods

2.1. Variable Fuzzy Model for Seawater Quality Evaluations. The comprehensive seawater quality level of an object u is identified according to the standard of m indices and c grades. In the standard interval of each index at level-h, point M_{ih} is sure to exist, and thus, the relative membership degree of M_{ih} to level-h is equal to one. The variable M_{ih} is defined as the standard value of index-i at level-h.

(1) According to m indices and c grades, determine the attraction domain matrix, $I_{ab} = ([a_{ih}, b_{ih}])$, of the variable set for seawater quality evaluation, range domain matrix, $I_{cd} = ([c_{ih}, d_{ih}])$, and M_{ih} point value matrix. M_{ih} can be determined according to the following formula:

$$M_{ih} = \frac{c-h}{c-1} a_{ih} + \frac{h-1}{c-1} b_{ih}. \tag{1}$$

If $h = 1$, then $M_{i1} = a_{i1}$, if $h = c$, then $M_{ic} = b_{ic}$, and if $h = (c+1)/2$, then $M_{il} = (a_{il} + b_{il})/2$.

(2) Calculating the relative membership degree matrix:

when x falls to the left side of point M_{ih}, the relative membership degree model is calculated as follows:

$$\underset{\sim}{\mu A}\left(x_{ij}\right) h = 0.5 * \left[1 + \left(\frac{x-a}{M_{ih}-a}\right)^{\beta}\right]; \quad x \in [a, M_{ih}],$$
$$\underset{\sim}{\mu A}\left(x_{ij}\right) h = 0.5 * \left[1 - \left(\frac{x-a}{c-a}\right)^{\beta}\right]; \quad x \in [c, a]. \tag{2}$$

When x falls to the right side of point M_{ih}, the relative membership degree model is calculated as follows:

$$\underset{\sim}{\mu A}\left(x_{ij}\right) h = 0.5 * \left[1 + \left(\frac{x-b}{M_{ih}-b}\right)^{\beta}\right]; \quad x \in [M_{ih}, b],$$
$$\underset{\sim}{\mu A}\left(x_{ij}\right) h = 0.5 * \left[1 - \left(\frac{x-b}{d-b}\right)^{\beta}\right]; \quad x \in [b, d] \tag{3}$$

in which $\beta = 1$ and the function model is a linear function.

(3) The comprehensive relative membership degree vector of sample j to level h is calculated as follows:

$$j\mu' h = \frac{1}{1 + \left\{\sum_{i=1}^{m} \left[wi\left(1 - \mu\underline{A}\left(xij\right) h\right)\right]^{p} / \sum_{i=1}^{m} \left(wi\mu\underline{A}\left(xij\right) h\right)^{p}\right\}^{a/p}}, \tag{4}$$

where α is the model optimization criteria parameter, p is the distance parameter, and α and p can have 4 combinations given as follows.

(a) When $a = 1$, $p = 1$, the model is the fuzzy comprehensive evaluation model:

$$v_h(u) = \sum_{i=1}^{m} w_i u_{ih}(u). \tag{5}$$

(b) When $a = 1$, $p = 2$, the model is the TOPSIS model:

$$v_h(u) = \frac{1}{1 + \sqrt{\sum_{i=1}^{m} \left[w_i\left(1 - u_{ih}(u)\right)\right]^2 / \sum_{i=1}^{m} \left[w_i u_{ih}(u)\right]^2}}. \tag{6}$$

(c) When $a = 2$, $p = 1$, the model is the activation function model of a neuron:

$$v_h(u) = \frac{1}{1 + \left[\left(1 - \sum_{i=1}^{m} w_i u_{ih}(u)\right) / \sum_{i=1}^{m} w_i u_{ih}(u)\right]^2}. \tag{7}$$

(d) When $a = 2$, $p = 2$, the model is the fuzzy optimization model:

$$v_h(u) = \frac{1}{1 + \sum_{i=1}^{m} \left\{w_i\left[1 - u_{ih}(u)\right]\right\}^2 / \sum_{i=1}^{m} \left[w_i u_{ih}(u)\right]^2}. \tag{8}$$

In the conditions of fuzzy concept classification, using the principle of the maximum membership degree to identify the level of an object in assessment for seawater quality can easily produce an incorrect final result. The level-characteristic value proposed in the equation by Chen and Hu [19] can fully express the whole distribution characteristics of h and $v_h(u)$, can make best information of the relative membership degree of level variables h to a certain level, and can be used as the criterion of the variable fuzzy set theory to judge, identify, and determine the level:

$$H(u) = \sum_{h=1}^{c} v^0(u) h. \tag{9}$$

2.2. Determination of Weight

2.2.1. Determination of the Experience Weight w_1 by the Nonstructural Decision-Making Fuzzy Theory Model. The limitation of the AHP model of putting a binary comparison of the element attributes into the comparison of the importance is analyzed, and a nonstructural decision-making fuzzy theory model was presented by Professor Chen [20]. The two adjectives are used to describe the fuzzy boundary values of 0.5 and 1.0 according to their degree of importance, which are equally important and incomparably important, and were further divided into 11 mood operators: "equally," "slightly," "somewhat," "rather," "obviously," "remarkably," "very," "extra," "exceedingly," "extremely" and "incomparably" which represent a different fuzzy scale (Table 1). The relative membership degree of the objective to the importance of the fuzzy concepts is calculated to attain the weight of the objective set. The specific calculation steps are as follows.

(1) An objective set $P = \{p_1, p_2, \ldots, p_m\}$ for comparing the importance and build a binary importance sequence matrix E according to the degree of importance of the target elements.

(2) Arrange the sum of E matrix lines from large to small, and obtain the importance sequence of the objective set.

(3) According to the matrix E, make a binary importance judgment by experience.

(4) By the relationships between different mood operators and fuzzy scales, calculate the relative membership degree of the objective to the importance of the fuzzy concepts and attain the nonnormalized weight vector w_1 according to formula (4):

$$\varphi_{1i} = \frac{1 - \beta_{1i}}{\beta_{1i}} \tag{10}$$

β_{1i} is the binary importance fuzzy scale value between objective 1 and objective i; φ_{1i} is the relative membership degree of objective i to the importance.

2.2.2. Determination of the Objective Weight w_2 by the Standard Level Method of Water Quality. Consider

$$w_2 = \begin{cases} \dfrac{x_i}{\overline{S}_i}, & x_i > \overline{S}_i, \\ 1, & x_i \le \overline{S}_i, \end{cases} \quad \overline{S}_i = \frac{1}{m}\sum_{j=1}^{m} S_{ij}, \quad m = 4. \tag{11}$$

For DO

$$w_2 = \begin{cases} 1, & x_i \ge \overline{S}_i \\ \dfrac{\overline{S}_i}{x_i}, & x_i < \overline{S}_i. \end{cases} \tag{12}$$

In the above formula, x_i is the measured value of the ith pollution factor; S_i is the standard seawater quality value of the jth pollution factor at the ith level; $\overline{S}_i$ is the average seawater quality value of four levels of the ith pollution factor; n is the number of pollution factors; and m is the level number in the seawater quality standard.

2.2.3. Comprehensive Weight. The weight determined by the nonstructural decision-making fuzzy theory model is an experience weight and can easily be influenced by anthropic factors. During the evaluation process, the effect of some indices may be overstated or reduced; the weight determined by the standard level method is a mathematical weight. The relative importance of some indices has not been considered. The two types of weight methods each have certain advantages and limitations.

Here referring to this paper [22], the combination weight is adopted to improve the reliability of weight setting, which combined the binary fuzzy clustering weight with the standard level weight, and the calculation formula is as follows:

$$w = \alpha w_1 + (1 - \alpha)\, w_2. \tag{13}$$

In this formula, w is the combinational weight, w_1 is the experience weight determined by the non-structural decision-making fuzzy model, w_2 is the objective weight determined by the standard level model, α is the sensitivity coefficient with values of $0 < a < 1$. In general, the range of a is 0.5~0.7. To reinforce the importance of the combinational weight, here an intermediate value 0.6 was obtained and was regarded as the sensitivity coefficient of the combinational weight.

3. Results and Discussion

The Qingdao marine dumping site was one of the first marine dumping sites for Category III dredged materials specified by the State Oceanic Administration and approved by the State Council since the implementation of the *Regulations of the People's Republic of China on Control over Dumping of Wastes in the Ocean*. The dumping site is located in the southeast of the Jiaozhou Bay estuary, 6.7 km from Qingdao, and its area is about 7 km^2, extending between 120°18′00″ and 120°20′00″ east longitude and from 35°59′24″ to 35°58′39″ north latitude. This site receives dredged materials primarily from Qingdao port, other small ports, and navigation channels.

From November 1986 to 2009, the dredged materials dumped at this site exceeded a volume of 8.00×10^7 m^3. This sea area is close to the navigation channel, the aquaculture area, and the holiday resort. It is the ecologically sensitive area that is also important for economic development. Therefore, it is important to accurately and timely evaluate the current environmental conditions at the dumping site for preventing ocean dumping from damaging the ecological environment, marine resources, and the submarine landform.

To facilitate comparisons, this study utilized monitoring data (Table 2) of the seawater quality at the Qingdao marine dumping site [23]. In view of the pollution conditions of dredged materials and the present seawater pollution situation at the dumping site [23, 24], nine evaluation factors were selected, that is, COD, oils, DO, inorganic nitrogen, PO_4-P, Cu, Pb, Zn and Cd for the index standards, please see the *Seawater Quality Standard* (GB 3097-1997). The variable fuzzy model was adopted to evaluate the situation of seawater quality at the Qingdao marine dumping site.

The data from 14 monitoring points (Table 2) in the Qingdao marine dumping site were used to validate the variable fuzzy model. The characteristic value matrix and the index standard value matrix of the seawater quality are established below according to the *Seawater Quality Standard* (GB 3097-1997) and the seawater quality monitoring data of the 14 sampling points in the Qingdao dumping site [23], that is, x and y:

$$x = \begin{Bmatrix} 0.655 & 0.051 & 7.678 & 108.500 & 9.050 & 3.183 & 1.553 & 41.425 & 0.176 \\ 0.705 & 0.049 & 7.620 & 121.000 & 6.350 & 4.393 & 1.940 & 73.767 & 0.208 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0.725 & 0.023 & 7.635 & 97.000 & 6.250 & 3.660 & 2.678 & 32.542 & 0.187 \end{Bmatrix}^T,$$

$$y = \begin{Bmatrix} [0,2] & [0,0.05] & [6,12] & [0,200] & [0,15] & [0,5] & [0,1] & [0,20] & [0,1] \\ [2,3] & [0,0.05] & [5,6] & [200,300] & [15,30] & [5,10] & [1,5] & [20,50] & [1,5] \\ [3,4] & [0.05,0.30] & [4,5] & [300,400] & [15,30] & [10,50] & [5,10] & [50,100] & [5,10] \\ [4,5] & [0.30,0.50] & [3,4] & [400,500] & [30,45] & [10,50] & [10,50] & [100,500] & [5,10] \\ (5,\infty) & (0.5,1) & [0,3) & (500,1000) & (45,\infty) & (50,\infty) & (50,\infty) & (500,\infty) & (10,\infty) \end{Bmatrix}^T. \quad (14)$$

In reference to the standard seawater quality value and the actual seawater quality conditions at the Qingdao dumping site, the attraction domain matrix, the range domain matrix, and the M_{ih} matrix of the variable fuzzy seawater quality evaluation model were determined. Here, for the oil, PO_4, Cu, and Cd, the standard values of the adjacent level are the same, so the mean value of the two levels was used in the actual classification to further divide the adjacent index levels. For example, for the PO_4 index, the standard value of Grade II and Grade III is 0.015~0.030 mg/L; to facilitate the evaluation, the interval of Grade II was taken as 0.015~0.0225 mg/L, while the interval of Grade III was taken as 0.0225~0.030 mg/L. Practice has proved that it has no influence on the evaluation results. M_{ih} can be determined by formula (1).

Therefore, the respective attraction domain matrix, the range domain matrix, and the M_{ih} matrix of the variable fuzzy seawater quality evaluation model are as follows:

$$I_{ab} = \begin{bmatrix} [0,2] & [0,0.025] & [6,7.768] & [0,200] & [0,15] & [0,5] & [0,1] & [0,20] & [0,1] \\ [2,3] & [0.025,0.05] & [5,6] & [200,300] & [15,22.5] & [5,10] & [1,5] & [20,50] & [1,5] \\ [3,4] & [0.05,0.3] & [4,5] & [3000,400] & [22.5,30] & [10,30] & [5,10] & [50,100] & [5.7.5] \\ [4,5] & [0.3,0.5] & [3,4] & [400,500] & [30,45] & [30,50] & [10,50] & [100,500] & [7.5,10] \\ [5,6] & [0.5,0.75] & [2,3] & [500,600] & [45,60] & [50,70] & [50,90] & [500,900] & [10,14] \end{bmatrix}^T,$$

$$I_{cd} = \begin{bmatrix} [0,3] & [0,0.05] & [5,7.678] & [0,300] & [0,22.5] & [0,10] & [0,5] & [0,50] & [0,5] \\ [0,4] & [0,0.3] & [4,7.678] & [0,400] & [0,30] & [0,30] & [0,10] & [0,100] & [0,7.5] \\ [2,5] & [0.025,0.5] & [3,6] & [200,500] & [15,45] & [5,50] & [1,50] & [20,500] & [1,10] \\ [3,6] & [0.05,0.75] & [2,5] & [300,600] & [22.5,60] & [10,70] & [5,90] & [50,900] & [5,14] \\ [4,6] & [0.3,0.75] & [2,4] & [400,600] & [30,60] & [30,70] & [10,90] & [100,900] & [7.5,14] \end{bmatrix}^T, \quad (15)$$

$$M_{ih} = \begin{bmatrix} 0 & 0 & 7.678 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2.25 & 0.03125 & 5.75 & 225 & 16.875 & 6.25 & 2 & 27.5 & 2 \\ 3.5 & 0.175 & 4.5 & 350 & 26.25 & 20 & 7.5 & 75 & 6.25 \\ 4.75 & 0.45 & 3.25 & 475 & 41.25 & 45 & 40 & 400 & 9.375 \\ 6 & 0.75 & 2 & 600 & 60 & 70 & 90 & 900 & 14 \end{bmatrix}.$$

If x_{ij} is on the left side of M_{ih}, select formula (2) to calculate the relative membership degree, otherwise select formula (3) to calculate the relative membership degree.

Based on the consideration of expert opinions, the numerous studies available in the literature [25–29], and the corresponding relationship between tone operators and the fuzzy scale values given in Table 1, the two-dimensional importance comparison matrix of 9 indicators from large to small at a consistent scale β can be determined, and the sum of the fuzzy measure values is as follows:

$$w = \left(\sum_{t=1}^{m} \beta_{1t}, \sum_{t=1}^{m} \beta_{2t}, \ldots, \sum_{t=1}^{m} \beta_{mt} \right) = (1.99, 3.4, 1.07, 2.58, 1.79, 2.09, 6.86, 5.14, 1.09). \quad (16)$$

By normalizing the fuzzy measure values matrix, we obtained the weight of 9 indexes of seawater quality evaluation:

$$W_1 = (0.077, 0.131, 0.041, 0.099, 0.069, 0.080, 0.264, 0.198, 0.042). \quad (17)$$

Using formulas (11) and (12), the weight (w_2) was obtained as follows:

$$W_2 = (0.110, 0.109, 0.107, 0.122, 0.112, 0.113, 0.109, 0.109, 0.109). \quad (18)$$

According to formula (13), we could obtain the combination weight of 9 indices:

$$W = (0.090, 0.122, 0.067, 0.108, 0.086, 0.093, 0.202, 0.162, 0.069). \quad (19)$$

Using (4)–(9) to calculate the characteristic level value of sample at the time of $a = 1$, $p = 1$; $a = 1$, $p = 2$; $a = 2$, $p = 1$; $a = 2$, $p = 2$, and then the final evaluation results were obtained according to the following formula (20), as shown in Table 3:

$$\begin{array}{ll} c - 0.25 \le H_j \le c & \text{Corresponds to level } c \\ h - 0.25 < H_j \le h + 0.25 & \text{Corresponds to level } h, \\ & h = 1, 2, \ldots, c-1, \\ h + 0.25 < H_j < h + 0.75 & \text{Between level } h \text{ and } h+1, \\ & \text{tending toward one of the} \\ & \text{two levels.} \end{array} \quad (20)$$

As shown in Table 3, the evaluation results of 14 sampling points based on the variable fuzzy evaluation model are more consistent with the evaluation results based on a BP network, fuzzy comprehensive evaluation method, and fuzzy genetic neural network method, and only slight differences between the evaluation results of the variable fuzzy evaluation model and the results of other models were found in few sampling points. With the variation of (a, p) parameters in variable fuzzy recognition model, the characteristic level value of four different mathematical models at the sampling points remained within a small fluctuation range. Furthermore, compared with other evaluation methods, the variable fuzzy evaluation model can provide a more accurate position of the water quality level. The conditions of seawater quality at sampling points can be accurately differentiated according to the characteristic level values. The evaluation results by the variable fuzzy evaluation method at sampling points 2 and 5 are slightly inconsistent with the results by the other methods. Now, differences between variable fuzzy recognition model and fuzzy comprehensive evaluation model at those points of discrepancy are analyzed further.

For sampling point 2, the evaluation result based on the variable fuzzy evaluation model is between I and II, tending toward II, while the result based on the BP network and fuzzy genetic neural network is Grade III, and the result of fuzzy comprehensive evaluation method is Grade II. According to analyzing the original data of point 2, the values of COD, DO, inorganic nitrogen, PO_4-P, Cu and Cd are 0.705 mg/L, 7.620 mg/L, 121 μg/L, 6.35 μg/L, 4.393 μg/L and 0.208 μg/L, respectively, which fall within the range of Grade I; the concentrations of Pb and Cd are 1.94 μg/L and 73.767 μg/L, respectively, which fall within the range of Grade II; and only the content of oil, is 0.049 mg/L, which is close to the Grade III standard, but still falls within the range of Grade I or Grade II. Based on the actual seawater quality conditions at the Qingdao dumping site, oil, Pb and Zn strongly influence the seawater quality of Qingdao dumping site. The weights of these indices are 0.122, 0.202 and 0.162, respectively, accounting for 48.6% of the total weight, so the classification result of between Grade I and Grade II, tending toward Grade II, is reasonable. The average characteristic level value by the variable fuzzy evaluation model was 1.66, and this evaluation result is more credible than the results of other models, more consist with the actual conditions of point 2. The results of BP network, fuzzy genetic neural network, and fuzzy comprehensive valuation method were Grade II or Grade III, overemphasizing a few pollution heavy factors and causing the evaluation results too high. Similarly, the situation of point 5 is similar to point 2.

Analyzing the causes of the differences between BP network, fuzzy comprehensive evaluation method, fuzzy genetic neural network evaluation method, and variable fuzzy evaluation model in Table 3, fuzzy comprehensive evaluation method determines the water quality level on the basis of the maximum membership degree principle, thus, when the membership degree of the water quality at a level site does not differ considerably from the adjacent level (e.g., the membership degrees attached to Grade I, Grade II, Grade III, and Grade IV are 0, 0.34, 0.37 and 0.29, resp.), many important information may be lost, therefore, which often leads to the final evaluation results incorrect. BP neural network and genetic neural network adopt feed-forward networks, and their network structure is determined by experience, which always shows a great deal of randomness. As a result, it is sometimes possible to obtain a very small global optimum, causing the misjudgment of the final result.

This paper combines the monitoring values of seawater quality indicators with the national standard to build a seawater quality evaluation model in variable fuzzy recognition model, to deal with greater subjectivity problems of water quality evaluation with limited data. To a certain extent, we will measure the ambiguity and uncertainty of water quality evaluation objectively and increase the credibility of the rank of a sample point [30].

The method of the seawater quality evaluation model based on variable vague set theory in this paper is able to combine linear model with nonlinear model through changes of the variable model parameters (a, p). This method not only avoids the instability of evaluation results caused by the single model but also can reflect the difference of the membership degree of the adjacent water level and finally take the stable level value as the final evaluation results of seawater environment. It can arrange the situation of water environment quality of various samples and clearly determine the water quality status that makes the evaluation results more trustworthy. According to the linear or nonlinear feature of the evaluation objects, seawater quality evaluation based on variable fuzzy recognition model can select variable models with the changes of variable model parameters (a, p) and combine the linear features with the nonlinear features of the evaluation objective, which weakened the influence of the index weight on the final results. It makes the model more flexible and accurate and avoids "over-fitting" because the neural network structure is too large. It also ensures that the model has better generalization ability and predictive ability.

However, for variable fuzzy recognition model, the rational weight setting is still an important factor to determine the reliability of the evaluation results. Due to cross-iteration of the parameters of variable fuzzy recognition model and the variability of indicators weight vector, it is very important to reasonably set the indicator weight according to the nature of a real case and the importance of actual decision objective in practice.

We use weight-determination method of the comprehensive weight which combines the subjective nonstructural

Table 1: Corresponding relationship between the tone operator and fuzzy scale values.

Tone operator	Equally	Slightly	Somewhat	Rather	Obviously	Remarkably
Fuzzy scale	0.50	0.55	0.60	0.65	0.70	0.75
Tone operator	Very	Extra	Exceedingly	Extremely	Incomparably	
Fuzzy scale	0.80	0.85	0.90	0.95	1.00	

Table 2: Monitoring results regarding seawater quality in the Qingdao dumping site in 2003.

Sampling point	COD	OIL	DO	Inorganic nitrogen	PO_4-P	Cu	Pb	Zn	Cd
Q1	0.655	0.051	7.678	108.500	9.050	3.183	1.553	41.425	0.176
Q2	0.705	0.049	7.620	121.000	6.350	4.393	1.940	73.767	0.208
Q3	0.680	0.027	7.612	131.000	8.100	4.768	4.133	38.775	0.378
Q4	0.735	0.034	7.567	102.500	8.150	4.023	1.228	29.725	0.130
Q5	0.700	0.034	7.502	101.500	6.550	5.982	2.493	57.383	0.226
Q6	0.750	0.029	7.553	88.000	6.550	3.475	2.298	49.400	0.286
Q7	0.935	0.100	7.635	128.500	6.300	3.265	2.867	45.925	0.193
Q8	0.835	0.049	7.635	122.000	7.900	4.393	1.522	33.958	0.124
Q9	0.600	0.062	7.519	97.500	5.950	3.143	2.317	39.675	0.201
Q10	0.585	0.018	7.572	92.000	5.900	4.957	1.253	27.350	0.143
Q11	0.640	0.024	7.594	97.000	6.850	3.407	2.218	35.150	0.154
Q12	0.670	0.028	7.676	84.500	6.100	4.517	2.192	41.200	0.188
Q13	0.655	0.021	7.517	83.500	4.850	6.967	2.670	32.792	0.248
Q14	0.725	0.023	7.635	97.000	6.250	3.660	2.678	32.542	0.187

The units for COD, oil and DO are mg/L; the units for inorganic nitrogen, PO_4-P, Cu, Pb, Zn and Cd are μg/L.

decision-making fuzzy weights with the objective standard level weights and provide a reference for weight setting. In the future, how to set index weight more reasonably in actual marine environment evaluation and how to determine the level of seawater quality according to the characteristic level values will be studied to improve the application of multitarget variable fuzzy recognition model for seawater quality evaluation. Each water quality evaluation method owns different emphases. Variable fuzzy recognition model can combine the linear features with the nonlinear features of the evaluation objective and provide a reference for the multi-objective decision solutions and can be promoted for the evaluation of other multi-index, multilevel, and nonlinear systems.

Using the monitoring data of the seawater quality at Qingdao dumping site (1985–2003), the comprehensive situation of Qingdao dumping site (1985–2003) is evaluated by variable fuzzy comprehensive evaluation model. The evaluation results are shown in Table 4. From Table 4, we can see that the results of seawater quality in the Qingdao dumping site all satisfy the standard of Grade II specified by the Sea Water Quality Standard (GB3097-1997). In 1997 and 2003, the seawater quality of Qingdao dumping site was rather poor and the characteristic level values were 1.59 and 1.64, respectively, which is within the range of Grade II. In other years, the water quality of Qingdao dumping site was satisfactory and met the standard of Grade I. From 1998 to 2003, the order of seawater quality was as follows: 2002 > 2000 > 1985 > 1991 > 1997 > 2003. Overall, the seawater quality of the Qingdao dumping area presents a drop-rise-drop trend. From 1985, the seawater quality of the Qingdao dumping area started to decline. In 1997, it reached a low point, and the characteristic level value of the Qingdao dumping area was only 1.59, which only corresponded to the standard of Grade II. From 2000, the seawater quality of the Qingdao dumping area tended to improve. However, in 2003, the seawater quality began to deteriorate again, and the characteristic level value of the Qingdao dumping area in 2003 was 1.64, which was the worst among the seawater quality situation of the examined years. The major impact factors affecting the seawater quality of the Qingdao dumping area are heavy metals Pb, Zn and oils; Pb is especially the most serious factor for the seawater quality of the Qingdao dumping area. From 1985 to 2003, the content of Pb is always within the range of Grade II or Grade III; only in 2000, the content of Pb is satisfied with the standard of Grade I. Secondly, the more serious pollutants are oils and Zn. In 1997, the content of oil exceeded the standard of Grade II 0.49 times. The dumped dredged materials are the primary cause affecting the seawater quality of this area, and the wastes dumped into the sea are mainly the dredged materials of Grade III. In these dredged materials, there are a number of pollutants which may affect the marine environment of the Qingdao dumping area, such as Pb, Zn, oils, and the compounds of other elements. These pollutants are transformed into harmful substances through chemical reactions and biological reactions, which affected the seawater environment of the Qingdao dumping area. Nevertheless, compared with the environmental conditions of the time when the dumping area was delimited, the environmental conditions of the dumping area have essentially

Table 3: Comparison of the comprehensive seawater quality evaluation results.

Point Number	Variable fuzzy evaluation method						Fuzzy comprehensive evaluation	BP neural Network	Fuzzy genetic neural Network
	$\alpha = 1$ $p = 1$	$\alpha = 1$ $p = 2$	$\alpha = 2$ $p = 1$	$\alpha = 2$ $p = 2$	Average value	Evaluation grade			
Q1	1.464	1.593	1.309	1.471	1.46	Between I and II, tending toward I	Grade II	Grade II	Grade II
Q2	1.645	1.894	1.327	1.762	1.66	Between I and II, tending toward II	Grade II	Grade III	Grade III
Q3	1.597	1.761	1.504	1.694	1.64	Between I and II, tending toward II	Grade II	Grade II	Grade II
Q4	1.354	1.427	1.231	1.357	1.34	Between I and II, tending toward I	Grade I	Grade II	Grade I
Q5	1.614	1.823	1.465	1.707	1.65	Between I and II, tending toward II	Grade II	Grade III	Grade III
Q6	1.492	1.705	1.304	1.553	1.51	Between I and II, tending toward II	Grade II	Grade II	Grade II
Q7	1.636	1.802	1.491	1.716	1.66	Between I and II, tending to II	Grade II	Grade II	Grade II
Q8	1.451	1.539	1.370	1.497	1.46	Between I and II, tending toward I	Grade I	Grade II	Grade II
Q9	1.512	1.645	1.399	1.586	1.54	Between I and II, tending toward II	Grade II	Grade II	Grade II
Q10	1.273	1.352	1.124	1.229	1.24	Grade I	Grade I	Grade I	Grade I
Q11	1.402	1.507	1.309	1.510	1.43	Between I and II, tending toward I	Grade I	Grade II	Grade II
Q12	1.463	1.611	1.348	1.538	1.49	Between I and II, tending toward I	Grade I	Grade II	Grade II
Q13	1.476	1.552	1.453	1.602	1.52	Between I and II, tending toward II	Grade I	Grade II	Grade II
Q14	1.421	1.529	1.346	1.557	1.46	Between I and II, tending toward I	Grade I	Grade II	Grade II

Table 4: Results of the variable fuzzy comprehensive evaluation of seawater quality in the Qingdao dumping area (1985–2003).

Year	COD	OIL	DO	Inorganic nitrogen	PO_4-P	Cu	Pb	Zn	Cd	Variable fuzzy evaluation model	
										Average	Evaluation grade
1985	0.45	0.038	8.73	1.26	0.31	0.49	4.13	2.715	0.13	1.30	I
1991	0.89	0.015	7.86	25.9	6.82	0.36	5.7	9.8	0.17	1.33	I
1997	0.65	0.0745	7.76	77.3	14.96	4.74	1.28	42.5	0.07	1.59	II
2000	1.54	0.021	9.17	94.42	15.75	2.98	0.86	13.7	0.18	1.14	I
2002	0.48	0.024	7.99	65.7	6.5	2.04	1.49	11.16	0.28	1.11	I
2003	0.72	0.039	7.64	102	6.9	4.30	2.25	41.54	0.19	1.64	II

remained unchanged. The benthic community structure in the dumping area has not undergone any significant changes due to the dumping of dredged materials. The dumped wastes of the Qingdao dumping area are somewhat controlled within the predicted management, and the dumping of dredged materials has no impact on the offshore marine environment; the basic function of the ocean dumping area is still to be maintained.

4. Conclusion

We build a seawater environmental quality assessment model based on variable fuzzy recognition model, in which uncertainty and ambiguity of the seawater quality evaluation are considered, and the monitoring values of seawater quality evaluation indicators and the standard value of seawater quality are combined. Through the application of this model for the Qingdao marine dumping site water quality evaluation and comparison in performance with other models, the model is proved to be an effective tool for seawater quality evaluation. The following conclusions can be drawn.

(1) Seawater environmental quality assessment model based on variable fuzzy recognition model considers the uncertainty and ambiguity involved in the seawater quality evaluation, combines monitoring values of seawater quality evaluation indicators and

the standard value of seawater quality, and selects the right variable model of the different parameters according to the linear or nonlinear features of the evaluation objects. Therefore, the method is more flexible than other models, and the evaluation results are more stable. It can arrange the situation of water environment quality of various samples and clearly determine the water quality status that makes the evaluation results more credible; therefore, it is more suitable for evaluation of a multi-index, multi-level, and nonlinear marine environment system.

(2) Different indices in different seawater environments have different effects on the evaluation results of seawater quality. In this paper, weight-determination method of the comprehensive weight which combines the subjective nonstructural decision-making fuzzy weights with the objective standard level weights and provides a reference for weight setting. When the evaluation model is applied to other applications, it is necessary to set the index weight reasonably according to the specific conditions of seawater quality evaluation.

(3) In the future, how to determine the level of seawater quality according to the characteristic level values is an important part, which needs to be improved in the application of a multitarget variable fuzzy recognition model for seawater quality evaluation.

References

[1] P. M. Chapman and F. Wang, "Assessing sediment contamination in estuaries," *Environmental Toxicology and Chemistry*, vol. 20, no. 1, pp. 3–22, 2001.

[2] A. N. Papanicolaou, A. Bdour, N. Evangelopoulos, and N. Tallebeydokhti, "Watershed and instream impacts on the fish population in the South Fork of the Clearwater River, Idaho," *Journal of the American Water Resources Association*, vol. 39, no. 1, pp. 191–203, 2003.

[3] N. Fierer, J. L. Morse, S. T. Berthrong, E. S. Bernhardt, and R. B. Jackson, "Environmental controls on the landscape-scale biogeography of stream bacterial communities," *Ecology*, vol. 88, no. 9, pp. 2162–2173, 2007.

[4] K. O. Adebowale, F. O. Agunbiade, and B. I. Olu-Owolabi, "Fuzzy comprehensive assessment of metal contamination of water and sediments in Ondo Estuary, Nigeria," *Chemistry and Ecology*, vol. 24, no. 4, pp. 269–283, 2008.

[5] J. Wang, X. Lu, J. Tian, and M. Jiang, "Fuzzy synthetic evaluation of water quality of Naoli river using parameter correlation analysis," *Chinese Geographical Science*, vol. 18, no. 4, pp. 361–368, 2008.

[6] X. L. Wang, T. Li, H. Yang et al., "Fuzzy comprehensive-quantifying assessment in analysis of waterquality: a case study in Lake Honghu, China," *Environmental Engineering Science*, vol. 26, no. 2, pp. 451–458, 2009.

[7] L. Z. Xu, X. P. Ma, Z. Lin et al., "Assessment method for water quality by multi-source information fusion based on BP neural networks and evidence theory," *Dynamics of Continuous Discrete and Impulsive Systems B*, vol. 2, pp. 520–523, 2005.

[8] Z. H. Guo, J. Wu, H. Y. Lu, and J. Z. Wang, "A case study on a hybrid wind speed forecasting method using BP neural network," *Knowledge-Based Systems*, vol. 24, no. 7, pp. 1048–1056, 2011.

[9] C. M. Mi, S. F. Liu, Y. Dang et al., "Study on 2-tuple linguistic assessment method based on grey clustering," *Journal of Grey System*, vol. 19, no. 3, pp. 257–268, 2007.

[10] J. Cao, H. Hu, S. Qian, and K. Xu, "Research on aggregative index number method in water quality assessment based on SVM," in *Proceedings of the 9th International Conference on Electronic Measurement and Instruments (ICEMI '09)*, pp. 4787–4791, August 2009.

[11] L. Cea, M. Bermúdez, and J. Puertas, "Uncertainty and sensitivity analysis of a depth-averaged water quality model for evaluation of Escherichia coli concentration in shallow estuaries," *Environmental Modelling & Software*, vol. 26, no. 12, pp. 1526–1539, 2011.

[12] M. T. Bhatti and M. Latif, "Assessment of water quality of a river using an indexing approach during the low-flow season," *Irrigation and Drainage*, vol. 60, no. 1, pp. 103–114, 2011.

[13] N. B. Chang, H. W. Chen, and S. K. Ning, "Identification of river water quality using the fuzzy synthetic evaluation approach," *Journal of Environmental Management*, vol. 63, no. 3, pp. 293–305, 2001.

[14] C. Ren, C. Li, K. Jia, S. Zhang, W. Li, and Y. Cao, "Water quality assessment for Ulansuhai Lake using fuzzy clustering and pattern recognition," *Chinese Journal of Oceanology and Limnology*, vol. 26, no. 3, pp. 339–344, 2008.

[15] J. H. Kim, C. M. Choi, S. B. Kim, and S. K. Kwun, "Water quality monitoring and multivariate statistical analysis for rural streams in South Korea," *Paddy and Water Environment*, vol. 7, no. 3, pp. 197–208, 2009.

[16] L. Liu, J. Zhou, X. An, Y. Zhang, and L. Yang, "Using fuzzy theory and information entropy for water quality assessment in Three Gorges region, China," *Expert Systems with Applications*, vol. 37, no. 3, pp. 2517–2521, 2010.

[17] D. R. Pathak and A. Hiratsuka, "An integrated GIS based fuzzy pattern recognition model to compute groundwater vulnerability index for decision making," *Journal of Hydro-Environment Research*, vol. 5, no. 1, pp. 63–77, 2011.

[18] S. Y. Chen, M. Li, and S. Y. Wang, "Rationality analysis and application test of variable fuzzy clustering iterative model," *Journal of Dalian University of Technology*, vol. 49, no. 6, pp. 932–936, 2009.

[19] S. Y. Chen and J. M. Hu, "Variable fuzzy assessment method and its application in assessing water resources carrying capacity," *Journal of Hydraulic Engineering*, vol. 37, no. 3, pp. 264–277, 2006.

[20] S. Y. Chen, *The System Fuzzy Decision Theory and Application*, Dalian University of Technology Press, Liaoning China, 1994.

[21] R. R. Zhou, *Research on Total Amount Control for Jiaozhou Bay Near Shore Area Pollution Based on ANN and Genetic Algorithms*, Ocean University of China, QingDao, China, 2009.

[22] J. Jin, H. Huang, and Y. Wei, "Comprehensive evaluation model for water quality based on combined weights," *Journal of Hydroelectric Engineering*, vol. 23, no. 3, pp. 13–19, 2004.

[23] L. Zheng, W. L. Cui, Y. Jia et al., "Evaluation on seawater quality by fuzzy comprehensive evaluation method in Qingdao dumping area," *Marine Environmental Science*, vol. 26, no. 1, p. 4, 2007 (Chinese).

[24] J. Yin, "On the management of marine dumping ground-taking the third category dredged material marine dumping ground outside the Jiaozhou bay, Qingdao as an example," *Coastal Engineering*, vol. 20, no. 1, p. 4, 2001 (Chinese).

[25] X. G. Li, J. M. Song, N. Li et al., "Source and biogeochemical characteristics of nitrogen and phosphorusin Jiaozhou Bay sediments," *Oceanologia et Limnologia Sinaca*, vol. 36, no. 6, pp. 562–571, 2005 (Chinese).

[26] Y. Li, Z. M. Yu, X. H. Cao et al., "Distribution and enrichment of heavy metals in surface sediments of Jiaozhou Bay," *Oceanologia et Limnologia Sinaca*, 2005.

[27] L. Wan, N. Wang, Q. Li et al., "Distribution of dissolved metals in seawater of Jinzhou Bay, China," *Environmental Toxicology and Chemistry*, vol. 27, no. 1, pp. 43–48, 2008.

[28] Z. G. Dong, A. G. Lou, and L. W. Cui, "Assessment of eutrophication of Jiaozhou Bay," *Marine Sciences*, vol. 34, no. 12, pp. 36–39, 2010.

[29] L. Q. Ma, Y. Li, Y. J. Zhao, S. Peng, and Q. Zhou, "Temporal and spatial trends of total petroleum hydrocarbons in the seawater of Bohai Bay, China from 1996 to 2005," *Marine Pollution Bulletin*, vol. 60, no. 2, pp. 238–243, 2010.

[30] D. Wang, V. P. Singh, and Y. Zhu, "Hybrid fuzzy and optimal modeling for water quality evaluation," *Water Resources Research*, vol. 43, no. 5, Article ID W05415, 2007.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 603629, 7 pages
http://dx.doi.org/10.1155/2013/603629

Research Article

Piecewise Trend Approximation: A Ratio-Based Time Series Representation

Jingpei Dan,[1] Weiren Shi,[2] Fangyan Dong,[3] and Kaoru Hirota[3]

[1] *College of Computer Science, Chongqing University, Chongqing 400044, China*
[2] *School of Automation, Chongqing University, Chongqing 400044, China*
[3] *Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology, 4259 Nagatsuta, Midoriku, Yokohama 226-8502, Japan*

Correspondence should be addressed to Jingpei Dan; danjingpei@cqu.edu.cn

Received 13 March 2013; Accepted 27 April 2013

Academic Editor: Fuding Xie

A time series representation, piecewise trend approximation (PTA), is proposed to improve efficiency of time series data mining in high dimensional large databases. PTA represents time series in concise form while retaining main trends in original time series; the dimensionality of original data is therefore reduced, and the key features are maintained. Different from the representations that based on original data space, PTA transforms original data space into the feature space of ratio between any two consecutive data points in original time series, of which sign and magnitude indicate changing direction and degree of local trend, respectively. Based on the ratio-based feature space, segmentation is performed such that each two conjoint segments have different trends, and then the piecewise segments are approximated by the ratios between the first and last points within the segments. To validate the proposed PTA, it is compared with classical time series representations PAA and APCA on two classical datasets by applying the commonly used K-NN classification algorithm. For ControlChart dataset, PTA outperforms them by 3.55% and 2.33% higher classification accuracy and 8.94% and 7.07% higher for Mixed-BagShapes dataset, respectively. It is indicated that the proposed PTA is effective for high dimensional time series data mining.

1. Introduction

Time series representation is one of the key issues in time series data mining, since the suitable choice of representation greatly affects the ease and efficiency of time series data mining. To address the high dimensionality issue in real-world time series data, a great number of time series representations by applying dimensionality reduction have been proposed.

Dimensionality reduction methods help to compare time series efficiently by modeling time series into a more compact form, whereas significant information about main trends in a time series, which are essential to effective similarity search, may be lost. To support accurate and fast similarity detection in time series, a number of special requirements that should be satisfied by any representation model are summarized as follows [1].

(i) *Time Warping-Awareness*. Time series should be modeled into a form that can be naturally mapped to the time domain. This will make it feasible to benefit from using dynamic time warping (DTW) that can compare time series with local time shifting and different lengths for similarity detection.

(ii) *Low Complexity*. Due to the high dimensionality of time series data, modeling time series should be performed maintaining a reasonably low complexity, which is possibly linear with the series length.

(iii) *Sensitivity to Relevant Features*. It is clearly desirable that time series approximation is able to preserve as much information in the original series as possible. For this purpose, approximating a time series should be accomplished in such a way that it tailors itself to the local features of the series, in order to capture the important trends of the series.

(iv) *Absence of Parameters*. Most representation models and dimensionality reduction methods require the

user to specify some input parameters, for example, the number of coefficients or symbols. However, prior domain knowledge is often unavailable, and the sensitivity to input parameters can seriously affect the accuracy of the representation model or dimensionality reduction method.

From an empirical viewpoint, it has been recently observed that there is no absolute winner among the time series representations in every application domain. Therefore, it is critical for time series representation to keep features that are important for corresponding application domains. The sensitivity to features can be considered according to three main subrequirements for the segments detected in an individual time series: (a) segments may have different lengths, (b) any segment represents different slopes (trends) of a subsequence of data points, and (c) segments capture the series trends [1].

Slopes [2] and derivative estimation [1] are adopted to denote trend of time series commonly in the literature. Due to the property of tangent function that is used to calculate slopes, it is difficult to distinguish two trends when the degrees of angles are close to $\pm 90^{\circ}$ by using slope to represent trend. In derivative time series segment approximation (DSA) representation [1], original time series is firstly transformed into the first derivative estimations of the points, and segmentation and approximation are based on the derivative estimations of time series. It has been observed that relative variations, the ratios between any two consecutive data points in a given time series, are suitable for representing trend in time series [3]. The magnitude of ratio reflects the variation degree of trend and the sign of ratio represents the changing direction of trend naturally. Based on ratio-based time series, a time series representation, piecewise trend approximation (PTA), is proposed, which retains the important feature of main trends of original time series concisely by dimensionality reduction. In contrast to the conventional representations based on raw time series data, the proposed PTA representation is based on local trends of raw time series data. That is to say, the raw data is firstly transformed into local trends (ratios), segmentation that separates time series into segments of different trends is then performed based on the ratios, and each segment is finally approximated by the ratios between the first and the last data points in the segment.

PTA is able to satisfy the first three requirements mentioned earlier.

(i) PTA representations can be compared by using DTW directly.

(ii) The ratio-based feature generation allows for representing a time series by focusing on the characteristic trends in the series.

(iii) Computational complexity for PTA is linear with the length of series, and the dimensionality of PTA is adaptive with the identified trends of the series.

To validate the proposed PTA, the performance of PTA for time series classification is compared to conventional representations. The experiments are based on two classical datasets by applying K-nearest neighbor (K-NN) classification method. The comparative experimental results show that PTA outperforms conventional representations in classification accuracy.

In Section 2, the time series representations with respect to different dimensionality reduce, techniques are reviewed. PTA representation is proposed in Section 3, and the experiments to validate the proposed PTA for time series classification are illustrated in Section 4.

2. Time Series Representations

To reduce dimensionality of a time series, a piecewise discontinuous function or low-order continuous function is usually applied to approximate it into a compact form. This study focuses on the first dimensionality reduction method, and the time series representations based on piecewise discontinuous functions are reviewed as follows.

The piecewise approximation-based representations include discrete wavelet transform (DWT) [4, 5], swinging door (SD) [6], Piecewise Linear Approximation (PLA) [7, 8], piecewise aggregate approximation (PAA) [9–11], adaptive piecewise constant approximation (APCA) [12], symbolic aggregate approximation (SAX) [13], and derivative time series segment approximation (DSA) [1].

Using DWT, a time series is represented in terms of a finite length, fast decaying, oscillating, and discretely sampled wave form (mother wavelet), which is scaled and translated in order to create an orthonormal wavelet basis. Each function in the wavelet basis is related to a real coefficient; the original series is reconstructed by computing the weighted sum of all the functions in the basis, using the corresponding coefficient as weight. The Haar basis [14] is the most widely used in wavelet transformation. The DWT representation of a time series of length n consists in identifying n wavelet coefficients, whereas a dimensionality reduction is achieved by maintaining only the first p coefficients (with $p > n$).

SD is a data compression technique that belongs to the family of piecewise linear trending functions. SD has been compared to wavelet compression. The SD algorithm employs a heuristic to decide whether a value is to be stored within the segment being grown or it is to be the beginning of a new segment. Given a pivot point, which indicates the beginning of a segment, two lines (the "doors") are drawn from it to envelop all the points up to the next one to be considered. The envelop has the form of a triangle according to a parameter that specifies the initial amplitude of the lines. The setup of this parameter has impact on the data compression level.

In the PLA method, a time series is represented by a piecewise linear function, that is, a set of line segments. Several methods have been proposed to recognize PLA segments (e.g., [7, 8]).

PAA transforms a time series of n points in a new one composed by p segments (with $p > n$), each of which is of size equal to n/p and is represented by the mean value of the data points falling within the segment.

Like PAA, APCA approximates a time series by a sequence of segments, each one represented by the mean value of its data points. A major difference from PAA is that APCA

can identify segments of variable length. Also, the APCA algorithm is able to produce high quality approximations of a time series by resorting to solutions adopted in the wavelet domain.

In SAX method, dimensionality of original time series is first reduced by applying PAA, then the PAA coefficients are quantized, and finally each quantization level is represented by a symbol so that SAX is a symbolic representation of time series.

The DSA representation is based on the derivative version of the original time series. DSA entails derivative estimation, segmentation, and segment modeling to map a time series into a different value domain which allows for maintaining information on the significant features of the original series in a dense and concise way.

For representing a time series of n points, it can be performed in $O(n)$ by using DWT, SD, (the fastest version of) PLA, PAA, SAX, and DSA, whereas the complexity of APCA is $O(n\log(n))$.

There are some other kinds of time series representations applying continuous polynomial functions to approximate time series, include Singular Value Decomposition (SVD) [15, 16], Discrete Fourier Transforms (DFT) [17, 18], splines, nonlinear regression, and Chebyshev polynomials [19, 20], of which the details are kindly referred to the references.

In contrast to conventional representations based on raw data, a time series representation based on ratios between any two consecutive data points in a given time series is proposed by applying piecewise segment approximation to reduce dimensionality in Section 3.

3. PTA: Piecewise Trend Approximation

Given a time series $Y = \{(y_1, t_1), \ldots, (y_n, t_n)\}$, where y_i is a real numeric value and t_i is the timestamp, it can be represented as a PTA representation

$$T' = \left\{\left(R_1, R_{t_1}\right), \ldots, \left(R_m, R_{t_m}\right)\right\}, \quad m \le n,\ n \in N, \tag{1}$$

where R_{t_i} is the right end point of the ith segment, R_i $(1 < i \le m)$ is the ratio between $R_{t_{i-1}}$ and R_{t_i} in the ith segment, and R_1 is the ratio between the first point t_i and R_{t_1}. The length of the ith segment can be calculated as $R_{t_i} - R_{t_{i-1}}$.

PTA approximates a time series by applying a piecewise discontinuous function to reduce dimensionality. The algorithm of PTA consists of three main steps:

(1) local trend transformation: the original time series is transformed into a new series where the values of data points are ratios between any two consecutive data points in original series;

(2) segmentation: the transformed local trend series is divided into variable-length segments such that two conjunctive segments represent different trends;

(3) segment approximation: each segment is represented by the ratios between the first and last data points within the segment, which indicates the characteristic of trend.

3.1. Local Trend Transform. Given a time series $Y = \{(y_1, t_1), \ldots, (y_n, t_n)\}$, $n \in N$, a new series $T = \{(r_1, t_2), \ldots, (r_n, t_n)\}$ is achieved from Y by local trend transform, where r_i is the value of ratio between (y_{i-1}, t_{i-1}), $i = 2, \ldots, n$.

Ratios between each two consecutive data points in Y are calculated according to the equation by justifying (1) as follows:

$$r_i = \frac{y_i - y_{i-1}}{y_{i-1}}, \quad i = 2, \ldots, n. \tag{2}$$

T is indeed a feature space of local trends mapped from the original data space with one dimension reduced. Although slope is often used to represents trend in the literature, it is difficult to distinguish two trends when the degrees of angles are close to $\pm 90^\circ$ due to the property of tangent function which is used to calculate slopes. Ratio, however, is more suitable for representing trend because the magnitude of ratio reflects the variation degree of trend and the sign of ratio represent the changing direction of trend naturally. Although T is one dimension reduced, it is not enough for many real-world applications. Hence, T will be compressed by the next two steps into a more concise form.

3.2. Segmentation. Given a time series $Y = \{(y_1, t_1), \ldots, (y_n, t_n)\}$, $n \in N$, Y is segmented into $S = \{S_1, \ldots, S_m\}$ $(m \le n, m \in N)$, where S_k $(k = 1, \ldots, m)$ is a subsequence of Y, which is decided by key points that certain behavior changes occur in Y. In PTA, segmentation is based on the local trend series T of original series Y. That is to say, the sequence $T = \{(r_1, t_2), \ldots, (r_n, t_n)\}$ is divided into the sequence $S = \{S_1, \ldots, S_m\}$ $(m \le n, m \in N)$, which is composed of m variable-length segments $S_k = \{(r_{k,1}, t_{k,1}), \ldots, (r_{k,j}, t_{k,j})\}$ $(k = 1, \ldots, m)\}$. Each two consecutive segments represent different trends. Since the segmentation in PTA is based on the ratios by local trend transform, of which signs represent trend directions, the main idea for segmentation is to separate T by finding out the first point such that the sign of it is different from those of the previous points. Assume that ε denotes the threshold of the ratios and $\text{sign}(r_i)$ denotes the sign of r_i in T, the sequence S_k is identified as a segment if and only if $\text{sign}(r_{k,1}) = \text{sign}(r_{k,2}) = \cdots = \text{sign}(r_{k,j})$, $|r_{k,j}| > \varepsilon$, and $\text{sign}(r_{k,j}) = \text{sign}(r_{k+1,1})$, $k = 1, \ldots, m$.

Accordingly, the raw data Y is segmented as $S' = \{S'_1, \ldots, S'_m\}$, $S'_k = \{(y_{k,l}, t_{k,l}), \ldots, (y_{k,j}, t_{k,j})\}$, $(k = 1, \ldots, m,\ m \le n,\ m \in N)$.

This segmentation aggregates the data points having the same changing directions so that the subsequences represent fluctuations in raw data intuitively. Thus, the reduced dimensionality is adaptive to the trend fluctuations and no parameter is needed.

3.3. Segment Approximation. To approximate the segments $S' = \{S'_1, \ldots, S'_m\}$, $m \le n$, $m \in N$, the ratio between the first and last point within each segment is calculated to represent the main trend information of any segment. Finally, the PTA representation $T' = \{(R_1, R_{t_1}), \ldots, (R_m, R_{t_m})\}$, $m \le n$, $n \in N$, is yielded such that

$$R_k = \frac{y_{k,j} - y_{k,l}}{y_{k,l}}, \quad k = 1, \ldots, m,$$
$$R_{t_k} = t_{k,j}. \tag{3}$$

The PTA representation maintains the important feature of trend variations in a concise form, while the computation complexity of it is linear with the length n of the sequence, that is, $O(n)$. In addition, since the length of PTA representation is determined by the fluctuations in original time series, similarities between PTA representations can be compared by applying dynamic time warping.

3.4. Distance Measure. To compare two time series data in similarity search tasks, various distance measures have been introduced. By far the most common distance measure for time series is the Euclidean distance [21, 22]. Given two time series X and Y of the same length n, the Euclidean distance between them is defined as

$$D(X, Y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}. \tag{4}$$

In PTA representation, original time series is segmented according to the change of local trend, and the length of the transformed PTA representation is thus adaptive with the trend variations in original time series. The Euclidean distance is limited to compare time series of equivalent length, and thus it cannot be applied to time series similarity search on PTA directly.

To address the limitation of Euclidean distance, dynamic time warping (DTW) has been proposed to evaluate the similarity of variable-length time series [23]. Unlike Euclidean distance, DTW allows elastic shifting of a sequence to provide a better match with another sequence; hence, it can handle time series with local shifting and different lengths. Therefore, DTW can be directly applied to measure similarity of time series in PTA form.

4. Experiments on Time Series Classification

Classification of time series has attracted much interest from the data mining community [24–26]. To validate the performance of the proposed PTA representation for similarity search in time series data, we design a classification experiment based on two classical datasets ControlChart and Mixed-BagShapes [27] by applying the most common classification algorithm, K-nearest neighbor (K-NN) classification. ControlChart is a synthetic dataset of six classes: normal, cyclic, increasing trend, decreasing trend, upward shift, and downward shift. Each class contains 100 instances. Figure 1 shows that representative sample instances in each class of ControlChart dataset. Mixed-BagShapes contains time series derived from 160 shapes with nine classes of objects, including bone, cup, device, fork, glass, hand, pencil, rabbit, and tool. The sample instances from each class of Mixed-BagShape are shown in Figure 2.

The proposed PTA is compared to two classical representations, PAA and APCA, which are introduced in Section 2. The K-NN classification algorithm is briefly reviewed in Section 4.1, data preprocessing is introduced in Section 4.2, and the experimental results are illustrated in Section 4.3.

4.1. K-Nearest Neighbor (K-NN) Classification. K-NN is one of the most widely used instance-based learning methods [28]. Given a set of n training examples, upon receiving a new instance to predict, the K-NN classifier will identify K-nearest neighboring training examples of the new instance and then assign the class label holding by the most number of neighbors to the new instance [29]. To classify time series data, it is straightforward to investigate the ability of time series representations for similarity search by applying K-NN algorithm since time series can be compared to the others as instances in K-NN.

4.2. Data Preprocessing. In order to reduce the noise in the data, original time series are usually preprocessed by smoothing techniques in time series data mining. It is essential to make data amenable to further data mining tasks by denoising. In PTA, it is necessary to denoise time series data before local trend transformation to avoid that the main trends are undistinguished from noise. Thus, smoothing is applied to denoise raw data before local trend transformation in PTA.

Commonly used smoothing techniques are moving average models including simple moving average, weighted moving average, and exponential moving average. In our experiments, exponential smoothing is applied to preprocess original data to reduce noise. Given a time series $X = \{x_t\}$ ($t = 0, \ldots, n$), the output $S = \{s_t\}$ ($t = 0, \ldots, n$) of the exponential smoothing algorithm is defined as

$$s_1 = x_0,$$
$$s_t = \alpha x_{t-1} + (1 - \alpha) s_{t-1}, \quad t > 1, \tag{5}$$

where α is the smoothing factor and $0 < \alpha < 1$.

4.3. Experimental Results of Time Series Classification. The most commonly used K-NN algorithm is utilized to facilitate independent confirmation of the proposed PTA representation. Concerning with the neighborhood size K in K-NN algorithm, the simple yet very competitive 1-NN algorithm is adopted in this experiment, that is, K-NN with K equal to 1. The parameter of sliding window in PAA representation and the threshold in PTA need to be predefined. The number of segments for PAA is decided by the sliding window while those of PTA and APCA are adaptive with fluctuations in original data. To compare the representations effectively, the parameters are tried several times such that the compressions (i.e., number of segments) of the representations are equal or at least very close. Classification accuracy is defined as

$$\text{accuracy} = 1 - E, \tag{6}$$

where E is the error rate.

The comparative results on ControlChart and Mixed-BagShapes by using leaving-one-out cross-validation are shown in Table 1. The results are the best results of each representation by trials of different parameters. For ControlChart,

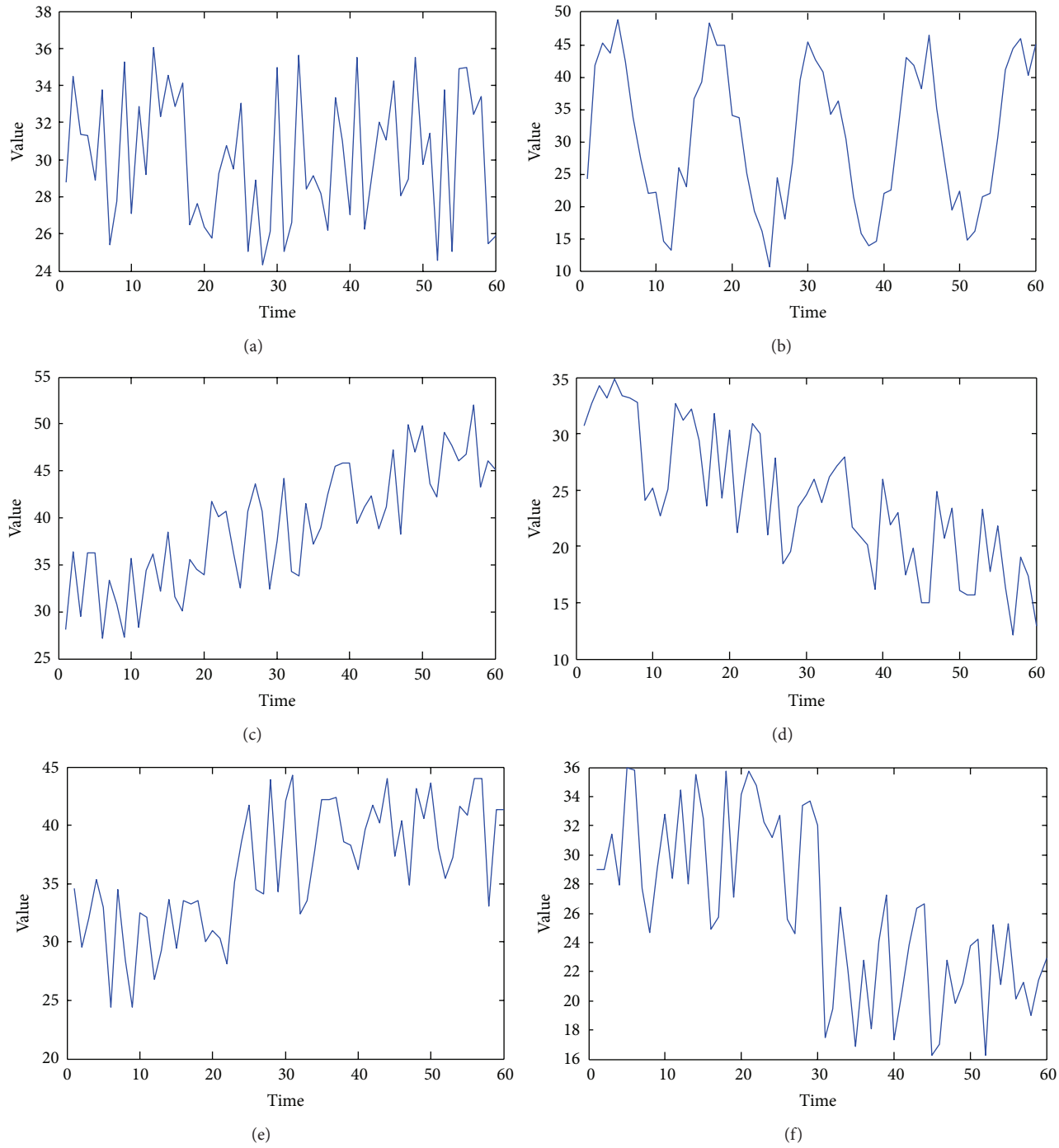


Figure 1: Sample instances from each class in ControlChart dataset: (a) normal; (b) cyclic; (c) increasing; (d) decreasing; (e) upward shift; (f) downward shift.

the proposed PTA outperforms PAA and APCA by 3.55% and 2.33% higher classification accuracy, respectively. For Mixed-BagShapes, PTA yields 8.94% and 7.07% improvement in classification accuracy compared with PAA and APCA, respectively. It is shown that the PTA outperforms the competitive representations by higher classification accuracy, which indicates that PTA is effective for time series classification by representing original data concisely with retaining important feature of trend variation.

5. Conclusions

In order to improve efficiency of time series data mining in high dimensional large-size databases, a time series representation piecewise trend approximation (PTA) is proposed to represent original time series into a concise form while retaining important feature of trend variations. Different from the representations based on original data space, PTA transforms original data space into the feature space of ratio

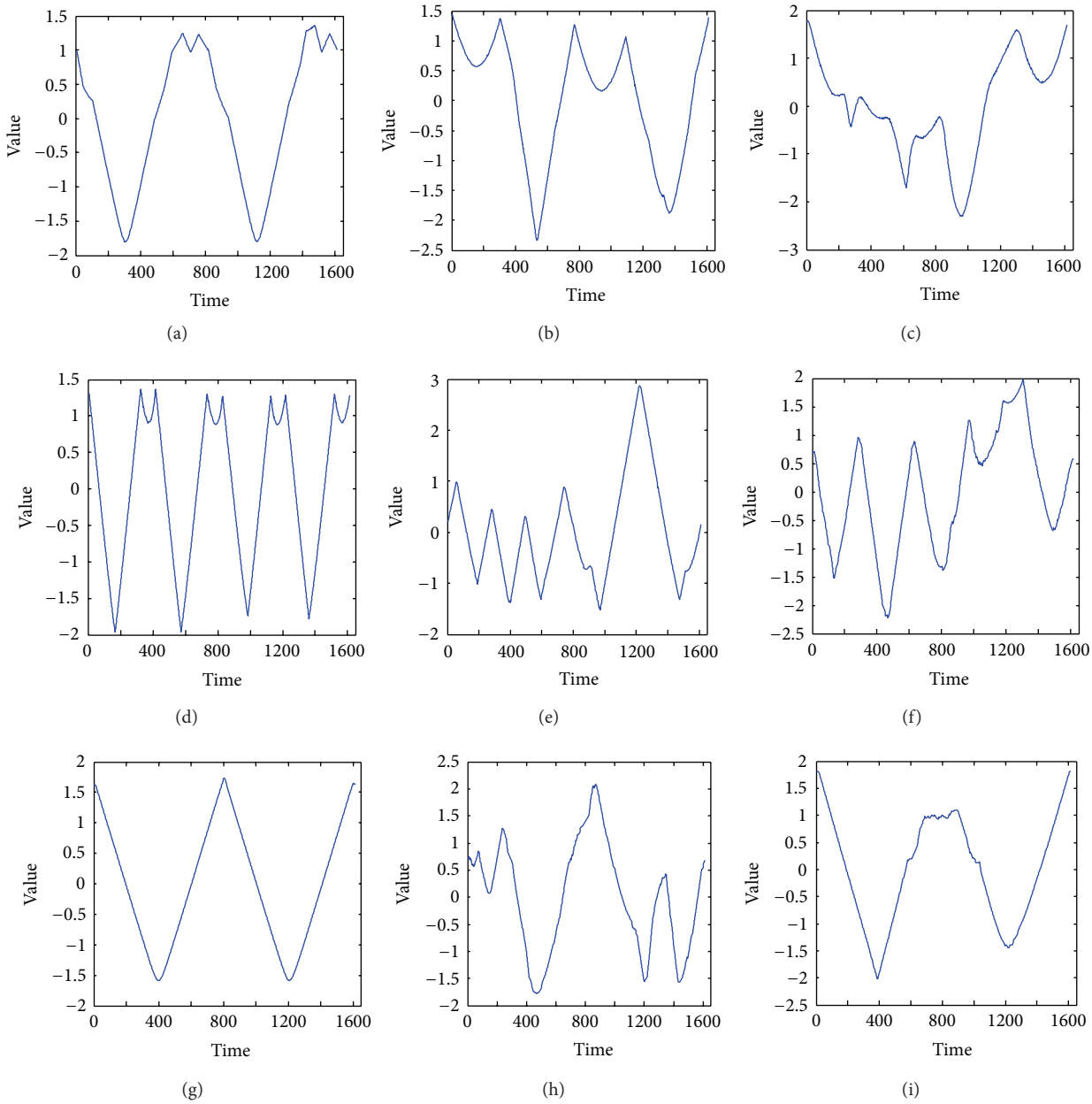


Figure 2: Sample instances from each class in Mixed-BagShapes dataset: (a) bone; (b) cup; (c) device; (d) fork; (e) glass; (f) hand; (g) pencil; (h) rabbit; (i) tool.

Table 1: Comparative results of classification accuracy for ControlChart and Mixed-BagShapes.

	PAA	APCA	Proposed PTA
ControlChart	0.952	0.964	0.987
Mixed-BagShapes	0.876	0.894	0.962

between any two consecutive data points in original time series, of which sign and magnitude indicate changing direction and degree of local trend, respectively. Based on the ratio-based feature space, segmentation is performed such that each two conjoint segments have different trends, and then the piecewise segments are approximated by the ratios between the first and last pints within the segments; dimensionality is, hence, reduced while keeping important feature of main trends in original data.

Based on two classical datasets, ControlChart and Mixed-BagShapes, by applying the commonly used time series classification algorithm K-NN, PTA is compared with classical PAA and APCA representations using DTW distance measure. The results for ControlChart show that PTA yields 3.55% and 2.33% improvements in classification accuracy compared to PAA and APCA, respectively. For Mixed-BagShapes, PTA outperforms PAA and APCA by 8.94% and 7.07% improvement, respectively. The time complexity of PTA algorithm is

linear with the length of original time series. The efficiency of time series data mining is, hence, enhanced by applying PTA representation. The applications of PTA in time series clustering, indexing, and other similarity search tasks will be validated and a symbolic time representation derived from PTA can be further developed.

Acknowledgment

This work is under Project no. 0216005202035 supported by the Fundamental Research Funds for the Central Universities in China.

References

[1] F. Gullo, G. Ponti, A. Tagarelli, and S. Greco, "A time series representation model for accurate and fast similarity detection," *Pattern Recognition*, vol. 42, no. 11, pp. 2998–3014, 2009.

[2] Y. Ding, X. Yang, A. J. Kavs, and J. Li, "A novel piecewise linear segmentation for time series," in *Proceedings of the 2nd International Conference on Computer and Automation Engineering (ICCAE '10)*, pp. 52–55, February 2010.

[3] K. Huarng and T. H. K. Yu, "Ratio-based lengths of intervals to improve fuzzy time series forecasting," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 36, no. 2, pp. 328–340, 2006.

[4] K. Chan and A. Fu, "Efficient time series matching by wavelets," in *Proceedings of the International Conference on Data Engineering (ICDE '99)*, pp. 126–133, 1999.

[5] Y. L. Wu, D. Agrawal et al., "A comparison of DFT and DWT based similarity search in time-series databases," in *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM '00)*, pp. 488–495, 2000.

[6] E. H. Bristol, "Swinging door trending: adaptive trending recording," in *Proceedings of the ISA National Conference*, pp. 749–753, 1990.

[7] T. Pavlidis and S. L. Horowitz, "Segmentation of plane curves," *IEEE Transactions on Computers*, vol. 23, no. 8, pp. 860–870, 1974.

[8] E. Keogh and M. Pazzani, "An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback," in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (KDD '98)*, pp. 239–241, 1998.

[9] E. Keogh, K. Chakrabarti, M. Pazzani, and S. Mehrotra, "Dimensionality reduction for fast similarity search in large time series databases," *Knowledge and Information Systems*, vol. 3, pp. 263–286, 2001.

[10] E. Keogh and M. Pazzani, "Scaling up dynamic time warping for data mining applications," in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (KDD '00)*, pp. 285–289, 2000.

[11] X. Feng, C. Cheng, L. Changling, and S. Huihe, "An improved process data compression algorithm," in *Proceedings of the International Conference on Intelligent Control and Automation*, pp. 2190–2193, 2002.

[12] K. Chakrabarti, E. Keogh, S. Mehrotra, and M. Pazzani, "Locally adaptive dimensionality reduction for indexing large time series databases," *ACM Transactions on Database Systems*, vol. 27, no. 2, pp. 188–228, 2002.

[13] J. Lin, E. Keogh, S. Lonardi, and B. Chiu, "A symbolic representation of time series, with implications for streaming algorithms," in *Proceedings of the ACM SIGMOD International Conference on Management of Data (DMKD '03)*, pp. 2–11, 2003.

[14] C. S. Burrus, R. A. Gopinath, and H. Guo, *Introduction to Wavelets and Wavelet Transforms, a Primer*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1997.

[15] F. Korn, H. V. Jagadish, and C. Faloutsos, "Efficiently supporting Ad Hoc queries in large datasets of time sequences," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 289–300, 1997.

[16] K. V. Kanth, D. Agrawal, and A. Singh, "Dimensionality reduction for similarity searching in dynamic databases," in *Proceedingss of the ACM SIGMOD International Conference on Management of Data*, pp. 166–176, 1998.

[17] D. Rafiei and A. Mendelzon, "Similarity-based queries for time series data," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 13–25, 1997.

[18] D. Rafiei and A. Mendelzon, "Efficient retrieval of similar time sequences using DFT," in *Proceedings of the International Conference on Foundations of Data Organization and Algorithms (FODO '98)*, pp. 249–257, 1998.

[19] J. C. Mason and D. C. Handscomb, *Chebyshev Polynomials*, Chapman & Hall/CRC, Boca Raton, Fla, USA, 2003.

[20] Y. Cai and R. Ng, "Indexing spatio-temporal trajectories with Chebyshev polynomials," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 599–610, 2004.

[21] G. Reinert, S. Schbath, and M. S. Waterman, "Probabilistic and statistical properties of words: an overview," *Journal of Computational Biology*, vol. 7, no. 1-2, pp. 1–46, 2000.

[22] E. Keogh and S. Kasetty, "On the need for time series data mining benchmarks: a survey and empirical demonstration," *Data Mining and Knowledge Discovery*, vol. 7, no. 4, pp. 349–371, 2003.

[23] D. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *Proceedings of the Workshop on Knowledge Discovery in Databases, at the 12th International Conference on Artificial Intelligence*, pp. 359–370, 1994.

[24] J. McQueen, "Some methods for classification and analysis of multivariate observation," in *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 281–297, 1967.

[25] A. Nanopoulos, R. Alcock, and Y. Manolopoulos, "Feature-based classification of time-series data," in *Information Processing and Technology*, pp. 49–61, Nova Science Publishers, Commack, NY, USA, 2001.

[26] C. A. Ratanamahatana and E. Keogh, "Making time-series classification more accurate using learned constraints," in *Proceedings of the 4th SIAM International Conference on Data Mining (SDM '04)*, pp. 11–22, April 2004.

[27] E. Keogh and T. Folias, "The UCR time series data mining archive," 2002, http://www.cs.ucr.edu/~eamonn/time_series_data/.

[28] T. M. Mitchell, ,*Machine Learning*, Computer Sciences Series, McGraw-Hill, New York, NY, USA, 1997.

[29] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transaction on Information Theory*, vol. 13, pp. 21–27, 1967.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 589386, 6 pages
http://dx.doi.org/10.1155/2013/589386

Research Article

An Enhanced Wu-Huberman Algorithm with Pole Point Selection Strategy

Yan Sun[1] and Shuxue Ding[2]

[1] *School of Psychology, Liaoning Normal University, Dalian 116029, China*
[2] *School of Computer Science and Engineering, Aizu University, Aizuwakamatsu 965-8580, Japan*

Correspondence should be addressed to Yan Sun; sunyan@lnnu.edu.cn

Received 26 February 2013; Accepted 23 April 2013

Academic Editor: Fuding Xie

The Wu-Huberman clustering is a typical linear algorithm among many clustering algorithms, which illustrates data points relationship as an artificial "circuit" and then applies the Kirchhoff equations to get the voltage value on the complex circuit. However, the performance of the algorithm is crucially dependent on the selection of pole points. In this paper, we present a novel pole point selection strategy for the Wu-Huberman algorithm (named as PSWH algorithm), which aims at preserving the merit and increasing the robustness of the algorithm. The pole point selection strategy is proposed to filter the pole point by introducing sparse rate. Experiments results demonstrate that the PSWH algorithm is significantly improved in clustering accuracy and efficiency compared with the original Wu-Huberman algorithm.

1. Introduction

Traditional data mining approaches can be categorized into two categories [1]: one is supervised learning, which aims to predict the labels of any new data points from the observed data-label pairs. Typical supervised learning methods include the support vector machine and the decision trees; the other one is unsupervised learning. The goal is just to organize the observed data points with no labels. Typical unsupervised learning tasks include clustering [2] and dimensionality reduction [3]. In this paper, we will focus on the clustering problem, which aims to divide data into groups with similar objects. From a machine learning perspective, clustering is to learn the hidden patterns of the dataset in an unsupervised way. From a practical perspective, clustering plays a vital role in data mining applications such as information retrieval, text mining, web analysis, marketing, and computing biology [4–7].

In the last decades, many methods [8–12] have been proposed for clustering. Recently, the graph-based clustering has attracted many interests in the machine learning and data mining community [13]. The cluster assignments of the dataset can be achieved by optimizing some criteria defined on the graph. For example, the spectral clustering is one kind of the most representative graph-based clustering approaches, and it aims to optimize some cut values (e.g., [14, 15]) defined on an undirected graph. After some relaxations, these criteria can usually be optimized via eigen decompositions, and the solutions are guaranteed to be globally optimal. In this way, the spectral clustering efficiently avoids the problems of the traditional K-means method.

Wu and Huberman proposed a clustering method based on the notation of voltage drops across the network [16]. The algorithm uses a statistical method to avoid the "poles problem" instead of solving it. The idea randomly picks two poles, then applies the algorithm to divide the graph into two communities, and repeats in this way for many times. The algorithm uses a majority vote to determine the communities [16]. However, after making some experiments, we have found that the choice of the pole points affects the accuracy of some of the clustering so seriously that the majority voting result is degraded. The specific details will be presented in Section 4.1 (Figure 1).

In order to overcome the above disadvantages of the Wu-Huberman algorithm, in this paper, first we construct a graph in terms of data points. Then we propose a novel strategy

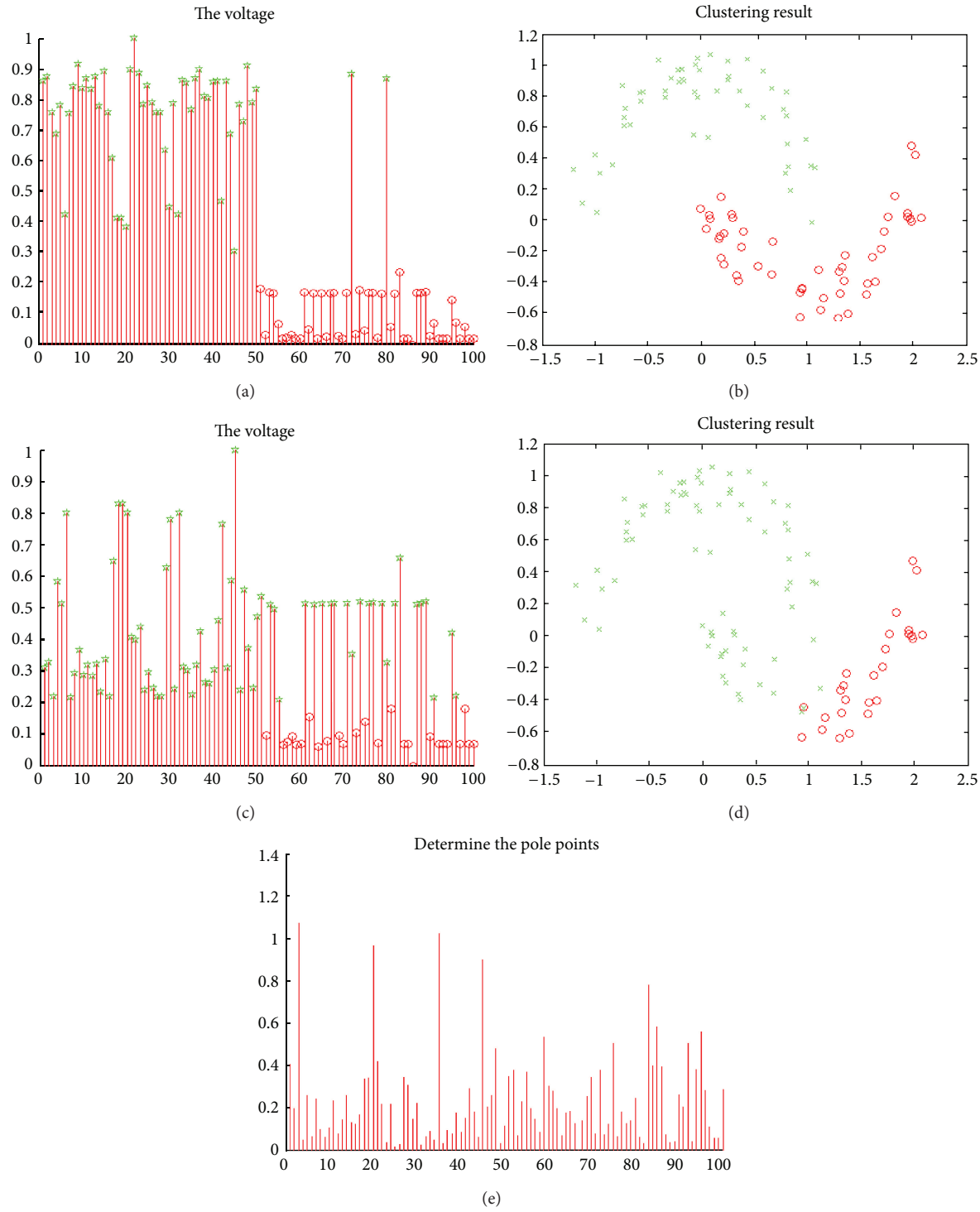


FIGURE 1: Clustering results of the Wu-Huberman algorithm for the two-moon pattern with different pole point selections. (a) The distribution of the voltage values when 22nd and 86th points have been chosen as the poles. (b) The clustering results corresponding with (a). (c) The distribution of voltage values when the 45th and 86th points have been chosen as the poles under the same dataset, algorithm, and parameters with (a). (d) The clustering results corresponding to (c). (e) The graph of determining the pole points. The x-axis is the data point number, and the y-axis is the value of sparse rate δ.

for pole point selection. After that, we iteratively solve the Kirchhoff equation to perform clustering. Finally, we get the clustering result. In this paper, we consider only the 2-community clustering case and will leave the case of k cluster problem into the future research.

2. Related Works

The Wu-Huberman algorithm exhibits the graph as an electric circuit. The purpose is to classify points in the graph into two communities, that is, clusters. We denote a graph by $G = (X, E)$, where X is the point set of graph and E is the edge set. The set of voltages of points is V. Suppose points A and B have been known to belong to different communities, G_1 and G_2, respectively. By solving Kirchhoff equations the voltage value of each point can be obtained, which of course should lie between 0 and 1. A point belongs to G_1 or G_2, which can be decided by voltage value of the point [17]. The graph is regarded as an electric circuit by associating a unit resistance to each of its edges. Two of the nodes, assumed to be node 1 and node 2, without losing the generality, in the graph are given a fixed potential difference. The Wu-Huberman method is based on an approximate iterative algorithm that solves the Kirchhoff equations for node voltages in linear time [16, 18].

The Kirchhoff equations of n-point circuit can be written as

$$V_1 = 1, \qquad V_2 = 0,$$
$$V_i = \frac{1}{d_i} \sum_{(i,j)\in E} V_j = \frac{1}{d_i} \sum_{j\in G} V_j a_{ij}, \quad \text{for } i = 3, \ldots, n, \tag{1}$$

where d_i is the degree of point x_i and a_{ij} is the adjacency matrix of the graph. After the convergence, each community, that is, cluster, is defined as the nodes with a specific voltage value within a tolerance. Without loss of generality, the algorithm has labeled the point in such a way that the battery is attached to point 1 and 2, which are termed as pole points.

Because of the complexity, the algorithm does not solve the Kirchhoff equations exactly rather solves it iteratively. The algorithm initially sets $V_1 = 1, V_2 = \cdots = V_n = 0$. In the first round, the algorithm starts updating from point 3 to the nth point in the following way. When the ith point, the voltage of it is substituted by the average value of its k neighbors according to (1). The updating process ends when the algorithm gets to the last point n, at which a *round* is finished. After repeating the updating process for a finite number of rounds, each point reaches voltage value that satisfies approximately the Kirchhoff equations within a certain precision. Then the algorithm finds community results by a threshold decision.

The Wu-Huberman algorithm inherits the superiority of the graph-based clustering. The final cluster solutions is global optimal. Especially, the running time of the algorithm is linear. However, the algorithm does not always work in many cases [16]. Besides, there is still one critical problem which seriously affects the accuracy and efficiency in real applications. That is, the accuracy and efficiency are greatly affected by the poles, that is, node 1 and node 2 selected. Therefore, it is most important to improve the method of selecting poles. In this paper, we present the PSWH algorithm to improve the accuracy and effectiveness of the algorithm by presenting the pole point selection strategy.

3. The PSWH Algorithm

3.1. Graph Construction. Let $G = (X, E)$ be an undirected graph with point set $X = \{x_1, \ldots, x_n\}$ and edge set $E \subseteq X \times X$. The degree of point $x_i \in X$ is defined as d_i, which is the edge number connecting with point x_i.

Constructing k nearest neighborhood graph is to model the local neighborhood relationships between the data points. Given data points $x_1, \ldots, x_n$, we link x_i and x_j with an undirected edge if x_i is among the k nearest neighbors of x_j or if x_j is among the k nearest neighbors of x_i. We define x_i and x_j to be adjacent if $x_i \in N(x_j)$ or $x_j \in N(x_i)$, $N(x_i)$, and $N(x_j)$ is the neighbor of x_i and x_j, respectively. w_{ij} is the similarity between x_i and x_j. w_{ij} is computed in the following way: $w_{ij} = e^{-(\|x_i - x_j\|^2/2\sigma^2)}$, where σ is a dataset-dependent parameter.

3.2. The Pole Point Selection Strategy. The Wu-Huberman algorithm selects pole point randomly. Based on plenty of experiments, we find that clustering results are very sensitive to the choosing of pole points. It may produce wrong clustering results if inappropriate points are chosen as the poles. Figure 1 gives us an intuitive illustration of such a problem.

For solving this problem, in this paper, we introduce a concept that is termed as "sparse points." There is the maximal diameter between the sparse point and its neighborhoods. The existence of sparse points will bias the final clustering results. An important fact of our experimental results is that if we choose sparse points as the pole points the Wu-Huberman algorithm will become less accurate. For this reason, the sparse points should not be selected as the pole points. Therefore, we propose the following sparse rate δ_i to discriminate the sparse points from the others. Additionally, in order to exclude the impact of the distribution in the similarity and degree, the averaging similarity of the neighbors and the similarity summation of the neighbors should be taken in the sparse rate δ_i. That is,

$$\delta_i = \frac{\gamma_i}{\left(\overline{\lambda}_i \times \lambda_i\right)}, \tag{2}$$

where γ_i is the maximum diameter between the ith point and its neighborhoods; $\gamma_i = \max \arg \sqrt{\sum_{j=1,p=1}^{d_i} \sum_{q=1}^{\text{number\_}f} (x_{ijq} - x_{ipq})^2}$, $i = 1$ to n, x_{ij} and x_{ip} are the neighborhoods of the x_i, j and p are from 1 to d_i, number_f is the feature number of x_i, and x_{ijq} is the qth attribute feature in the jth neighborhood of x_i. Here λ_i is the similarity (weight) summation of x_i's neighborhood, $\lambda_i = \sum_{j=1}^{d_i} w_{ij}$, $i = 1$ to n. $\overline{\lambda}_i$ is the average weight of x_i's neighborhood, $\overline{\lambda}_i = \lambda_i / d_i$.

Figure 1(e) shows the sparse rate of each point in Figure 1. A point can be determined as the pole point whose sparse

rate is significantly larger than those of the most other points. Sparse points are far from other points between two different clusters, so they should not be chosen as the pole points.

We define an extent to describe the range of allowed sparse points' number. For example, an extent of 5% in the two-moon example means that the allowed sparse point number is the number of points $*$ extent $= 100 * 5\% = 5$. That is to say, we choose top 5 points upon the sparse rate as the sparse points. The specific experimental details are shown in Section 4.1.

3.3. Iteratively Solving the Kirchhoff Equations. We will illustrate the computation procedure for iteratively solving the Kirchhoff equations by using an example. According to the results of (2), we get that the pole points are 1st and nth points. That is to say, $v_1 = 1, v_2 = \cdots = v_n = 0$. Then use (1) to obtain the voltage value of each point excluding the pole points, at which the voltage values are fixed. That is, the value of each point is the similarity average of its neighbor point. The updating process ends when we go through 2th to n-1th points. Repeat this process till voltage value converges within stable error range. In our experiments, we set 0.001 as the terminative conditioning of the iteration.

3.4. The Procedure of the PSWH Algorithm

Input. Dataset $X = \{x_i\}_{i=1}^n$ and the neighborhood size k.

Output. The cluster membership of each data point.

Procedure

Step 1: construct the k nearest neighborhood.

Step 2: compute sparse rate δ_i using (2) and apply the extent to determine the pole points. Then exclude the sparse points in graph and choose randomly two other points as the pole points.

Step 3: obtain the voltage value of each data point based on (1).

Step 4: output the cluster assignments of each data point.

4. Experimental Results

In this section, we will use the well-known two-moon example to illustrate the effectiveness of PSWH algorithm. The original dataset is a standard benchmark for machine learning algorithms [19] and is generated according to a pattern of two intertwining crescent moons. This benchmark is online available at http://www.ml.uni-saarland.de/GraphDemo/GraphDemo.html. In the experiments, the Gaussian noise with mean 0 and variance 0.01 has been added. The number of data points is set as 100 for the two moons.

4.1. Pole Points' Influence on the Clustering Accuracy. In the Wu-Huberman algorithm, the choice of the pole points affects significantly the clustering results. Taking the two-moon dataset as an example, we set σ as 0.5 and k as 5. In Figure 1(e), the sparse points are the 3rd, 20th, 35th, 45^{th}, and 83rd points. In order to improve the clustering accuracy, we do not choose the sparse points as the poles. The clustering accuracy is 100%. Figure 1(c) illustrates that no matter what threshold is chosen, the cluster accuracy is low. That is to say, the choice of the poles has great effect on the clustering results.

4.2. Pole Points' Influence on the Iterate Number. In the experiment, we find that the choice of the pole points has an impact on the iterate number. The two-moon dataset is taken as an example. All of the experiments are conducted in the same parameter conditions: such as $\sigma = 0.5$, the iterate error is 0.001, and the maximum iterate number is 100.

We first construct the KNN ($k = 5$) graph of original dataset. Then the degree of each point was computed and displayed in Figure 2(b). Next, we obtain the sparse rate of each point based on the degree distribution, which is the same as Figure 1(e). Finally, we choose the poles based on the sparse rate, compute (1) to obtain the voltage value of each point, and, respectively, display the iterate number of each point in Figures 2(c) and 2(d) when different poles are chosen.

In Figure 2, we can draw a conclusion that the greater degree of the poles corresponds to the more iterate number for convergence. Therefore, in order to decrease the iterate number of the algorithm, we should choose the points with smaller sparse degree as the poles. The clustering accuracy of Figure 2 is 100%.

4.3. Comparison with Other Algorithms. We compare the PSWH algorithm with other algorithms on the UCI repository, which is available at http://archive.ics.uci.edu/ml/.

From Table 1, we can find that the PSWH algorithm does slightly better than other algorithms in most dataset. However, in some conditions, the PSWH algorithm is lower than LCLGR algorithm. Considering the complexity of algorithm is linear, which is lower than LCLGR algorithm. Therefore, in general, the PSWH algorithm is an excellent algorithm than the others.

5. Conclusions and Future Work

In this paper, we propose PSWH algorithm for enhancing the clustering accuracy and efficiency of the Wu-Huberman algorithm, which can extend the applicability and increase the robustness of the algorithm. The concept of sparse points and selection procedure are presented to obtain the suitable pole points for the algorithm. The experimental results showed that the PSWH algorithm is very effective and stable when applied to clustering problems. In the future, we will give the theoretical analysis of the new algorithm and employ the new algorithm to more general and larger datasets. Furthermore, we will try to extend the new algorithm to textual, image, and video retrievals.

Table 1: Comparison with other algorithms on the clustering accuracy.

Data sets	K-means	Ncut [15]	LCLGR [1]	Wu-Huberman [16]	PSWH
BUPA	0.5623	0.5710	0.6493	0.5304	0.6145
Balance	0.5472	0.5195	0.5664	0.9983	0.9983
Monks	0.5806	0.7097	0.7339	0.6452	0.6690
Iris	0.5533	0.9867	0.9933	1	1
Crx	0.5038	0.6677	0.7871	0.5758	0.6263
Wine	0.5000	0.7416	0.8371	0.6667	0.7536
Hayes-Roth	0.4242	0.4015	0.4394	0.4015	0.4318

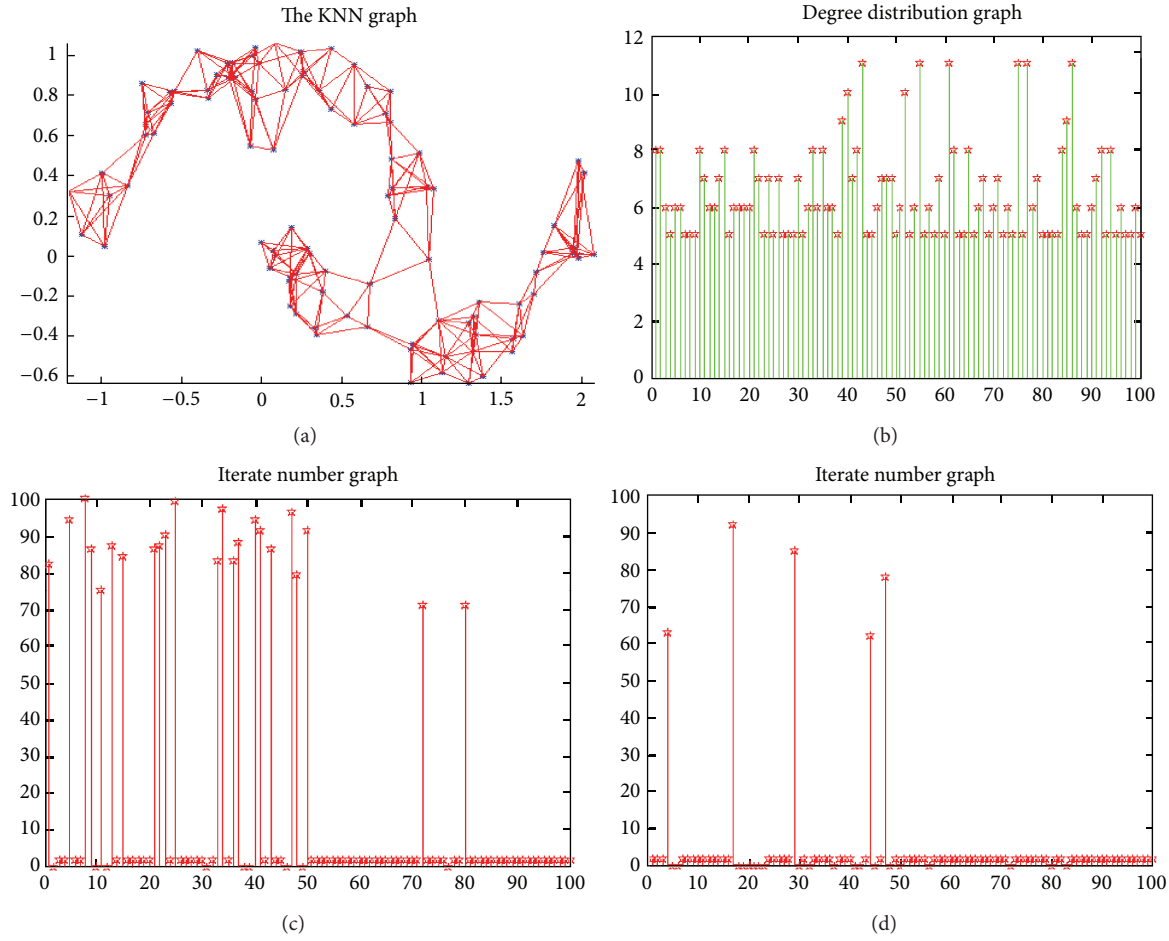


Figure 2: Different pole points of the Wu-Huberman algorithm were applied, which leads to different iterate number of convergence. (a) The KNN ($k = 5$) graph. (b) The degree distribution graph. (c) The iterate number via vertical axis when the poles are the 2nd point (its degree is 8) and the 77th point (its degree is 11). (d) The iterate number via vertical axis when the poles are the 5th point (its degree is 6) and the 56th point (its degree is 5), where the x-axis represents the data points and y-axis represents the iterate number.

Acknowledgments

This work was supported by the key project of the National Social Science Fund (11AZD089) and Educational Commission Scientific Project of Liaoning Province (no. L2012381).

References

[1] F. Wang and C. Zhang, "Label propagation through linear neighborhoods," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 1, pp. 55–67, 2008.

[2] Y. Sun, Y. Y. Tang, and L. Z. Yang, "An adaptive selection strategy and separation measure for improving the Wu-Huberman clustering," *ICIC Express Letters B*, vol. 3, no. 6, pp. 1531–1536, 2012.

[3] L. K. Saul, K. Q. Weinberger, F. Sha, J. Ham, and D. D. Lee, *Spectral Methods for Dimensionality Reduction*, MIT Press, 2006.

[4] W. H. Cui, W. Wang, X. B. Liu, and J. S. Wang, "An improved clustering algorithm for product family design of paper currency sorter," *IICIC Express Letters B*, vol. 3, no. 4, pp. 909–915, 2012.

[5] C. Cheng, D. Zhang, Z. Yu, and H. Li, "High speed data streams clustering algorithm based on improved SS tree," *ICIC Express Letters B*, vol. 3, no. 1, pp. 207–212, 2012.

[6] F. Wang, C. Zhang, and T. Li, "Clustering with local and global regularization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 12, pp. 1665–1678, 2009.

[7] X. Wang, X. Wang, and D. M. Wilkes, "A divide-and-conquer approach for minimum spanning tree-based clustering," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 7, pp. 945–958, 2009.

[8] D. X. Chang, X. D. Zhang, and C. W. Zheng, "A genetic algorithm with gene rearrangement for K-means clustering," *Pattern Recognition*, vol. 42, no. 7, pp. 1210–1222, 2009.

[9] L. Jing, M. K. Ng, and J. Z. Huang, "An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 8, pp. 1026–1041, 2007.

[10] M. K. Ng, M. J. Li, J. Z. Huang, and Z. He, "On the impact of dissimilarity measure in κ-modes clustering algorithm," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 503–507, 2007.

[11] M. H. Wang, Y. F. Tseng, H. C. Chen, and K. H. Chao, "A novel clustering algorithm based on the extension theory and genetic algorithm," *Expert Systems with Applications*, vol. 36, no. 4, pp. 8269–8276, 2009.

[12] K. R. Žalik, "An efficient k'-means clustering algorithm," *Pattern Recognition Letters*, vol. 29, no. 9, pp. 1385–1391, 2008.

[13] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: a geometric framework for learning from labeled and unlabeled examples," *Journal of Machine Learning Research*, vol. 7, no. 11, pp. 2399–2434, 2006.

[14] C. H. Q. Ding, X. He, H. Zha, M. Gu, and H. D. Simon, "A min-max cult algorithm for graph partitioning and data clustering," in *Proceedings of the 1st IEEE International Conference on Data Mining (ICDM '01)*, pp. 107–114, San Jose, Calif, USA, December 2001.

[15] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[16] F. Wu and B. A. Huberman, "Finding communities in linear time: a physics approach," *European Physical Journal B*, vol. 38, no. 2, pp. 331–338, 2004.

[17] S. E. Schaeffer, "Graph clustering," *Computer Science Review*, vol. 1, no. 1, pp. 27–64, 2007.

[18] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, "Complex networks: structure and dynamics," *Physics Reports*, vol. 424, no. 4-5, pp. 175–308, 2006.

[19] O. Chapelle, V. Sindhwani, and S. Keerthi, "Branch and bound for semi-supervised support vector machines," in *Advances in Neural Information Processing Systems*, B. Scholkopf, J. Platt, and T. Hoffman, Eds., vol. 19, MIT Press, Cambridge, Mass, USA, 2007.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 631717, 10 pages
http://dx.doi.org/10.1155/2013/631717

Research Article

The Sustainable Island Development Evaluation Model and Its Application Based on the Nonstructural Decision Fuzzy Set

Quanming Wang,[1,2] Peiying Li,[1,3] and Qinbang Sun[2]

[1] *College of Environmental Science and Engineering, Ocean University of China, Qingdo 266003, China*
[2] *National Marine Environment Monitoring Center, Dalian 116023, China*
[3] *The First Institute of Oceanography, SOA, Qingdao 266061, China*

Correspondence should be addressed to Quanming Wang; qmwang@nmemc.gov.cn

Received 27 February 2013; Accepted 7 April 2013

Academic Editor: Fuding Xie

Due to the complexity and diversity of the issue of sustainable island development, no widely accepted and applicable evaluation system model regarding the issue currently exists. In this paper, we discuss and establish the sustainable development indicator system and the model approach from the perspective of resources, the island environment, the island development status, the island social development, and the island intelligence development. We reference the sustainable development theory and the sustainable development indicator system method concerning land region, combine the character of the sustainable island development, analyze and evaluate the extent of the sustainable island development, orient development, and identify the key and limited factors of sustainable island development capability. This research adopts the entropy method and the nonstructural decision fuzzy set theory model to determine the weight of the evaluating indicators. Changhai County was selected as the subject of the research, which consisted of a quantitative study of its sustainable development status from 2001 to 2008 to identify the key factors influencing its sustainability development, existing problems, and limited factors and to provide basic technical support for ocean development planning and economic development planning.

1. Introduction

In recent years, the overdevelopment of and random construction on numerous islands has directly influenced the natural environment and the ecological balance of the islands, which has resulted in environmental deterioration and resource reduction and has even had catastrophic effects on the ecological environment of the islands [1]. Therefore, the quantitative description and evaluation of the extent of sustainable island development and the development orientation and restrictive factors are important for formulating island development strategy and developing future island economies [2].

Extensive research on sustainable development has been conducted at home and abroad, and many evaluation index systems and evaluation models related to sustainable development have been established, such as the ecological footprint model (EF) [3, 4], green GDP accounting system [5], resource carrying capacity model [6], and human development index (HDI) model [7]. But so far, because of the complexity and diversity of sustainable island development [2, 8–14], no generally accepted and applicable evaluation index system for sustainable island development has been established. This study has referred to the references about basic sustainable development theory [15–18] and the concept of a sustainable development index system [1, 2, 19–21], selected indexes from the point of view of the survival resource support, ecological environmental support, economic development support, and social and intellectual support [22], and established the evaluation index system and the evaluation model for sustainable island development. In addition, for multipurpose evaluation and decision, the rationality of the setting weights is emphasis and difficulty of evaluation and decision. The key of multipurpose evaluation and decision all the time is to improve the rationality of weight setting to avoid arbitraries. While analyzing the AHP

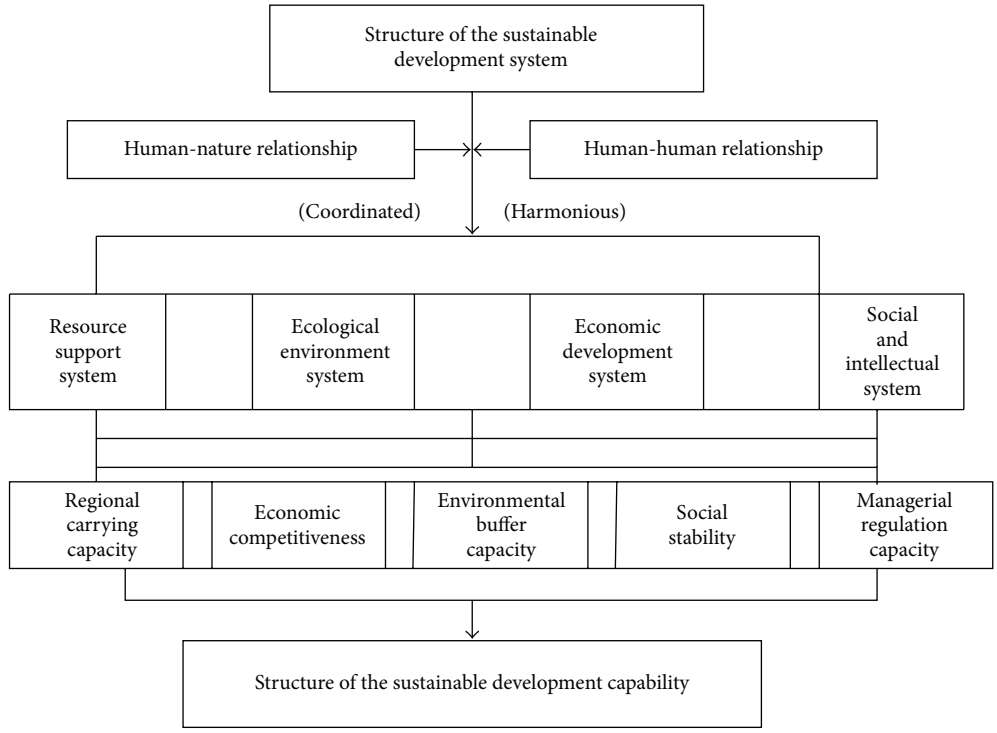


Figure 1: The framework of the sustainable development evaluation system.

model, Professor Chen Shouyu [23] made a binary comparison of the properties of elements by comparing their importance. Subsequently, with certain defects, the nonstructural decision-making fuzzy theory model has been offered. The nonstructural decision fuzzy set has been widely used in the evaluation of water resources carrying capacity [24] and the coordinated development of marine resources [25] and provides a scientific and reasonable decision method for the rationality of the subjective weight setting. The synthetic evaluation criterion weight was determined by combining the entropy weight and the nonstructural decision fuzzy set theory model in this paper. This paper analyzed and evaluated the extent of island sustainable development, development orientation, and restrictive factors to provide relevant theoretical support for the reasonable utilization and effective protection of island resources.

2. Materials and Methods

2.1. Sustainable Development Evaluation Model for Islands

2.1.1. Evaluation Index System Framework. The framework is the brief summary for the evaluation object, and the framework model can be used to define the boundary, composition, and structure of the evaluation object, to describe the correlations and interacting mechanisms of the subsystems, and to help the evaluator select and organize a series of problems [26]. The integration of population, resources, environment, development, and management decision-making is the core of sustainable development strategy and is also the key to coordinating human-nature relationships and interpersonal relationships (Figure 1). The coordination and sustainability of island development is the basic principle and core connotation for island sustainable development [8].

This research evaluated the sustainable island development potential from the view of the survival resource support, ecological environmental support, economic development support, and social and intellectual support. The survival resource support system is the basic condition for sustainable island development; the ecological environment system is the restrictive condition for sustainable island development; the economic development support system is the impetus condition for sustainable island development; the social and intellectual support system is the assuring and continuing condition for island sustainable development [27].

2.1.2. Evaluation Indexes of Sustainable Island Development. The index is the measurement of specific attributes of sustainable island development and the subsystems, which is further abstracted from the framework model and is a further decomposition of the overall goal and subgoals of sustainable development. This research has referred to such factors as island survival resources, survival sustainability, island development cost, level of environmental control and protection, level of ecological development, level of social development, educational, science, technology, and management ability. By combining the selection principle of the sustainable development evaluation index with the extent of collecting data about sustainable island development, this paper adopted the principal component analysis method, frequency statistical method, theoretical analysis method, and the Delphi method to establish three layers of the index system of sustainable

island development: the overall layer (A), the system layer (B), and the index layer (C; Figure 2). The evaluation system of sustainable island development selected 27 indexes to completely express the level, the capability, behavior, reason, and the impetus of change of sustainable island development and attempted to conduct a quantitative evaluation of the sustainable development conditions for the comprehensive island. Furthermore, this evaluation system is also significative and instructive in some extent to provide a reference for the development evaluations of the nonresident islands.

2.2. The Weight of the Evaluation Index. The determination of weight is the crucial problem for the variety of evaluation and decision-making, and the rationality of the weight setting directly affects the accuracy of the evaluation results. Consequently, avoiding artificial randomness to the greatest extent and obtaining a rational weight setting are very important and difficult for multiobjective evaluation and decision-making.

2.2.1. Determination of Objective Weight. The entropy method [28] is a relatively objective weighting method. This method completely relies on the relations of the assessment indictor data and forms a judgment matrix to calculate the entropy to determine the weight of each index. Thus, the determination of the index weight is objective, and the specific calculation steps are as follows.

Entropy of the assessment index i:

$$H_i = -\frac{1}{\ln n}\sum_{j=1}^{n} f_{ij} \ln f_{ij}, \tag{1}$$

among which:

$$f_{ij} = \frac{1 + b_{ij}}{\sum_{j=1}^{m}\left(1 + b_{ij}\right)}, \quad i = 1, 2, \ldots, m;\ \ j = 1, 2, \ldots, n;\ \ 0 \le H_i < 1. \tag{2}$$

Consequently, the entropy weight of the assessment index is

$$w_i^* = \frac{1 - H_i}{m - \sum_{i=1}^{n} H_i}. \tag{3}$$

If $H_i \to 1$ $(i = 1, 2, \ldots, m)$ of the index, the small changes of the entropy may lead to the changes in the weight of the indexes exponentially. Hence, it is very unreasonable. Referring to Zhou et al. [28], the following improvements of formula (3) were made:

$$w_i^{*\prime} = \frac{\sum_{i=1}^{n} H_i + 1 - 2H_i}{\sum_{i=1}^{n}\left(\sum_{i=1}^{n} H_i + 1 - 2H_i\right)}. \tag{4}$$

2.2.2. Experience Weight by the Nonstructural Decision-Making Fuzzy Model. While analyzing the AHP model, Professor Chen Shouyu [23] made a binary comparison of the properties of elements by comparing their importance. Subsequently, with certain defects, a nonstructural decision-making fuzzy theory model has been offered. The specific calculation steps are as follows.

Set the following target set which is about to compare the importance:

$$P = \{p_1, p_2, \ldots, p_m\}. \tag{5}$$

p_i is the target number i of this target set, $i = 1, 2, \ldots, m$; m is the total number of the target set. Conduct the binary comparison between p_k and p_l. If

(1) p_k is more important than p_l, the sort scale $e_{kl} = 1$, and $e_{lk} = 0$;

(2) p_k and p_l is of equal importance, $e_{kl} = 0.5$, and also $e_{lk} = 0.5$;

(3) p_l is more important than p_k, then $e_{kl} = 0$, and $e_{lk} = 1$, $k = 1, 2, \ldots, m$; $l = 1, 2, \ldots, m$.

And obtain the following binary comparison matrix of target set concerning the importance:

$$\mathbf{E} = \begin{pmatrix} e_{11} & e_{12} & \cdots & e_{1m} \\ e_{21} & e_{22} & \cdots & e_{2m} \\ \vdots & \vdots & & \vdots \\ e_{m1} & e_{m2} & \cdots & e_{mm} \end{pmatrix} = \left(e_{kl}\right), \tag{6}$$

which meets

(1) the value of e_{kl} can be set only from 0, 0.5, 1;

(2) $e_{kl} + e_{lk} = 1$;

(3) $e_{kk} = e_{ll} = 0.5$, $k = l$.

According to the descending sequence of the sum of each row in the binary comparison matrix, the order of importance of the target can be obtained under the circumstance of consistency.

Next, the targets are compared with each other in terms of the order of importance according to Table 1, and we get the binary comparison matrix

$$\boldsymbol{\beta} = \begin{pmatrix} \beta_{11} & \beta_{12} & \cdots & \beta_{1m} \\ \beta_{21} & \beta_{22} & \cdots & \beta_{2m} \\ \vdots & \vdots & & \vdots \\ \beta_{m1} & \beta_{m2} & \cdots & \beta_{mm} \end{pmatrix} = \left(\beta_{ij}\right), \tag{7}$$

which meets

$$0 \le \beta_{ij} \le 1, \quad i \ne j, \qquad \beta_{ij} + \beta_{ji} = 1, \qquad \beta_{ij} = 0.5, \quad i = j. \tag{8}$$

In this matrix, β_{ij} is the fuzzy scale of the importance of the target i for j, when the target i compares with j in terms of importance; β_{ji} is the fuzzy scale of the importance of the target j for i. Then, sum the fuzzy scale value β_{st} of each row

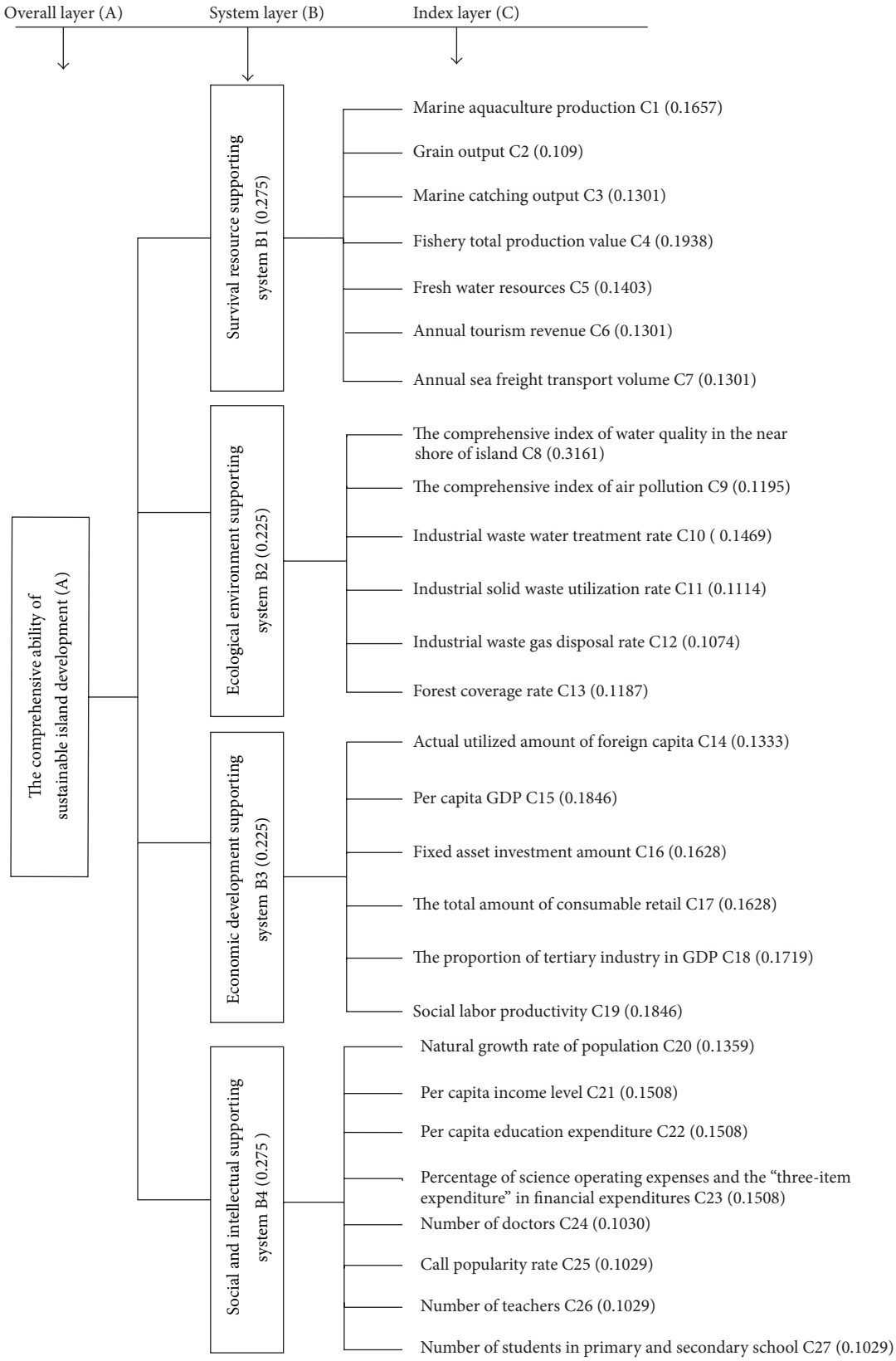


Figure 2: The evaluation index system of sustainable island development.

Table 1: The table of the relationship between fuzzy tone factor and fuzzy scale.

Fuzzy tone factor	Fuzzy scale
Equal	0.50
	0.525
Slightly	0.55
	0.575
Somewhat	0.60
	0.625
Rather	0.65
	0.675
Obviously	0.70
	0.725
Remarkably	0.75
	0.775
Very	0.80
	0.825
Extra	0.85
	0.875
Exceedingly	0.90
	0.925
Extremely	0.95
	0.975
Incomparable	1.0

in the matrix $\boldsymbol{\beta}$, and further normalize to get the following weight vector of the target set:

$$\mathbf{w} = (w_1 \quad w_2 \quad \cdots \quad w_m) = \left(\frac{1}{\sum_{i=1}^{m} (1-\beta_{1i})/\beta_{1i}} \quad \frac{(1-\beta_{12})/\beta_{12}}{\sum_{i=1}^{m} (1-\beta_{1i})/\beta_{1i}} \cdots \frac{(1-\beta_{1m})/\beta_{1m}}{\sum_{i=1}^{m} (1-\beta_{1i})/\beta_{1i}} \right). \tag{9}$$

2.2.3. Comprehensive Weight. The comprehensive weight of evaluation indexes is as follows:

$$w_i = \frac{\left(w_i^1 \cdot w_i^2\right)}{\sum_{i=1}^{m} w_i^1 w_i^2}. \tag{10}$$

W_{i1} is the objective weight of i index determined by entropy method. w_{i2} is the experience weight of i index determined by the nonstructural decision-making fuzzy model.

2.3. Calculation of the Synthetic Index. The synthetic index of sustainable island development is got by weighted summing up of the index coefficients of sustainable island development at the base of the normalization of index data, and the formula is as follows:

$$I = \sum_{i=1}^{n} (W_i * U_i), \qquad U_i = \sum_{j=1}^{n} \left(W_j * V_i\right). \tag{11}$$

W_i is the weight of supporting systems of sustainable island development, $\sum_{i=1}^{n} W_i = 1$; U_i is the assessment index value of the supporting systems; W_j is the weight of the assessment indexes of sustainable island development, $\sum_{j=1}^{n} W_j = 1$; V_i is the standardized value of the assessment indexes of sustainable island development. Considering the features and relevant studies [14, 18, 20, 29, 30] of sustainable island development, the synthetic index of sustainable island development can be divided into 5 levels: $I \geq 0.8$ represents an excellent condition of sustainable development; $0.7 \leq I < 0.8$ represents a good condition of sustainable development; $0.5 \leq I < 0.7$ represents a moderate condition of sustainable development; $0.4 \leq I < 0.5$ presents a weak condition of sustainable development; $I < 0.4$ represents a very weak condition of sustainable development.

3. Results and Discussion

3.1. Overview of the Study Area. The relatively ideal condition for demonstrating the assessment of sustainable island development is to take a certain island as a research sample. However, considering the difficulties in acquiring island data and the significant role of the country island in the implementation of the management of the marine environment, marine resources, maritime rights, and interests, this paper selected counties with more comprehensive information as the basic statistic unit of sustainable island development, and Changhai County of China is selected as the sample of sustainable island development assessment.

Located in the northern Yellow Sea on the east of Liaodong Peninsula of Liaoning Province of China, Changhai County is the only county in northeast China with territory consisting of islands, and it is also the only island county in the Chinese boundary (Figure 3). Covering a total island area of 119 square kilometers and with a coastal line of 359 kilometers and a maritime space of 7,720 square kilometers, Changhai County is primarily engaged in fisheries, and it is also the most developed island area in Liaoning Province. Changhai County governs two towns of Dachangshan Island and Zhangzi Island, three townships of Xiaochangshan, Guanglu, and Haiyang, 23 administrative villages, and 7 communities, with its county government located in the town of Dachangshan Island. By the end of 2008, the county had 26,232 registered households with a population of 74,010 among which 41,771 people lived in towns and townships.

Changhai County of China is a typical example of Chinese island development, displaying the striking features of island development in terms of strategic value, economic and resource values, ecological and environmental values, as well as social and cultural values. However, due to its incomplete

FIGURE 3: Geographical location map of Changhai County of China.

infrastructure, with the economic development of the island, the resources of Changhai County are consumed rapidly, and the contradictions among economic system, environmental system, and ecological system become increasingly sharp during the process of economic development, which has restricted the development of Changhai and made the economic development of Changhai County rank in a relatively backward position among the 12 island counties of China. Therefore, it is of particular importance to research the sustainable development system of Changhai.

3.2. Sources of Data. The index data are primarily from statistical yearbooks of Changhai County (2001–2008), the environmental quality reports of Changhai County (2001–2005), the annual environmental monitoring reports of Changhai County (2006-2007), the statistical annuls of the economy, and the social development of Changhai County (2001–2008). A majority of the index data can be obtained directly from the statistical annuals, and a minority of the data was acquired through calculation.

3.3. The Calculation of Index Weight and the Synthetic Index. The following paragraph explains the detailed process of determining the synthetic index by assigning the following 6 index weights: the comprehensive index of seawater quality index of island environmental supporting system layer (C_8), the comprehensive index of air pollution (C_9), industrial waste water treatment rate (C_{10}), industrial solid waste utilization rate (C_{11}), industrial waste gas disposal rate (C_{12}), and forest coverage rate (C_{13}).

We determined and standardized the index statistical data matrix of Changhai County from 2001 to 2007. The standardized matrix is as follows:

$$B = \begin{bmatrix} 1 & 0.4043 & \cdots & \cdots & 1 & 1 \\ 0 & 1 & \cdots & \cdots & 0.3765 & 0 \\ 0.9229 & 0.0284 & \cdots & \cdots & 0.8580 & 0.9589 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0.0898 & 0.6454 & \cdots & \cdots & 0.1488 & 0.7352 \end{bmatrix}. \quad (12)$$

Following the entropy weight solution procedure, we can obtain the 6 index weights

$$w_1' = (0.7078, 0.5182, 0.4901, 0.3717, 0.3583, 0.3339). \quad (13)$$

Normalizing the entropy weight of the index, we get the objective weight of the indexes

$$w_1 = (0.2546, 0.1864, 0.1763, 0.1337, 0.1289, 0.1201). \quad (14)$$

According to the theorem of the order of importance, the following matrix E, the scale of the order of importance of the six indexes, can be obtained:

$$E = \begin{bmatrix} 0.5 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0.5 & 1 & 1 & 1 \\ 0 & 0.5 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0.5 \\ 0 & 0 & 0 & 0.5 & 1 & 0 \end{bmatrix} \begin{matrix} ① \\ ③ \\ ② \\ ⑤ \\ ⑥ \\ ④ \end{matrix}. \quad (15)$$

According to the order of importance matrix E, based on the consideration of expert opinions and references to numerous studies in the literature, we conclude that C_8 is slightly more important than C_9, is obviously more important than C_{10}, C_{11}, and C_{12}, and is slightly more important than C_{13}; Compared with C_{10}, C_{11}, and C_{12}, C_9 is in between slightly more important and relatively more important and is equally important with C_{13}; C_{10} is equally important with C_{11} and C_{12}; compared with C_{10}, C_{13} is between slightly more important and relatively more important. According to the corresponding relation between fuzzy tone factors and fuzzy scale values in Table 1, a two-dimensional comparison matrix β with 6 targets can be determined as follows:

$$\beta = \begin{bmatrix} 0.5 & 0.6 & 0.7 & 0.7 & 0.7 & 0.6 \\ 0.4 & 0.5 & 0.625 & 0.625 & 0.625 & 0.5 \\ 0.3 & 0.375 & 0.5 & 0.5 & 0.5 & 0.375 \\ 0.3 & 0.375 & 0.5 & 0.5 & 0.5 & 0.375 \\ 0.3 & 0.375 & 0.5 & 0.5 & 0.5 & 0.375 \\ 0.4 & 0.5 & 0.625 & 0.625 & 0.625 & 0.5 \end{bmatrix}. \quad (16)$$

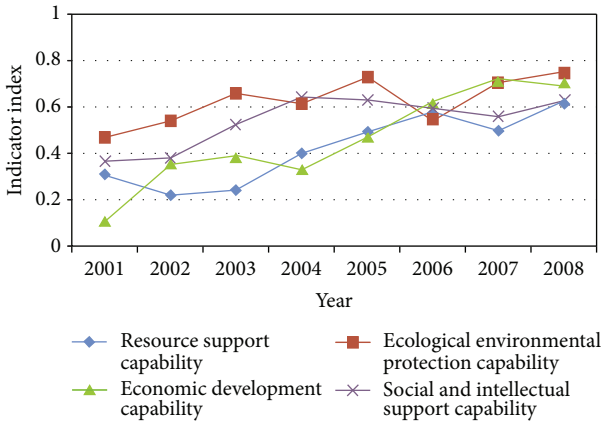


Figure 4: The comprehensive index of supporting systems.

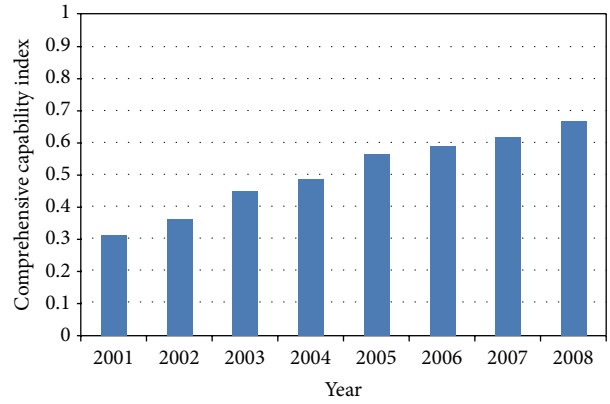


Figure 5: The comprehensive capability index of sustainable development (2001–2008).

The sum of the fuzzy scale values can be used to show the weight vector of the index, as follows:

$$w = \left(\sum_{t=1}^{m}\beta_{1t}, \sum_{t=1}^{m}\beta_{2t}, \ldots, \sum_{t=1}^{m}\beta_{mt}\right) = (3.8, 3.275, 2.55, 2.55, 2.55, 3.025). \quad (17)$$

Normalizing the fuzzy scale values matrix, we obtain the weights of 6 indexes of the island resources support system:

$$w_2 = (0.2141, 0.1845, 0.1437, 1437, 0.1437, 0.1704). \quad (18)$$

According to formula (10), we can combine the forementioned mathematical weights with the experience weight by the nonstructural decision-making fuzzy model and eventually obtain the comprehensive weights of 6 indexes:

$$w = (0.3161, 0.1995, 0.1469, 0.1114, 0.1074, 0.1187). \quad (19)$$

Because the calculation method of the weights of other layers is the same as this, this paper omitted the details and directly showed the weight results of sustainable island development support system, which is shown in Figure 2.

The synthetic evaluation indexes of the sustainable development systems of Changhai County are shown in Figure 4, Figure 5, and Table 2.

3.4. The Analysis of Results. During the period from 2001 to 2008, the comprehensive sustainable development capability of Changhai County improved continuously and steadily, with its value (A) increasing to 0.67 in 2008 from 0.31 in 2001, which indicates that the comprehensive strength of sustainable development had doubled. By 2008, the comprehensive sustainable development capability of Changhai County had reached the middle development level (0.67), and Changhai County had achieved significant development in economic, social, scientific, and educational levels.

In the evaluation system of sustainable development of Changhai County, the ecological environmental protection system is of the highest realization degree in sustainable development, whereas the survival resource system and the social and intellectual support system are of a relatively low realization degree. The low sustainable development capability of the survival resource system and the social intellectual support system is the primary reason why the comprehensive sustainable development level of Changhai County is in the middle, which limits the improvement of the comprehensive sustainable development capability.

The survival resource support ability of Changhai County is at a weaker level of sustainable development during the period from 2001 to 2008, with the trend of slow rising. Within the period, all indexes of the survival resource support system are at lower levels, except for per capita amount of aquatic products. Although per capita water supply capacity, which is among these indexes, becomes stronger and stronger, the maximum amount of per capita water supply capacity is 33.8 cubic meters per person, which owns only 25% of the national average and 73.4% lower than the average level of Liaoning Province. The power supply capacity of Changhai County is strengthening, and the capacity in 2011 (1497.1 kwh per person) grows three times as much as that of 2001, strengthening the power resource supply ability of sustainable development of Changhai County. However, the base number of per capita power supply capacity is low, only owning 32% of the average of Dalian (in Dalian, the amount of per capita power supply is 4685.8 kwh per person). Within the period, the area of per capita marine area lasts between 9.5 hm^2 and 10.5 hm^2, which are among the intermediate and advanced levels. This indicates that marine areas, coastlines, beaches, and other resources are well protected, and the utilization of marine area is quite efficient, which ensures the sustainable development of Changhai County marine islands. The yield of agricultural products in Changhai is quite low, with the per capita yield only 102.55 kg at most and 75% lower than the average level of China and Liaoning Province. The grain supply mainly depends on mainland, but the grain sources and marine transport are restricted by many circumstances. Thus, the gross supply capacity of Changhai County is very vulnerable. The fishery of Changhai County is one of the competitive industries all the time, which has been proved that during the period, the per capita aquatic product

Table 2: The synthetic evaluation indexes of sustainable development systems of Changhai county.

	2001	2002	2003	2004	2005	2006	2007	2008
Resource support capability	0.31	0.22	0.24	0.40	0.49	0.58	0.50	0.62
Ecological environmental protection capability	0.47	0.54	0.66	0.62	0.73	0.55	0.71	0.75
Economic development capability	0.11	0.35	0.39	0.33	0.47	0.62	0.72	0.70
Social and intellectual support capability	0.37	0.38	0.53	0.65	0.63	0.59	0.56	0.63
Comprehensive capability of sustainable development	0.31	0.36	0.45	0.49	0.57	0.59	0.62	0.67

yield is rising, and the yield in 2011 is 2.3 times as much as that in 2001, which ranks the second in the list of China's 12 island counties. The fishery development of Changhai County has been in the forefront of the nation, which promotes the sustainable development of Changhai County to some extent. The cargo throughput of the ports of Changhai County is at low level, and the maximum amount of marine transport in 2011 is 1.79 million tons, which ranks the tenth in the list of China's 12 island counties. That is to say, the port resources are not thoroughly utilized. In conclusion, although many of the indexes of survival resource support ability of Changhai County are rising, the maximum value of each index remains at the low level, showing that the resource support capacity of sustainable development is quite poor.

The ecological environmental support capacity of Changhai County is at a stronger level during the period from 2001 to 2008. During the period, the average for many years of seawater water quality comprehensive index of inshore region of Changhai County is 2.09, and that of drinking water in land is 1.41. In this case, the quality of both marine water and drinking water has reached higher criterion. The per capita public grassland area of Changhai County is rising from 7 square meters per person in 2001 to 10.43 square meters per person in 2011 by 49%. This indicates that Changhai County is paying more and more attention to ecological environment. Marine disaster has become the main natural disaster, which brings about that the loss disasters can directly affect the level of sustainable development. The proportion of the loss of marine disasters in GDP goes up, and such proportion in 2011 is 16.6%. This indicates that Changhai County is weak in handling marine disasters. In conclusion, although the ecological environmental support ability of Changhai County is generally at a higher level, it is quite poor for Changhai County in overcoming natural disasters. Therefore, it is highly suggested that the government should improve the marine disaster forecast and prevention.

Among 4 support systems of sustainable development of Changhai County, the economic development support system has the most rapid growth. The sustainable development index reached 0.70 in 2008, indicating the middle sustainable development level. The success of the economic development support system is closely related to the development strategy of constructing the "Liaoning at Sea" and "Dalian at Sea" demonstration area projects and to prosperous fisheries, industries, and tourism, which have been implemented by Changhai County in recent years. Social labor productivity, investment in fixed assets, and total sales of consumer goods have increased greatly. In 2008, Changhai County realized a gross regional product (GRP, large-caliber) of 3,463.46 million Yuan, and GRP per capita reached 46,729 Yuan (Statistical Bureau of Changhai County in China (2001–2008)), which indicates that Changhai County ranked among the top national counties for marine economy. However, similar to most island counties, a natural traffic obstacle exists between Changhai County and the outside due to its unique geographic location and resource features. The traffic construction between islands and between land and the islands is slow, and meteorological conditions greatly influence the flow of people and logistics, having become one of the important factors limiting the economic development of Changhai County. Furthermore, the economic structure of Changhai County is singular, with the primary industry focusing on fishery. In 2008, the proportion of the three industries (primary industry, secondary industry, and tertiary industry) in Changhai County was 74.2 : 13.9 : 11.9, and the contribution rate of the primary industry to economic growth reached 90%. So, Changhai County needs to change the existing fishery-focused economic development mode, accelerate the economic restructuring, and promote the overall optimization and upgrading of three industries.

Among 4 support systems of the sustainable development of Changhai County, the social and intellectual support system shows the slowest growth. As observed from the internal realization indexes of sustainable development, the proportion of educational expenditure per capita, expenditure on science and three items of expenditure on science and technology in financial expenditure, as well as the number of teachers, the number of enrolled students in primary and middle schools, and other intellectual support factors greatly influence the realization of sustainable development of the social and intellectual support system. During the period from 2001 to 2008, the income per capita of Changhai increased continuously and steadily, increasing to 17,256 Yuan in 2008 from 5,399 Yuan in 2001, an increase of 219%. However, the increase in the investment in educational funds per capita was not high. From 2001 to 2007, the investments in educational funds per capita were 336 Yuan, 340 Yuan, 390 Yuan, 430 Yuan, 617 Yuan, 517 Yuan, and 693 Yuan, respectively; the proportion of expenditure on science and three items of expenditure on science and technology in financial expenditure was approximately 1% (Statistical Bureau of Changhai County in China (2001–2008)). The insufficient

investment in education and scientific research and "having difficulty in attracting and retaining talents" are still problems restricting the development of Changhai County. Consequently, it is necessary for Changhai County to provide flexible employment mechanisms, establish a sound talent training system, increase investment in education and scientific research, and stabilize a talent team to enhance sustainable development.

4. Conclusion

Islands are an important part of Chinese territory and are also important bases for ocean development. The sustainable island development relates to the sustainable development of the ocean and even the national economic system [31]. This paper (1) selected indexes in terms of survival resource support, ecological environmental support, economic development support, and social and intellectual support, (2) established the evaluation index system and evaluation model of sustainable island development, (3) analyzed and evaluated the sustainable island development level, development orientation, and restrictive factors, (4) combined the entropy method with the nonstructural decision-making fuzzy set theory model to determine the weight of the evaluation indexes of sustainable island development, (5) took Changhai County of China as an evaluation example of sustainable island development, and (6) conducted quantitative research on the sustainable development conditions of the county. The results showed that the comprehensive sustainable development capability of Changhai County improved continuously. The sustainable development index increased to 0.67 in 2008 from 0.31 in 2001, reaching the middle development level. Among the support systems for the sustainable development of Changhai County, the ecological environmental system has been most fully realized in sustainable development, whereas the survival resource system and the social and intellectual support system are of a relatively low realization degree. The primary reason why the sustainable development of Changhai County is in the middle level is that the county has relatively few survival resources per capita, poor survival sustainability, a relatively low educational level, and low technological and management capability, which limit the improvement of the comprehensive sustainable development capability. The survival sustainability level and island environmental and protection level are important factors in determining the sustainable development capability, whereas fresh water resources, mariculture, the extent of tertiary industry development, educational investment, and the state of scientific research are restrictive factors influencing sustainable development. Therefore, Changhai County should properly enhance the infrastructure construction and increase the investment in education and scientific research to promote the improvement of the comprehensive sustainable development capability. Furthermore, in the process of island development and construction, Changhai County should enhance the protection of the resources and the environment, reasonably develop and utilize island resources, and truly promote the construction and development of the island.

Acknowledgment

Foundation item: under the auspices of 908 special fund of the State Oceanic Administration: the offshore marine environment quality evaluation of Liaoning province (LN-908-02-04).

References

[1] L. Sun, J. Ni, and A. G. Borthwick, "Rapid assessment of sustainability in Mainland China," *Journal of Environmental Management*, vol. 91, no. 4, pp. 1021–1031, 2010.

[2] J. Li and G. C. Wang, "Establishment and discussion of appraisement indicators system of islands stainable development," *Marine Environmental Science*, vol. 23, no. 1, pp. 54–57, 2004 (Chinese).

[3] I. Moffatt, "Ecological footprints and sustainable development," *Ecological Economics*, vol. 32, no. 3, pp. 359–362, 2000.

[4] B. Du, K. Zhang, G. Song, and Z. Wen, "Methodology for an urban ecological footprint to evaluate sustainable development in China," *International Journal of Sustainable Development and World Ecology*, vol. 13, no. 4, pp. 245–254, 2006.

[5] C. Yang and J. P. H. Poon, "A regional analysis of China's green GDP," *Eurasian Geography and Economics*, vol. 50, no. 5, pp. 547–563, 2009.

[6] Q. Di, Z. Han, G. Liu, and H. Chang, "Carrying capacity of marine region in Liaoning Province," *Chinese Geographical Science*, vol. 17, no. 3, pp. 229–235, 2007.

[7] D. J. Chen, P. Li, J. Du, L. Liu, and X. Xu, "The evaluation of sustainable development based on ecological footprint and human development index—a case of marine fishery resources utilization in china," *Science and Society*, no. 5, pp. 96–103, 2006 (Chinese).

[8] J. W. Rogers, "Sustainable development patterns: the Chesapeake Bay region," *Water Science and Technology*, vol. 26, no. 12, pp. 2711–2721, 1992.

[9] M. D. Griffith and J. Ashe, "Sustainable development of coastal and marine areas in small island developing states: a basis for integrated coastal management," *Ocean and Coastal Management*, vol. 21, no. 1–3, pp. 269–284, 1993.

[10] M. D. Griffith, "Reflections on the implementation of the programme of action on the sustainable development of small island developing states (SIDS)," *Ocean and Coastal Management*, vol. 29, no. 1–3, pp. 139–163, 1995.

[11] K. C. Tran, "Public perception of development issues: public awareness can contribute to sustainable development of a small island," *Ocean and Coastal Management*, vol. 49, no. 5-6, pp. 367–383, 2006.

[12] T. B. Ramos, S. Caeiro, C. H. Douglas, and C. Ochieng, "Environmental and sustainability impact assessment in small islands: the case of Azores and Madeira," *International Journal of Environmental Technology and Management*, vol. 10, no. 2, pp. 223–240, 2009.

[13] J. Forster, I. R. Lake, A. R. Watkinson, and J. A. Gill, "Marine biodiversity in the Caribbean UK overseas territories: perceived threats and constraints to environmental management," *Marine Policy*, vol. 35, no. 5, pp. 647–657, 2011.

[14] X. Ni, Y. Q. Wu, J. Wu, J. Lu, and P. C. Wilson, "Scenario analysis for sustainable development of Chongming Island: water resources sustainability," *Science of the Total Environment*, vol. 439, no. 15, pp. 129–135, 2012.

[15] S. Avdimiotis, "Necessity of the early warnings system for the development of new sustainable forms of tourism. The case of Chios island, North Aegean," *Journal of Environmental Protection and Ecology*, vol. 9, no. 2, pp. 431–435, 2008.

[16] M. Fortuny, R. Soler, C. Cánovas, and A. Sánchez, "Technical approach for a sustainable tourism development. Case study in the Balearic Islands," *Journal of Cleaner Production*, vol. 16, no. 7, pp. 860–869, 2008.

[17] R. Ciegis, J. Ramanauskiene, and B. Martinkus, "The concept of sustainable development and its use for sustainability scenarios," *Engineering Economics*, vol. 2, no. 62, pp. 28–37, 2009.

[18] B. Melnikas, "Sustainable development and creation of the knowledge economy: the new theoretical approach," *Technological and Economic Development of Economy*, vol. 16, no. 3, pp. 516–540, 2010.

[19] A. Barrera-Roldán and A. Saldívar-Valdés, "Proposal and application of a sustainable development index," *Ecological Indicators*, vol. 2, no. 3, pp. 251–256, 2002.

[20] M. Golusin and O. Munitlak Ivanović, "Definition, characteristics and state of the indicators of sustainable development in countries of Southeastern Europe," *Agriculture, Ecosystems and Environment*, vol. 130, no. 1-2, pp. 67–74, 2009.

[21] D. Karahasanovic, A. Avdic, and M. Cinjarevic, "Improving of sustainable development indicator with special focus on transition countries," *Technics Technologies Education Management*, vol. 5, no. 4, pp. 760–772, 2010.

[22] A. Vallega, "The role of culture in island sustainable development," *Ocean and Coastal Management*, vol. 50, no. 5-6, pp. 279–300, 2007.

[23] S. Y. Chen, *The Variable Fuzzy Sets Theory Model and Its Application*, Dalian University of Technology Press, Dalian, China, 2009.

[24] S. Y. Chen and J. M. Hu, "Variable fuzzy assessment method and its application in assessing water resources carrying capacity," *Journal of Hydraulic Engineering*, vol. 37, no. 3, pp. 264–277, 2006 (Chinese).

[25] M. Gai and L. Zhou, "A study of the coordinated development of marine resources environment and economy of Liaoning Province using a variable fuzzy recognition model," *Resources Science*, vol. 33, no. 2, pp. 356–363, 2011 (Chinese).

[26] A. van Zeijl-Rozema, R. Cörvers, R. Kemp, and P. Martens, "Governance for sustainable development: a framework," *Sustainable Development*, vol. 6, no. 16, pp. 410–421, 2008.

[27] W. Y. Niu and Y. X. Lu, *Overview of China's Sustainable Development*, Science Press, Beijing, China, 2007.

[28] H. C. Zhou, G. H. Zhang, and G. L. Wang, "Multi-objective decision making approach based on entropy weights for reservoir flood control operation," *Journal of Hydraulic Engineering*, vol. 38, no. 1, pp. 100–106, 2007 (Chinese).

[29] D. F. Ye, "Relativity and sustainable development," *Chinese Geographical Science*, vol. 14, no. 1, pp. 75–81, 2004.

[30] W. X. Luan and N. Shen, "The research on social economic support system of sustainable development in changshan," *Archipelagos Pacific Journal*, vol. 10, pp. 65–75, 2005 (Chinese).

[31] Z. Tu, S. L. Yang, D. Hu, and D. Zhao, "Evaluation method of sustainable development capability of marine economy and application in Fujian Province," *Marine Environmental Science*, vol. 30, no. 6, pp. 819–822, 2011 (Chinese).

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 323072, 10 pages
http://dx.doi.org/10.1155/2013/323072

Research Article

Vision Target Tracker Based on Incremental Dictionary Learning and Global and Local Classification

Yang Yang, Ming Li, Fuzhong Nian, Huiya Zhao, and Yongfeng He

School of Computer and Communication, Lanzhou University of Technology, Lan Zhou 730050, China

Correspondence should be addressed to Ming Li; lim3076@163.com

Received 28 February 2013; Accepted 2 April 2013

Academic Editor: Yong Zhang

Based on sparse representation, a robust global and local classification algorithm for visual target tracking in uncertain environment was proposed in this paper. The global region of target and the position of target would be found, respectively by the proposed algorithm. Besides, overcompleted dictionary was obtained and updated by biased discriminate analysis with the divergence of positive and negative samples at current frame. And this over-completed dictionary not only discriminates the positive samples accurately but also rejects the negative samples effectively. Experiments on challenging sequences with evaluation of the state-of-the-art methods show that the proposed algorithm has better robustness to illumination changes, perspective changes, and targets rotation itself.

1. Introduction

Visual target tracking in uncertain environment is an important component in the field of computer vision [1]. In uncertain scene, the negative impact on quality of target is mainly caused by occlusion, pose changes, significant illumination variations, and so on. Therefore, discrimination method with a strong robustness against the target and environment changes is required for accurate tracking. Visual target tracking can be treated as a binary classification problem between targets and backgrounds, target candidate set of which is established by affine transformation, and classifier is then used to discriminate the target from candidate set [2]. Therefore, classifier should be not only well discriminated to targets but also capable of rejecting the discrimination of background feature and even has better robustness to occlusions, pose changes, and illumination variations.

In this paper, an incremental tracking algorithm was proposed for resolving the target appearance variations and occlusions problems. The system chart as Figure 1. Object is represented with global and local sparse representation, and tracking task is formulated as sparse representation binary classification problem with dictionary incremental learning. For object representation, targets are treated as positive samples, whereas backgrounds are treated as negative samples. Then positive and negative samples are used to establish the discriminatory dictionary, where target appearance model is treated as linear combination with discriminatory dictionary and sparse coding. In the first frame, targets are affine-transformed to affine transformation subspace, where the target is found with the minimum reconstruction error. Global classifier with sparse representation is established to determine the global region of target from center-point collection, while sparse representation local classifier is used to set up discrimination to find the target location from global region. As we know, the appearance of the target itself and the external scenes vary in real time, so dictionary needs to be updated with features of the next frame by incremental learning to ensure tracking result accurately.

The rest of the paper is organized as follows. Section 2 reviews the related works. Section 3 proposes target motion model and sparse representation globe classifier and local classifier algorithm. Furthermore, dictionary learning and

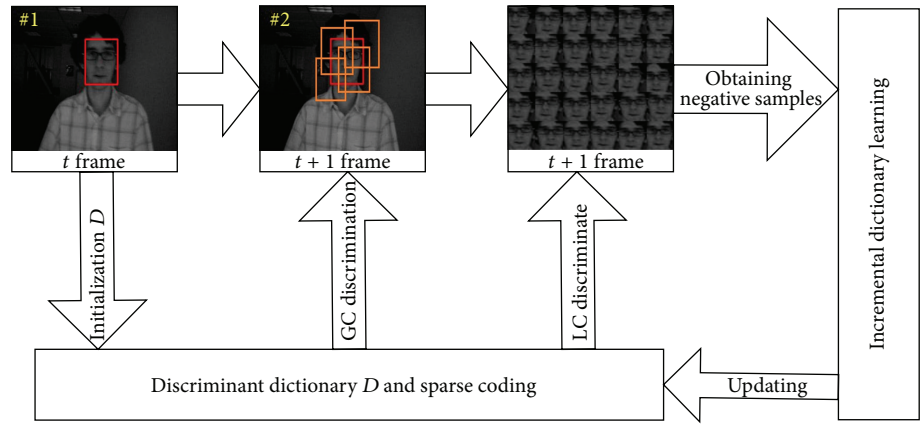


Figure 1: Diagram of targets tracking via sparse representation global and local classifier.

incremental updating are introduced in Section 4. Section 5 reports the experimental results. Finally, the conclusions are summarized in Section 6.

2. Related Works

Currently, according to the target appearance model, target tracking methods can be divided into two categories: generated and discriminative model methods. The generated model method use appearance model to replace the target observed template, tracking result get from the highest similarity search with the appearance model of the area. For example, the mean-shift [3] and the incremental tracking [4]. In [4], in order to make the algorithm adapt to the real-time changes of target appearance effectively, the target appearance models are incremental learned by a group of low-dimensional subspaces. Discriminative model method: cast the tracking as a binary classification problem. Tracking is formulated as finding the target location that can accurately separate the target from the background. In [5], online multiple instance learning methods improve the robustness of the target tracking system to the influence of occlusion. In [6], visual object tracking is construed as a numerical optimization problem and applies cluster analysis to the sampled parameter space to redetect the object and renew the local tracker. In [7], an ensemble of weak classifiers is trained online to distinguish between the object and the background, and the weak classifiers are combined into a strong classifier using AdaBoost; then, the strong classifier is used to label pixels in the next frame as either belonging to the object or the background.

Some scholars have access to a stable target tracking system, which takes advantage of sparse representation classification model for target tracking. In [8], the set of trivial template are constructed with occlusion and corruption, target candidate is sparsely represented by target and trivial templates at new frames, then, the smallest projection error is taken to find target during tracking. In [9], sample-based adaptive sparse representation is proposed to address partial occlusion or deterioration; object representations are described as sample basis with exploiting L1-norm minimization. In [10], the paper proposed a dynamic group sparsity and two-stage sparse optimization to jointly minimize the target reconstruction error and maximize the discriminative power. In [11], tracking is achieved by minimum error bound and occlusion detection, the minimum error bound is calculated for guiding particle resampling, and occlusion detection is performed by investigating the trivial coefficients in the L1-norm minimization.

In addition, to resolve the problems that the overcompleted dictionary cannot discriminate sufficiently, Xuemei utilized target template to structure dictionary $[T, I, -I]$ [9, 11]; I and $-I$ are not related unit matrix, and T is a small piece of target templates. In [12], the use of the learning obtained in the sparse representation dictionary is more effective than a preset dictionary. In [13] proposed a dictionary learning function, the dictionary was obtained by K-SVD algorithm and linear classifier. In [14], dictionary learning problem is deemed as optimization of a smooth nonconvex over convex sets and proposed an iterative online algorithm that solves this problem by efficiently minimizing at each step a quadratic surrogate function of the empirical cost over the set of constraints. On this basis, paper [15] proposes a new discriminative DL framework by employing the Fisher discrimination and criterion to learn a structured dictionary.

3. Sparse Representation Global and Local Classifier

3.1. Motion Model of Targets. We denote affine transformation parameters $X_t = (x, y, s, r, \theta, \lambda)$ as target state in frame t, where X and Y are coordinates of center point, s is change of scale, r is bearing rate, θ is rotation angle, and λ is angle of inclination. The motion model of the object through

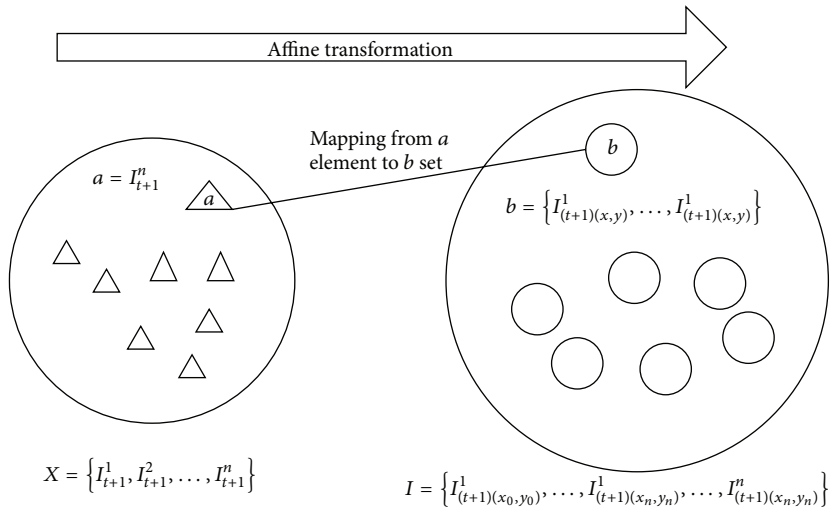


Figure 2: Affine transformation from X to I.

the transfer of the probability state of the affine transformation parameters is obtained, the function of motion model is as follows:

$$P(X_{t+1} \mid X_t) = N(X_{t+1} \mid X_t, \sigma), \quad (1)$$

where $N(X_{t+1} \mid X_t)$ is modeled independently by a Gaussian distribution, σ is a covariance diagonal matrix, and the elements of the diagonal matrix are the variance of each of the affine parameters. $\{X^1_{t+1}, X^2_{t+1}, \ldots, X^n_{t+1}\}$ is a group of affine parameter sets which are randomly generated by function (1), in current frame, and $\{I^1_{t+1}, I^2_{t+1}, \ldots, I^n_{t+1}\}$ is area of the target that may occur (candidate image area) which can be constructed by affine transformation from $\{X^1_{t+1}, X^2_{t+1}, \ldots, X^n_{t+1}\}$. Then, find the area of target from candidate image by sparse representation classifier; the classifier is trained by using previous tracking result.

3.2. Sparse Representation Classifier. Wright et al. [16] proposed the sparse representation-based classification (SRC) method for robust face recognition (FR). We denote $A = [A_1, A_2, \ldots, A_C]$ as the set of original training samples, where A_i is the subset of the training samples from class i. c is class numbers of subjects, and y is a testing sample. The procedures of sparse representation classifier are as follows:

$$\widehat{a} = \arg\min_a \left\{ \|y - A\alpha\|_2^2 + \gamma\|\alpha\|_1 \right\}, \quad (2)$$

where γ is a scalar constant,

classification via

$$\text{identity}(y) = \arg\min_i \{e_i\}, \quad (3)$$

where $e_i = \|y - A_i\widehat{\alpha}_i\|$, $\widehat{\alpha}_i = [\widehat{\alpha}_1; \widehat{\alpha}_2; \ldots; \widehat{\alpha}_c]$ and $\widehat{\alpha}_i$ is the coefficient vector associated with class i.

3.3. Sparse Representation Global and Local Classifier. We divided the set of target states $X_t = (x, y, s, r, \theta, \lambda)$ into two parts: $X_t = (x, y)$ and $X_{t,(x,y)} = (s, r, \theta, \lambda)$, $X_t = (x, y)$ is center coordinates and $X_{t,(x,y)} = (s, r, \theta, \lambda)$ is local state in time t, $\{I^1_{t+1}, I^2_{t+1}, \ldots, I^n_{t+1}\}$ is obtained by affine transform from $X_t = (x, y)$, $I^i_{(t+1)(x,y)}$ is obtained by affine transform from $X_{t,(x,y)} = (s, r, \theta, \lambda)$, and the relationship between two sets is as shown in Figure 2.

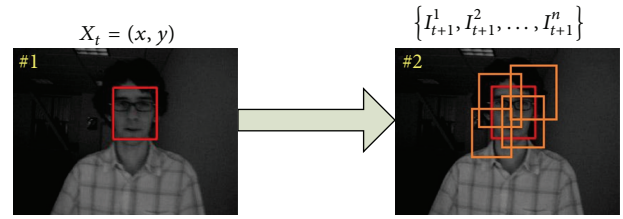


Figure 3: Image set via affine transformation.

Each element of the set I can be obtained from the affine transformation of $X_t = (x, y, s, r, \theta, \lambda)$; usually, the element numbers in set I are very large, and computational cost for discriminating the set immediately is the key issue. All of the subsets b in set I can be obtained by the affine transformation of element a in set X illustrated in Figure 3. In order to reduce the computational cost, search element a from set X first, and, then, search element b from the subset that is mapping of element a. However, that implies the need for training the two classifiers to role set X and a collection of I, and the computational cost for classifiers is raised once again. In sparse representation classifier models, the method updating completed dictionary can achieve the purpose of training multiple classification and then reduce the computation cost for the classifiers trained.

The X set constructed by center-coordinates affine transformation,in which, most of elements containing numerous negative samples features. Figure 3 shows that the classified algorithm for set X is equivalent to sparse representation global classifier, namely, SRGC. Considering target in two

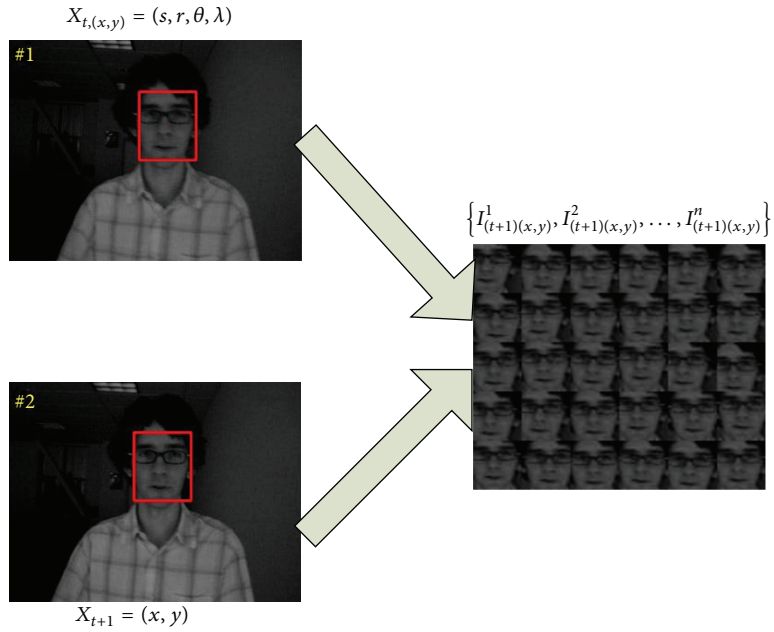


Figure 4: Obtaining target set of states.

frames with maximum likelihood, nonzero entries in the sparse coding are more concentrated in the same position; we add a constraint $\|\widehat{\alpha}_{\mathrm{GC}} - X\|_2^2$ in reconstruction error function to make sure of the maximum likelihood at sparse coding in current frames and prior to the tracking results. We define the metric for classification as follows:

$$\widehat{\alpha}_{\mathrm{GC}} = \arg\min_{a_{\mathrm{GC}}} \left\{ \|y - D\alpha_{\mathrm{GC}}\|_2^2 + \gamma \|\alpha_{\mathrm{GC}}\|_1 \right\},$$
$$e = \|y - D\widehat{\alpha}_{\mathrm{GC}}\|_2^2 + \omega \|\widehat{\alpha}_{\mathrm{GC}} - X\|_2^2, \tag{4}$$

where γ is a scalar constant, ω is a preset weight coefficient, α_{GC} is coding coefficient vector for determining the location of the center coordinates, X is coefficient vector and prior to the tracking results. The sparse representation global classifier is made by (3).

In current frame, fix the target center-point, then the set of target local status is constructed by affine transformation with the last tracking result, illustrated in Figure 4. Taking into account that most target features are imprisoned in elements of set I, we consider that discrimination in this part can be equivalent to sparse representation local classifier, as SRLC. The Objective function can be transformed into coding function over local dictionary by adding constraint $\|\alpha_{\mathrm{LC}} - X\|_2^2$ in. We add coding discriminant fidelity term $\|\alpha_{\mathrm{LC}}\|_1$ in reconstruction error function to ensure that the target is the most sparse coding in the local dictionary. We define the function for classification as follows:

$$\widehat{a}_{\mathrm{LC}} = \arg\min_{a_{\mathrm{LC}}} \left\{ \|y - D\alpha_{\mathrm{LC}}\|_2^2 + \gamma_1 \|\alpha_{\mathrm{LC}}\|_1 + \gamma_2 \|\alpha_{\mathrm{LC}} - X\|_2^2 \right\},$$
$$e = \|y - D\widehat{\alpha}_{\mathrm{LC}}\|_2^2 + \gamma_1 \|\alpha_{\mathrm{LC}}\|_1. \tag{5}$$

The sparse representation local classifier is made by (3). The proposed algorithm is summarized in Algorithm 1.

4. Learning and Incremental Updating for Dictionary

According to the aforementioned code and discriminant function mentioned in last section, the coefficient of α_{GC} and α_{LC} could not have prominent sparsity if overcompleted dictionary has bad discriminant results as well; all of the samples could probably be chosen as the sparsest code, which is bad for the classification performance of reconstruction error function.

Therefore, biased discriminant analysis [17] (BDA) method was introduced into the dictionary learning function in this paper, taking effect for objective and opposite for nonobjective. So the dispersity expressions of plus and minus samples of BDA (S_+ and S_-) are as follows:

$$S_+ = \sum_{i=1}^{N_+} (y_i^+ - u_+)(y_i^+ - u_+)^T,$$
$$S_- = \sum_{j=1}^{N_-} (y_j^- - u_+)(y_j^- - u_+)^T. \tag{6}$$

N_+ and N_- are the total number of plus and minus samples, respectively; y_i^+ and y_j^- are the ith and jth element of plus set $\{y_1^+, y_2^+, \ldots, y_{N_+}^+\}$ and minus set $\{y_1^-, y_2^-, \ldots, y_{N_+}^-\}$; u_+ is the mean value of plus sample set.

4.1. Dictionary Learning Using Biased Discriminant Analysis (BDA). Give a dictionary $D = [d_1, d_2, \ldots, d_n]$, where d_i is

Input: I_t is the tracking result of prior frame, $\{I_{t+1}^1, I_{t+1}^2, \ldots, I_{t+1}^n\}$ is set of candidate samples credible positions of the center-point coordinates in next frame. D is Over-complete dictionary, T is frame numbers.

(1) for $t = 1 : T$
(2) SRGC
calculate $\widehat{a}_{\mathrm{GC}} = \arg\min_{a_{\mathrm{GC}}} \left\{ \left\| y - D\alpha_{\mathrm{GC}} \right\|_2^2 + \gamma \left\| \alpha_{\mathrm{GC}} \right\|_1 \right\}$
(3) calculate identity $(y) = \arg\min_i \{e_i\}$; where $y = \{I_{t+1}^1, I_{t+1}^2, \ldots, I_{t+1}^n\}$,
(4) obtaining the center point (x, y) from $I_{(t+1)}^i$
(5) obtaining the $\left\{ I_{(t+1)(x,y)}^1, I_{(t+1)(x,y)}^2, \ldots, I_{(t+1)(x,y)}^n \right\}$ by (x, y) and Affine transformation of I_t
(6) SRLC to $\left\{ I_{(t+1)(x,y)}^1, I_{(t+1)(x,y)}^2, \ldots, I_{(t+1)(x,y)}^n \right\}$;
calculate $\widehat{a}_{\mathrm{LC}} = \arg\min_{a_{\mathrm{LC}}} \{ \left\| y - D\alpha_{\mathrm{LC}} \right\|_2^2 + \gamma_1 \left\| \alpha_{\mathrm{LC}} \right\|_1 + \gamma_2 \left\| \alpha_{\mathrm{GC}} - X \right\|_2^2 \}$;
(7) calculate identity $(y) = \arg\min_i \{e_i\}$, where $e = \left\| y - D\widehat{a}_{\mathrm{LC}} \right\|_2^2 + \gamma_1 \left\| \alpha_{\mathrm{LC}} \right\|_1$

Output: $I_{(t+1)(x,y)}^i$

Algorithm 1: Algorithm of Fisher discrimination dictionary learning.

an n-dimensional vector $d_i = [d_i^1, d_i^2, \ldots, d_i^n]^T$ and d_i^j is the jth element of ith vector which is called atom of dictionary. $A = [A_+, A_-]$ is the training sample set, where A_+ and A_- are the characterized and noncharacterized samples for objective, which are also called plus and minus samples.

However, for objective tracing, only the region of objective is interested, so the background characteristics, noises, occlusions, and so on are regarded as noncharacterized samples A_-. Let $X = [x_1, x_2, \ldots, x_n]$ be the code coefficient vector of sample set A of dictionary D, provided that the tested sample set can be denoted as $A \approx DX$. Furthermore, dictionary learning function is

$$J_{(D,X)} = \arg\min_{(D,X)} \left\{ \left\| A_+ - DX \right\|_F^2 + \lambda_1 \|X\|_1 + \lambda_2 f(X) \right\}, \quad (7)$$

where $\|A_+ - DX\|_F^2$ is the discriminant fidelity term which is only used for A_+, since the interesting thing in objective tracing is only the area of objective. $\|X\|_1$ is $l1$-norm sparse constraint term, and $f(X)$ is the discriminant constraint with respect to coefficient vector X.

According to BDA discriminant rule, let $f(X)$ be $\mathrm{tr}(S_+) - \mathrm{tr}(S_-)$. Let $\|X\|_F^2$ be added into $f(X)$ as a relaxed term because the function $f(X)$ is nonconvex and unstable, therefore

$$f(X) = \mathrm{tr}(S_+) - \mathrm{tr}(S_-) + \eta \|X\|_F^2, \quad (8)$$

where η is the control variable. Furthermore, the proposed BDDL method can be formed as

$$J_{(D,X)} = \arg\min_{(D,X)} \left\{ \left\| A_+ - DX \right\|_F^2 + \lambda_1 \|X\|_1 + \lambda_2 \left(\mathrm{tr}(S_+) - \mathrm{tr}(S_-) \right) + \eta \|X\|_F^2 \right\}. \quad (9)$$

Similar to [15], (D, X) is nonconvex for function J which is the convex function on set X when D is already known and also the convex function on set D when X is already known. So, J is in fact a biconvex function on sets D and X.

4.2. Dictionary Incremental Updating. A new plus and minus samples set $Y_{\mathrm{new}}^+ = \{y_1^+, y_2^+, \ldots, y_{M_+}^+\}_{\mathrm{new}}$ and $Y_{\mathrm{new}}^- = \{y_1^-, y_2^-, \ldots, y_{M_+}^-\}_{\mathrm{new}}$ can be obtained according to current objective tracing result, the mean value of which is $u_{\mathrm{new}}^+ = (1/m_+) \sum_{i=1}^{m_+}{}_{\mathrm{new}} y_i^+$ and $u_{\mathrm{old}}^+ = (1/n_+) \sum_{i=1}^{n_+}{}_{\mathrm{old}} y_i^+$, respectively; m_+ and n_+ are the number of new and old samples. Furthermore, the weighted mean of these two mean values of plus sample set is

$$u_+ = \frac{n_+ u_{\mathrm{old}}^+ + m_+ u_{\mathrm{new}}^+}{n_+ + m_+}. \quad (10)$$

Similarly, the new dispersity expression of plus sample set using weighted mean value u_+ is

$$S_{\mathrm{new}}^+ = \sum_{i=1}^{M_+} \left({}_{\mathrm{new}} y_i^+ - u_+ \right) \left({}_{\mathrm{new}} y_i^+ - u_+ \right)^T. \quad (11)$$

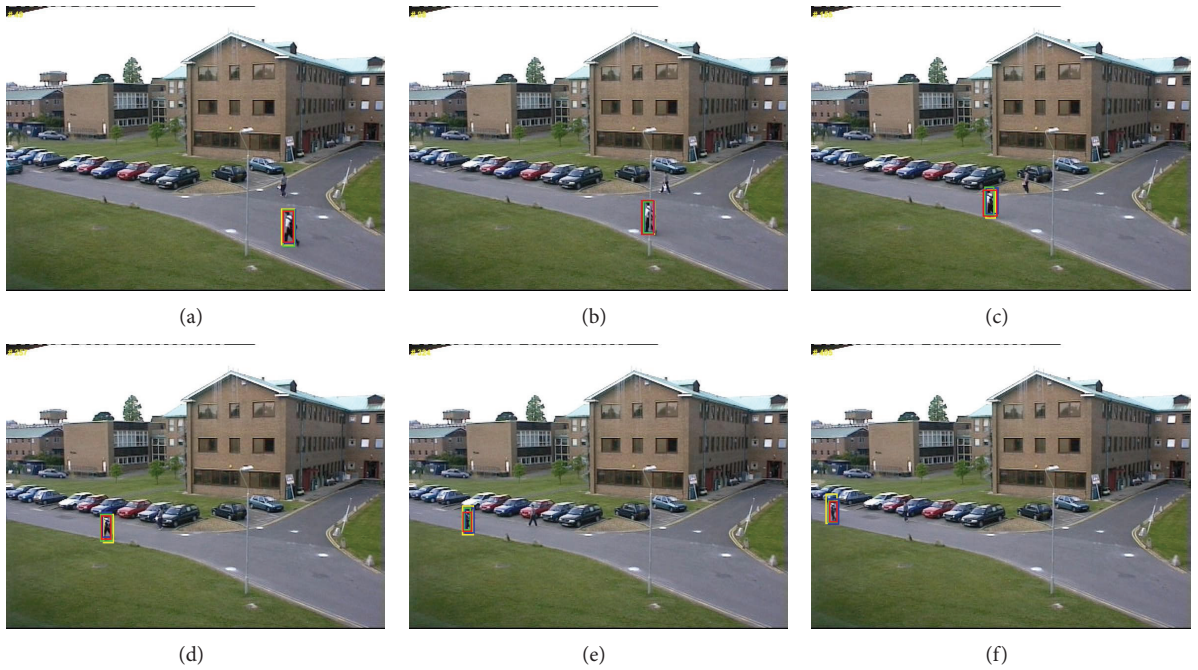
The dispersity expression of the updated plus sample S_+ is

$$S_+ = S_{\mathrm{old}}^+ + S_{\mathrm{new}}^+ + \frac{n_+ m_+}{n_+ + m_+} \left(u_{\mathrm{new}}^+ - u_{\mathrm{old}}^+ \right) \left(u_{\mathrm{new}}^+ - u_{\mathrm{old}}^+ \right)^T. \quad (12)$$

S_{old}^+ is the old dispersity expression of plus sample set.

However, we need just refused-discriminant to negative samples, instead of discriminant it in real time, then the dispersity of negative samples is as follows:

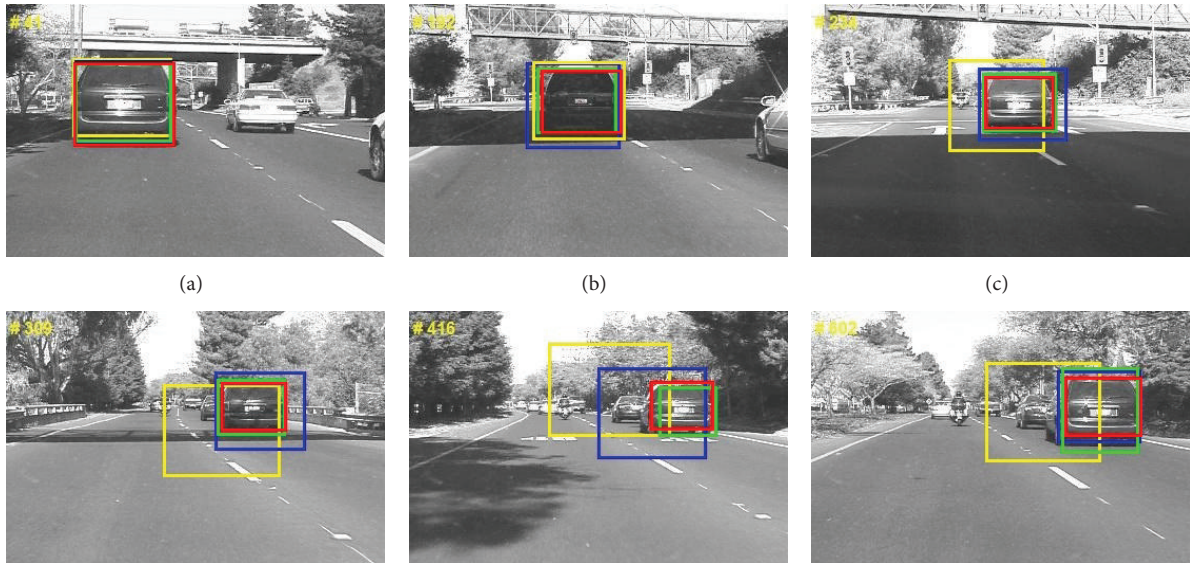
$$S_- = S_{\mathrm{old}}^- + S_{\mathrm{new}}^-,$$
$$S_{\mathrm{new}}^- = \sum_{j=1}^{M_-} \left(y_j^- - u_+ \right) \left(y_j^- - u_+ \right)^T. \quad (13)$$



(a) (b) (c)

(d) (e) (f)

Figure 5: Tracking results of the PETS01D1Human1 sequence (MIL is yellow, IVT is blue, L1 is green, and our tracker is red).



(a) (b) (c)

(d) (e) (f)

Figure 6: Tracking results of the Car4 sequence (MIL is yellow, IVT is blue, L1 is green, and our tracker is red).

If we take S_- and S_+ into consideration, then the updated function $f(X) = \mathrm{tr}(S_+) - \mathrm{tr}(S_-) + \eta\|X\|_F^2$ can be represented as

$$f(X) = \mathrm{tr}\left(S_{\mathrm{old}}^{+} + S_{\mathrm{new}}^{+} + \Psi\right) - \mathrm{tr}\left(S_{\mathrm{old}}^{-} + S_{\mathrm{new}}^{-}\right) + \eta\|X\|_F^2, \tag{14}$$

where $\Psi = (n_+ m_+/(n_+ + m_+))(u_{\mathrm{new}}^{+} - u_{\mathrm{old}}^{+})\ (u_{\mathrm{new}}^{+} - u_{\mathrm{old}}^{+})^T$.

According to (14), fix D_{old}, and then compute X; D is reconstructed by obtaining X that is updating D, where X is not used for discriminant immediately and is just reconfigurable coding coefficient matrix:

$$J_{(X)} = \arg\min_{X} \left\{\left\|A_+ - D_{\mathrm{old}}X\right\|_F^2 + \lambda_1\|X\|_1 + \lambda_2 f(X)\right\}, \tag{15}$$

where $f(X) = \mathrm{tr}(S_{\mathrm{old}}^{+} + S_{\mathrm{new}}^{+} + \Psi) - \mathrm{tr}(S_{\mathrm{old}}^{-} + S_{\mathrm{new}}^{-}) + \eta\|X\|_F^2$, D_{old} is the old dictionary; A_+ is the joint matrix of samples in current and previous frames, which is represented

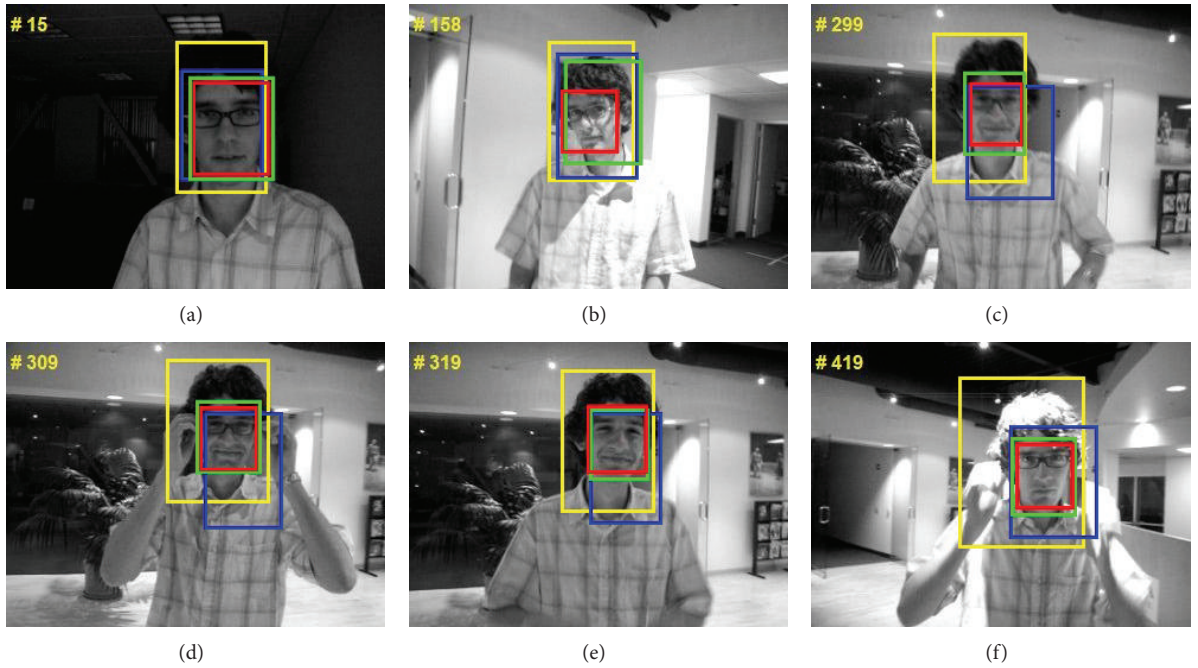


Figure 7: Tracking results of the David Indoor sequence (MIL is yellow, IVT is blue, L1 is green, and our tracker is red).

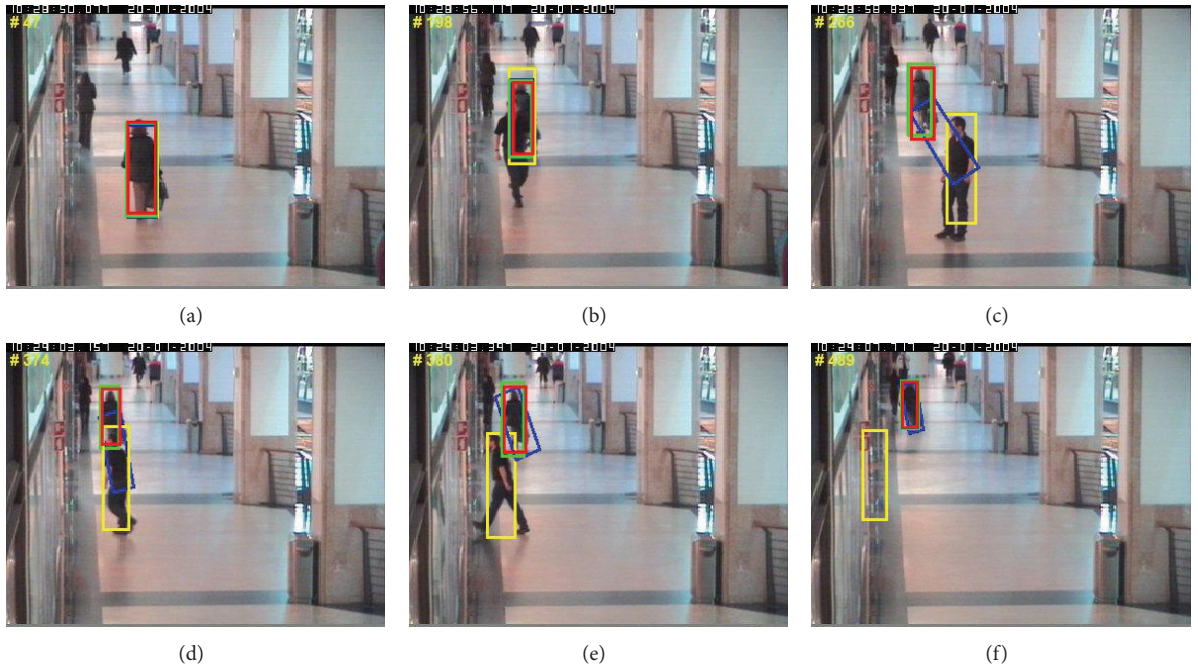


Figure 8: Tracking results of the OneLeaveShopReenter2cor sequence (MIL is yellow, IVT is blue, L1 is green, and our tracker is red).

as $A_+ = [Y^+_{\text{new}_{t-1}}; Y^+_{\text{new}_t}]$, $Y^+_{\text{new}_{t-1}} = \{y^+_1, y^+_2, \ldots, y^+_{M_+}\}_{\text{new}_{t-1}}$ and $Y^+_{\text{new}_t} = \{y^+_1, y^+_2, \ldots, y^+_{M_+}\}_{\text{new}_t}$. Then function $J_{(D,X)}$ could be rewritten as

$$J_{(D)} = \arg\min_D \|A_+ - DX\|_F^2. \tag{16}$$

In first frames we need initialization; the target is framed manually, the Y_0^+ is set of initial moment positive samples, Y_0^- is the set of initial moment negative sample, u_0^+ is the mean value of initial moment positive sample, compute S_+ and S_- by (6). We initialize all atoms p of dictionary D as random vector with $l2$-norm, solve X by solving (15), and then fix X and solve D by solving (16).

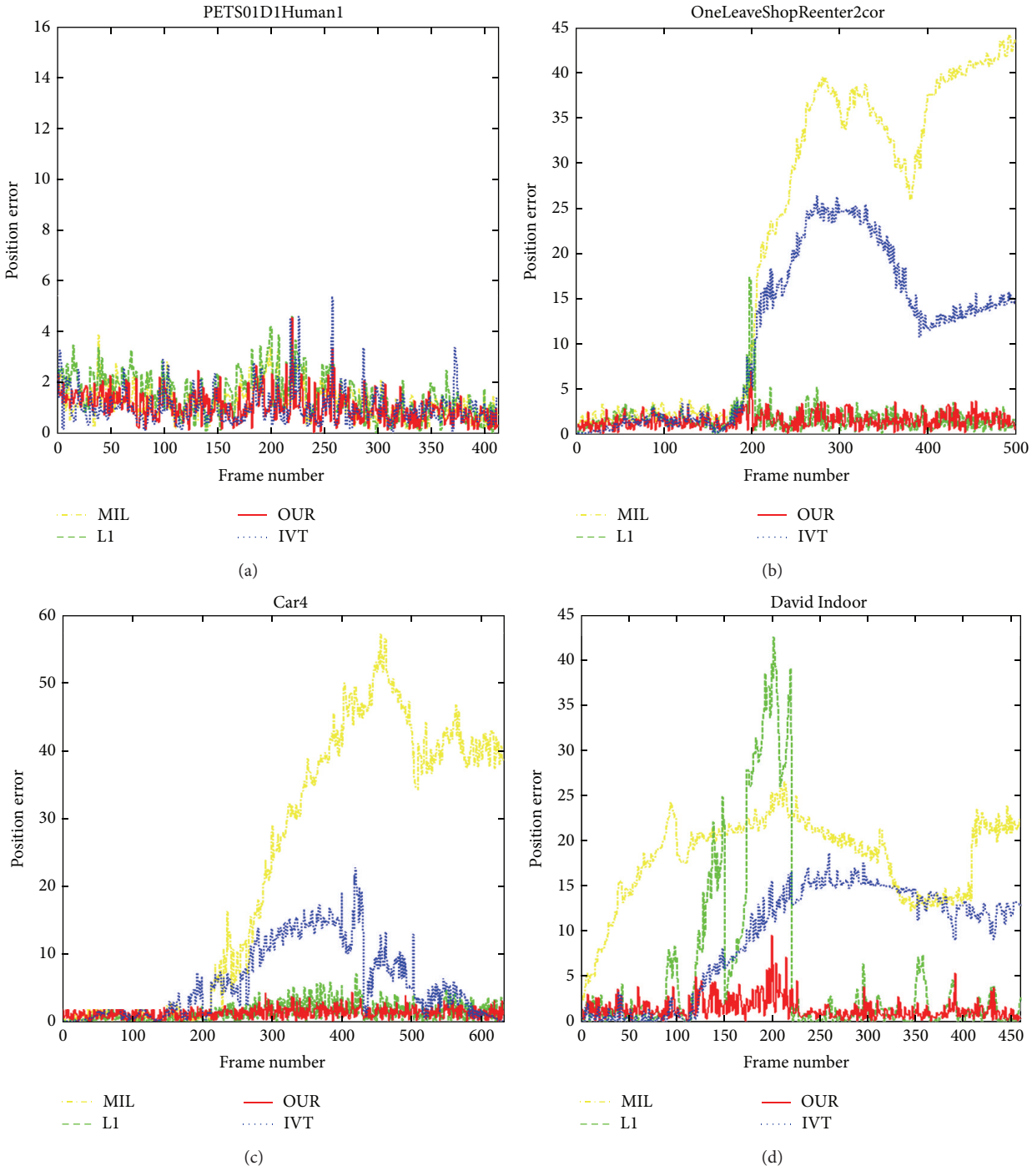


Figure 9: Position error plots of the tested sequences.

5. Experiments

Experiments were performed with four video sequences, which included occlusion, illumination variation, appearance variation and other corruption factors. In the experiment, target location of the first frame is framed manually, and initial dictionaries were randomly generated, and track results were to be released as rectangular boxes. Figures 5, 6, 7 and 8 show a representative sample of tracking results. finally, target tracking method of this paper contrasts with incremental visual tracking (IVT) [4], multiple instance Learning (MIL) [5] and L1 tracker (L1) [8]. To further evaluate the proposed method, each method applies to four video sequences and then compares the track results, that need to be evaluated qualitatively and quantitatively. The paper uses gray histogram as a way of presentation characteristics of

Table 1: Analysis of location errors.

	IVT			L1			MIT			Proposed		
	Max	Mean	Std	Max	Mean	Std	Max	Mean	Std	Max	Mean	Std
OneLeaveShopReenter2cor	26.32	11.26	8.82	7.33	2.46	**1.72**	44.21	21.67	16.68	*6.65*	*2.02*	1.86
David Indoor	18.76	9.43	5.98	42.59	57.64	9.93	18.76	20.51	4.67	*6.71*	*1.45*	*1.21*
Car4	22.71	5.89	2.22	8.71	2.89	**1.44**	59.54	23.07	6.37	*7.41*	*1.36*	1.45
PETS01D1Human1	5.98	**0.93**	**0.87**	**5.09**	1.68	0.97	6.12	1.16	1.14	5.95	1.39	0.91

each method, and this is easy to test and judge the sensitivity of corruption. Besides, each run of the experiment used the same starting position, initial frame, frame number of video sequences, and software environment.

5.1. Qualitative Comparison. The test sequences, PETS01D1Human1, show that a person went to the extreme left side from the lower right corner of the screen, which telephone poles will cause short-term shelter to it. Tracking results are shown in Figure 5; the image frames are # 49, # 88, # 155, # 257, # 324, and # 405. All methods can effectively track the target, and the tests show that in the circumstances of the same light intensity, the same camera angle, and slight shelter, all methods can effectively track the target. It also indicates that the proposed method in the paper and the contrastive method are effective target tracking algorithms.

In the Car4 sequence, when the cars pass through the bridge and the shade, intensity of illumination altered obviously. Tracking results are shown in Figure 6; and the image frames are #41, #792, #234, #309, #416, and #602. When the cars go through the bridge, MIL will be ineffective significantly, but will not lose target; IVT will also be ineffective, but it can snap back. The method in this paper and L, compared with MIL and IVT, can locate the target accurately.

In the David Indoor sequence, the degeneration is include twice illumination change, expression change, and partial occlusion. The track result was shown in Figure 7, and the image frames are #15, #158, #299, #309, #319, and #419. The method in this paper can locate the target accurately; contrastively, the result of L1 is ineffective. The reason is that the target gray histogram was changed by light intensity, that affects the feature of image gray histogram; the methods of MIL and IVT may be more sensitive to the effects.

The OneLeaveShopReenter2cor sequence shows a woman walking through a corridor, when a man walks by, which lead to large occlusions Clothes with similar color are the occluder. The track result was shown in Figure 8; the image frames are #47, #198, #266, #374, #380, and #489. The method in this paper and L1 can locate the target accurately. When occlusion happened, MIL put the occluder as target and missed the target; The target is similar with the occluded,and then the IVT is difficult to discriminate object and occluded.

In conclusion, both the method in this paper and L1 can locate the target accurately. And they have strong robustness for occlusions, pose changes, significant illumination variations, and so forth.

5.2. Quantitative Comparison. We evaluate the tracking performance by position error. The position Error is approximated by the distance between the central position of the tracking result and the manually labeled ground truth. Table 1 shows the statistical data of position error which includes maximum, mean and standard deviation. Figure 9 shows the errors of all four trackers.

From previous comparison results, we can see that proposed method can track the target more accurately in video sequence OneLeaveShopReenter2cor, David Indoor, and Car4 than other methods. The max, mean, and standard deviation of position errors are smaller than IVT and MIT. Therefore, in complex environment, our method has a better robustness. Comparing with L1, the result of tracking to sequence OneLeaveShopReenter2cor and Car4 shows that L1 has higher stability in the scene where illumination did not change significantly. However, the standard deviation of position error of L1 tracker in those sequences is smaller than proposed method, that L1 update capability is less than proposed method, when grayscale histogram distribution changed greatly. The dictionary in L1 is constructed by target template, so robustness of learned dictionary is better than it.

6. Conclusion

In this paper, a tracking algorithm was proposed based on sparse representation and dictionary learning. Based on biased discriminant analysis, we proposed an effective Incremental learning algorithm to construct overcompleted dictionary. Positive and negative samples are obtained during tracking process and are used for updating discriminant dictionary by biased discriminant analysis. Then we proposed sparse representation global and local classification for set of central points and set of local states. Compared to the state-of-the-art tracking methods, the proposed algorithm improves the discriminating performance of completed dictionary and the adaptive ability of appearance model. It has a strong robustness to illumination changes, perspective changes, and targets rotation itself.

References

[1] T. Bai and Y. F. Li, "Robust visual tracking with structured sparse representation appearance model," *Pattern Recognition*, vol. 45, pp. 2390–2404, 2012.

[2] F. Chen, Q. Wang, S. Wang, W. Zhang, and W. Xu, "Object tracking via appearance modeling and sparse representation," *Image and Vision Computing*, vol. 29, pp. 787–796, 2011.

[3] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564–577, 2003.

[4] D. A. Ross, J. Lim, R. S. Lin, and M. H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1–3, pp. 125–141, 2008.

[5] B. Babenko, S. Belongie, and M. H. Yang, "Visual tracking with online multiple instance learning," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR '09)*, pp. 983–990, June 2009.

[6] Z. Yin and R. T. Collins, "Object tracking and detection after occlusion via numerical hybrid local and global mode-seeking," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '08)*, pp. 1–8, June 2008.

[7] S. Avidan, "Ensemble tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 261–271, 2007.

[8] X. Mei and H. Ling, "Robust visual tracking using L1 minimization," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '09)*, pp. 1436–1443, 2009.

[9] Z. Han, J. Jiao, B. Zhang, Q. Ye, and J. Liu, "Visual object tracking via sample-based Adaptive Sparse Representation (AdaSR)," *Pattern Recognition*, vol. 44, no. 9, pp. 2170–2183, 2011.

[10] B. Liu, L. Yang, J. Huang, P. Meer, L. Gong, and C. Kulikowski, "Robust and fast collaborative tracking with two stage sparse optimization," *Lecture Notes in Computer Science*, vol. 6314, no. 4, pp. 624–637, 2010.

[11] X. Mei, H. Ling, Y. Wu, E. Blasch, and L. Bai, "Minimum error bounded efficient L1 tracker with occlusion detection," in *Proceedings of the IEEE Computer Vision and Pattern Recognition (CVPR '11)*, pp. 1257–1264, 2011.

[12] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proceedings of the IEEE*, vol. 98, pp. 948–958, 2010.

[13] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 2691–2698, June 2010.

[14] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proceedings of the 26th International Conference On Machine Learning (ICML '09)*, pp. 689–696, June 2009.

[15] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Fisher discrimination dictionary learning for sparse representation," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '11)*, pp. 543–550, 2011.

[16] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[17] J. Wen, X. Gao, Y. Yuan, D. Tao, and J. Li, "Incremental tensor biased discriminant analysis: a new color-based visual tracking method," *Neurocomputing*, vol. 73, no. 4–6, pp. 827–839, 2010.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 151520, 8 pages
http://dx.doi.org/10.1155/2013/151520

Research Article

Land Use Patch Generalization Based on Semantic Priority

Jun Yang,[1,2] Fanqiang Kong,[2] Jianchao Xi,[1] Quansheng Ge,[1] Xueming Li,[2] and Peng Xie[2]

[1] *Institute of Geographic Sciences and Natural Resources Research, CAS, Beijing 100101, China*
[2] *Liaoning Key Laboratory of Physical Geography and Geomatics, Dalian 116029, China*

Correspondence should be addressed to Fanqiang Kong; kongfankongfan@163.com

Received 31 January 2013; Accepted 25 March 2013

Academic Editor: Jianhong (Cecilia) Xia

Land use patch generalization is the key technology to achieve multiscale representation. We research patches and achieve the following. (1) We establish a neighborhood analysis model by taking semantic similarity between features as the prerequisite and accounting for spatial topological relationships, retrieve the most neighboring patches of a feature using the model for data combination, and thus guarantee the area of various land types in patch combination. (2) We establish patch features using nodes at the intersection of separate feature buffers to fill the bridge area to achieve feature aggregation and effectively control nonbridge area deformation during feature aggregation. (3) We simplify the narrow zones by dividing them from the adjacent feature buffer area and then amalgamating them into the surrounding features. This effectively deletes narrow features and meets the area requirements, better generalizes land use features, and guarantees simple and attractive maps with appropriate loads. (4) We simplify the feature sidelines using the Douglas-Peucker algorithm to effectively eliminate nodes having little impact on overall shapes and characteristics. Here, we discuss the model and algorithm process in detail and provide experimental results of the actual data.

1. Introduction

Land use generalization is a complicated process involving complex spatial and semantic relationships between land use features, and thus it is very difficult to satisfy such conditions concurrently. A significant amount of research has been conducted in this area: for example, Zongbo [1] discussed proportion image generalization, purpose image generalization, and visual image generalization in image map compilation and elaborated the compilation process on the basis of practice; Chithambaram et al. [2] integrated the data based on extracting feature skeletons; that is, secondary patches were compressed into lines or points, secondary lines were compressed into points, and evaluations were given; Ai and Wu [3] conducted neighborhood analysis using the Delaunay triangulation network and carried out a consistency correction for the shared boundary of vector patches after simplification; Ai et al. [4] applied the Delaunay triangulation network executing neighborhood analysis to retrieve neighbor patches in patch aggregation and subdivided, merged, and simplified secondary patches by generating skeleton lines using the Delaunay triangulation network; Harrie [5] established appropriate weights for various generalization constraints to solve the balance between constraining conditions and map qualification; Kulik et al. [6] proposed an ontology-oriented cartographic generalization and matched the appropriate needs for different users; Zhao et al. [7] studied the consistent update system of geospatial databases based on digital map generalization; Li et al. [8] and Huang et al. [9] discussed the area proportion of each patch after generalization and investigated patch boundary simplification, achieving constraints in the balanced area of various features in boundary simplification and attaining good adaptability; Stoter et al. [10] discussed the noncustomized automated cartographic generalization of commercial software, comprehensively considered the elevation results of man and machine, and revealed the possible differences; Qiao and Zhang [11] studied cartographic generalization in a distributed environment, which could be adapted to large quantity spatial data; Dilo et al. [12] proposed tGAP to achieve map generalization between two scales in a certain area, with large-scale maps used for generalization and small-scale maps used for constraint; Stanislawski [13] achieved automated generalization in U.S.

national hydrological datasets by deleting the corresponding features based on upstream drainage areas; Foerster et al. [14] studied the feasibility of geospatial data integration in a network service environment; Ai et al. [15] and Liu et al. [16], respectively, provided a detailed analysis and calculation models for the semantic similarity of land use data; Zhu et al. [17] applied a curve fit algorithm to line generalization and compared it with traditional algorithms.

The above studies comprehensively considered the semantic and spatial neighborhood of features when establishing an integrated model and obtained quantitative results through the corresponding weights of various parts. However, the requirements for total area of each land use type before and after land use generalization are strict, and the total area of each feature must fluctuate within a certain range. Thus, this paper prioritized the semantic neighborhood when establishing the model and took the spatial topology relationship as an auxiliary factor to determine final results relating to the same semantic neighborhoods and thus ensured the total area of each land use type optimally.

2. Analysis Model of Feature Neighborhood

2.1. Semantic Neighborhood of Features. Land use data is completely encompassed, seamless, and nonoverlapping in space, has hierarchical semantic divisions [18], and generalizes the feature set in the above premise. Land use data is divided into three-level land types as shown in Figure 1 (each layer is one level from top to bottom). Integration is difficult due to semantic diversity, so a clear generalization rule can only be developed after defining the relationship between semantics and determining the semantic neighborhood.

Land use data is often concerned with the total amount of first and second land use and is only interested in urban and rural construction land subclasses for third land use. Accordingly, this paper argues that semantic neighborhoods exist only among features at the same first land use type or that semantics are unrelated. We developed a land type sequence of semantic neighborhoods at the same level for each second and third land use. Taking arid land of the third land type, we first considered the lands with the same parent type and obtained the following sequence: arid land, irrigated land, and paddy field (see Figure 1). We then considered the relationship between the same first land types and arid land; that is, paddy field was followed by garden plot, woodland, grassland, raised path, irrigation and water conservancy land, agricultural land, and rural road (building land and other first land use types were not related to arid land semantics).

2.2. Definition of Feature Relationship in the Model. We supposed land use data as LandUseSet = $\{F1, F2, F3 \ldots Fn\}$, and SArea and DFeature respectively represented the minimum area of features in the map and the minimum distance between the features; land type name was represented by Land Name (Fn); the parent land type of feature land type (e.g., the parent land type of farmland and garden plot was agricultural land) was represented by Father[LandName (Fi)]; feature area was represented by Area (Fi); Dis (Fi, Fj) represented the minimum distance between features Fi and Fj; the spatial topology relationship between features Fi and Fj was represented by TopoRel (Fi, Fj); the semantic similarity was represented by SemRel (Fi, Fj). The values of TopoRel (Fi, Fj) and SemRel (Fi, Fj) are as follows.

(1) The values of TopoRel (Fi, Fj) were −1, 0, and 1. We first determined ColLine (Fi, Fj) (whether two features are collinear), with spaces of features Fi and Fj being adjacent if they were collinear, and thus TopoRel (Fi, Fj) = 0; otherwise we determined the relationship between Dis (Fi, Fj) and DFeature; if Dis (Fi, Fj) < DFeature, the $F1$ and $F2$ spaces were adjacent, and TopoRel (Fi, Fj) = 1; otherwise TopoRel (Fi, Fj) = −1, and the $F1$ and $F2$ spaces were unrelated.

(2) The range of SemRel (Fi, Fj) was determined by the number of land types close to Fi. As mentioned before, there were 10 land types with similar semantics (including itself); when Fi was arid land, the values of SemRel (Fi, Fj) were $0, 1, 2 \ldots 9$ in order based on the semantic neighborhood of dry land; when the semantics of features Fi and Fj were unrelated, SemRel (Fi, Fj) = −1.

2.3. Model Rules. Land type area in each administrative region should be counted before and after land use integration, so the administrative region is an independent integrated unit. The following rules were formulated under this precondition. The secondary feature dataset FeaSet{Fi} (Area (Fi) < SArea) should be obtained before integration. According to 2.2, when TopoRel (Fi, Fj) = −1, no relationship existed between Fj and Fi due to the too long distance; when SemRel (Fi, Fj) = −1, the semantics of the two features were unrelated, so aggregation treatment cannot be conducted. The model process was as follows. Step 1: retrieve feature dataset FeaSet{} based on condition (1) SemRel (Fi, Fj) = 0 and TopoRel (Fi, Fj) = 0, and the feature that had the longest shared boundary with Feature Fi was the desired one in the dataset. For example, Feature $F7$ in Figure 2(a) was a secondary feature, the dataset meeting condition (1) should be FeaSet{$F3, F4$}, and the feature with the longest boundary with $F7$ was the desired one, which was the nearest feature in the dataset ($F4$). If the dataset meeting condition (1) was empty, Step 2 was conducted: retrieve feature dataset FeaSet{} based on condition (2) SemRel (Fi, Fj) = 0 and TopoRel (Fi, Fj) = 1; the feature with the largest area in the buffer of the DFeature radius of Feature Fj was the desired one. Taking $F7$ in Figure 2(b) as an example, when the dataset meeting condition (1) was empty, the dataset meeting condition (2) was FeaSet{$F4, F3$} and consisted of two features, and the buffer of the $F7$ Buffer ($F7$) was made by taking DFeature as the radius; attention should be paid to $F3$ and $F4$ in Buffer ($F7$), with $F3$ as being the desired feature because its area was larger than that of $F4$ in Buffer ($F7$). If the nearest feature was not retrieved after the aforementioned two steps, 1 was added to the value of SemRel (Fi, Fj) for recycling, until the most neighboring feature was retrieved. If the aforementioned features were not found when the maximum of SemRel (Fi, Fj) was achieved, Feature Fi was integrated into the neighboring feature with the largest area. For example, Feature $F7$ in Figure 2(c) was finally merged

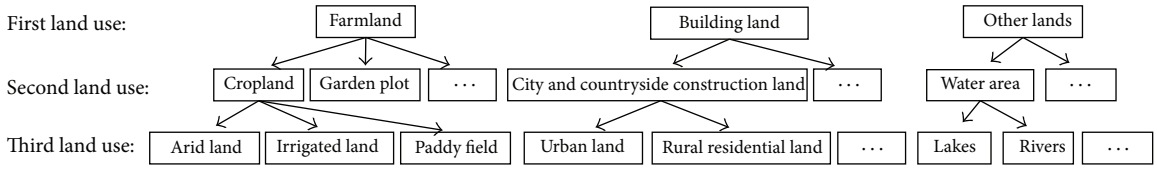


Figure 1: Hierarchical tree of land use type.

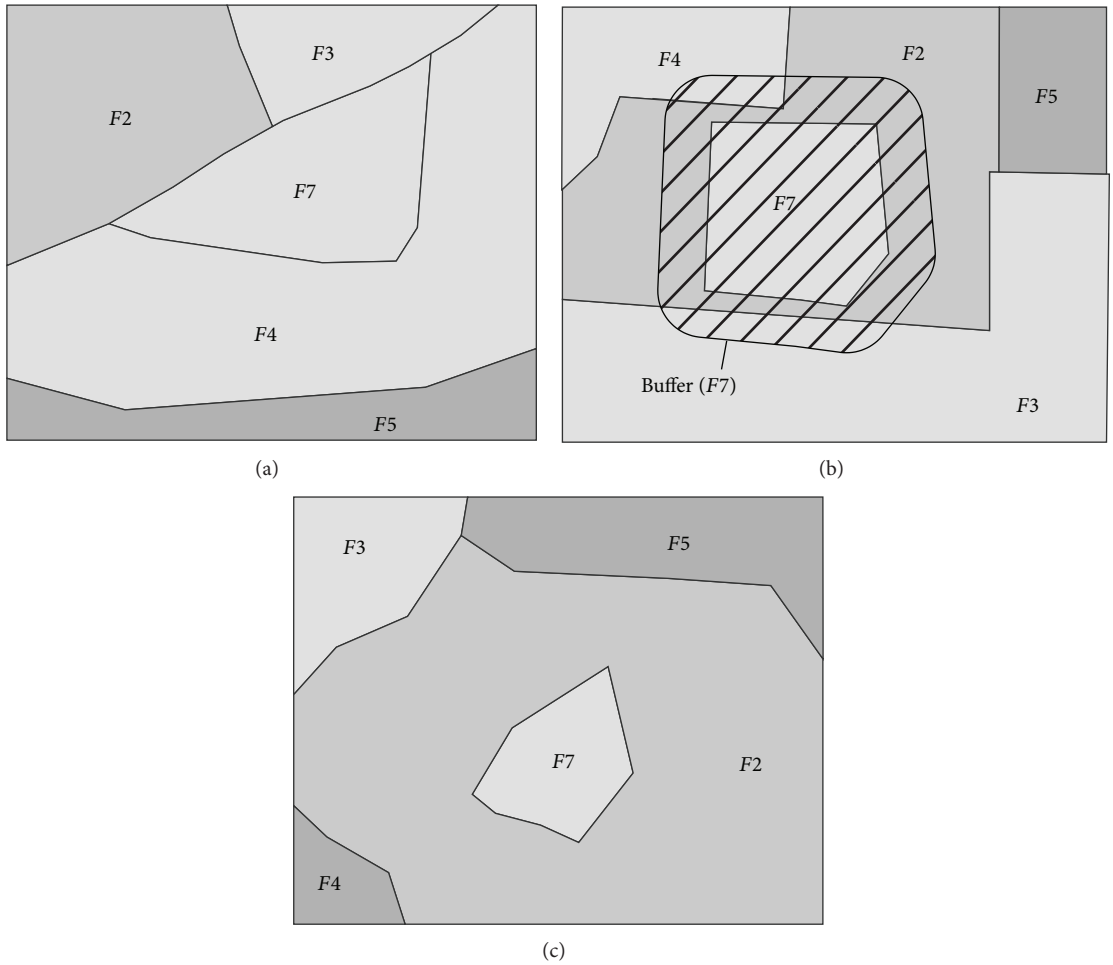


Figure 2: Neighborhood degree.

into $F2$. In the previous process, if the Fi and Fj spaces were adjacent, the amalgamation method was taken; if the Fi and Fj spaces were neighboring, the aggregation method was taken. This model determines the nearest feature of secondary features by focusing on the semantic neighborhood of features with spatial topology relationships. This modeling process was simple and the changes in each land type area were minimized during integration, and the requirements of land use integration were met. The workflow of the neighborhood analysis model is shown in Figure 3.

3. Feature Processing Algorithms

3.1. Aggregation Processing. Feature aggregation is the merging of separate features in space, and it can prevent the same type of features with short distance from being removed and avoid large changes in total land type area after integration [19]. The specific aggregation algorithm steps in buffer were as follows (taking $F1$ in Figure 4(a) as an example): (1) create the buffer of the $F1$ Buffer ($F1$) using DFeature (the minimum distance between features); (2) look for neighborhood patch $F2$ intersecting Buffer ($F1$); (3) create the buffer of the $F2$ Buffer ($F2$) by taking DFeature as the buffer radius, as shown in Figure 4(a); (4) calculate Buffer ($F1$) ∩ Buffer ($F2$) of the two buffers, and the buffer intersection of the two features (Figure 4(a)) was the grid region in the middle part; (5) calculate NodeSet$\{N1, N2, N3 \ldots Ni\}$, the node set of features $F1$ and $F2$ in the buffer intersection (black boundary in Figure 4(b)); (6) establishe patch Feature Fn using the nodes in the NodeSet, that is, the dark brown

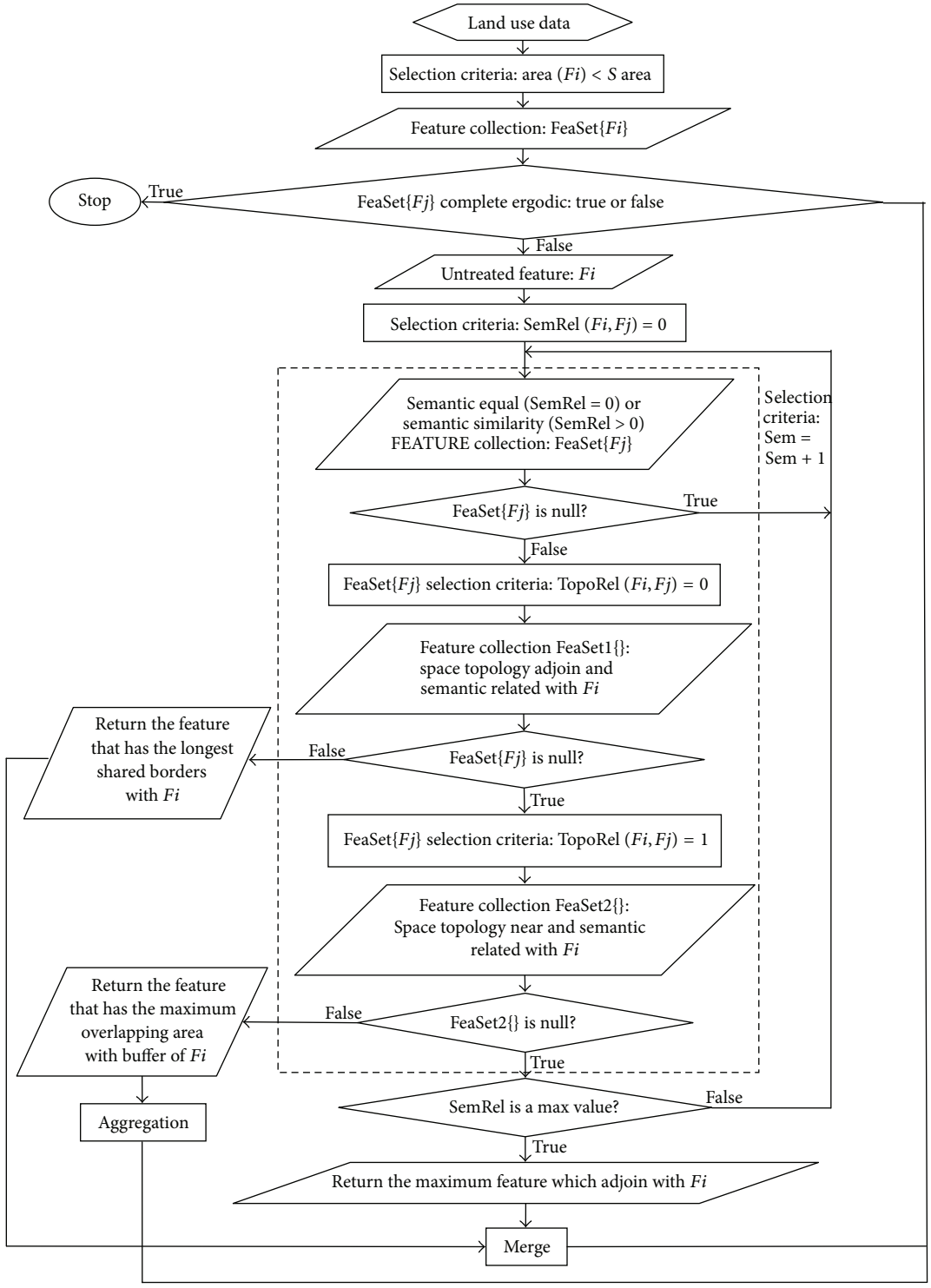


Figure 3: Model workflow.

region in the middle part of Figure 4(b); (7) merge $F1$, $F2$, and Fn to generate the feature after aggregation, as shown in Figure 4(c).

As seen from Figures 4(b) and 4(c), the feature using the buffer intersection nodes was the bridge area of separate features, which was effectively eliminated after the separate features were merged, effectively maintained the original shapes and characteristics of features, and met the requirements of integration. Attention should be paid to the feature overlapping when conducting aggregation processing by this method, and the bridge area can be directly excised for newly added features and overlapping in the bridge area.

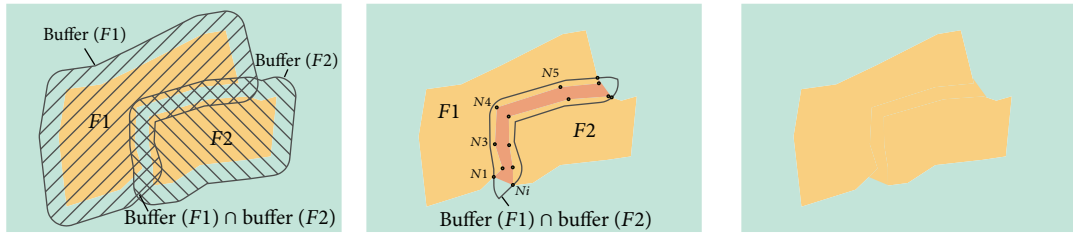


Figure 4: Process of aggregation.

3.2. Processing of Narrow Features. Narrow features in land use data mainly include railways, roads, rivers, and ditches. Simple integration or aggregation with the surrounding features is not enough because the data is long and narrow and the influence of the feature on the data cannot be eliminated by simple merger processing. We propose a buffer-based method to subdivide the narrow features according to semantic similarity and integrate the divided features into the surrounding features. The algorithm is simple and easy to implement with high efficiency.

Taking into account the semantic similarity of narrow features with spatial adjoining features at both sides, we first extracted the centerline of the narrow surface feature (such as the crimson line centerline in Figure 5(a)) and then divided the narrow feature River into upper and lower parts using the centerline (Upriver and Downriver in Figure 5(a)). Upriver was divided by $F1$, which adjoined it in space; although $F2$ and $F3$ adjoined River, in space they did not directly contact Upriver or participate in the division; while Downriver adjoined $F2$ and $F3$ in space, so it can be divided by Feature $F2$ and $F3$. We took the division of Downriver as an example to describe the processing steps of narrow features. (1) Establish buffer ($F2$) and buffer ($F3$), the buffers of features $F2$ and $F3$ adjoining Down_River in space (buffer distance was half the widest length of the narrow surface feature), and the buffers were overlapping, as shown in Figure 5(b). (2) Judge the features with neighboring semantics based on SemRel ($F2$, River), SemRel ($F3$, River), and the semantic similarity of $F2$ and $F3$ with the River. The semantics of $F2$ were more neighboring with those of the River. (3) Cut the buffers of the other features with the buffer of the feature that had neighboring semantics with the narrow feature; that is, cut buffer ($F3$) with buffer ($F2$), as shown in Figure 5(c). At this stage, there was no overlapping in the buffer. When $F2$ and $F3$ belonged to the same type, we cut the buffer with a small area with the one with a large area. (4) Divid Downriver with the buffer after processing. Downriver was divided into River 1 and River 2, as shown in Figure 5(d). (5) Respectively, merge River 1 and River 2 into the corresponding features and merge River 1 into $F2$ and River 2 into $F3$. The final processing results are shown in Figure 5(e). For land use integration, dimension-reduction treatment should be conducted for narrow surface features to compress the strip surface into the line feature with partial proportional scale. As for this example, the centerline extracted by strip feature could be used as its line feature, and this line feature did not run through the strip feature, so the topological location of the feature was expressed clearly.

3.3. Sideline Simplification Algorithm. Line feature simplification algorithms consist of some classic algorithms, such as the Douglas-Peucker algorithm [9, 20], progressive approach simplification algorithm [21], oblique dividing curve algorithm [22], and Li-Openshaw algorithm. The Douglas-Peucker algorithm was used in this paper. Commonly used in global line simplification, this algorithm not only maintains the shape characteristics of vector lines but also determines the simplification tolerance based on mapping requirements and effectively removes nodes that have small influence on the overall shape of features. Its principle is to first connect two line endpoints into a straight line, measure the vertical distance from each node between the two endpoints to the straight line, remove all nodes between the two endpoints if the maximum distance is within the specified tolerance limit, make two straight lines, respectively, from the node to the two endpoints if the distance from a certain node to the straight line is greater than the tolerance limit, and then, respectively, compare them, until the line cannot be divided (see Figure 6).

When conducting sideline simplification for land use data, we note that consistent simplification should be conducted for important lines of administrative boundaries, roads, and rivers, and independent simplification should be avoided because it will result in inconsistent administrative boundaries or changes in topological relationships between rivers, roads, and other surrounding features.

4. Discussion and Conclusions

Data from the second national land survey of Longtou Subdistrict of Dalian Lushun Port of Liaoning province was used in this study. We unified the land use type of the data into type division of Appendix B in the People's Republic of China land management industry standard TD/T 1027–2010 file (Figure 7(a)). The minimum patch area of research data is 400 m^2, and the scale is 1 : 10,000. According to the 1 : 10 land use data requirements of the 2006–2020 overall plan for land utilization, the minimum patch area of a map is 10,000 m^2, and 30 m is the furthest aggregation process distance. We used the previous algorithm to generalize the data the results of which are shown in Figure 7(b). The number of patches in the data decreased from 1007 to 428, and the compression

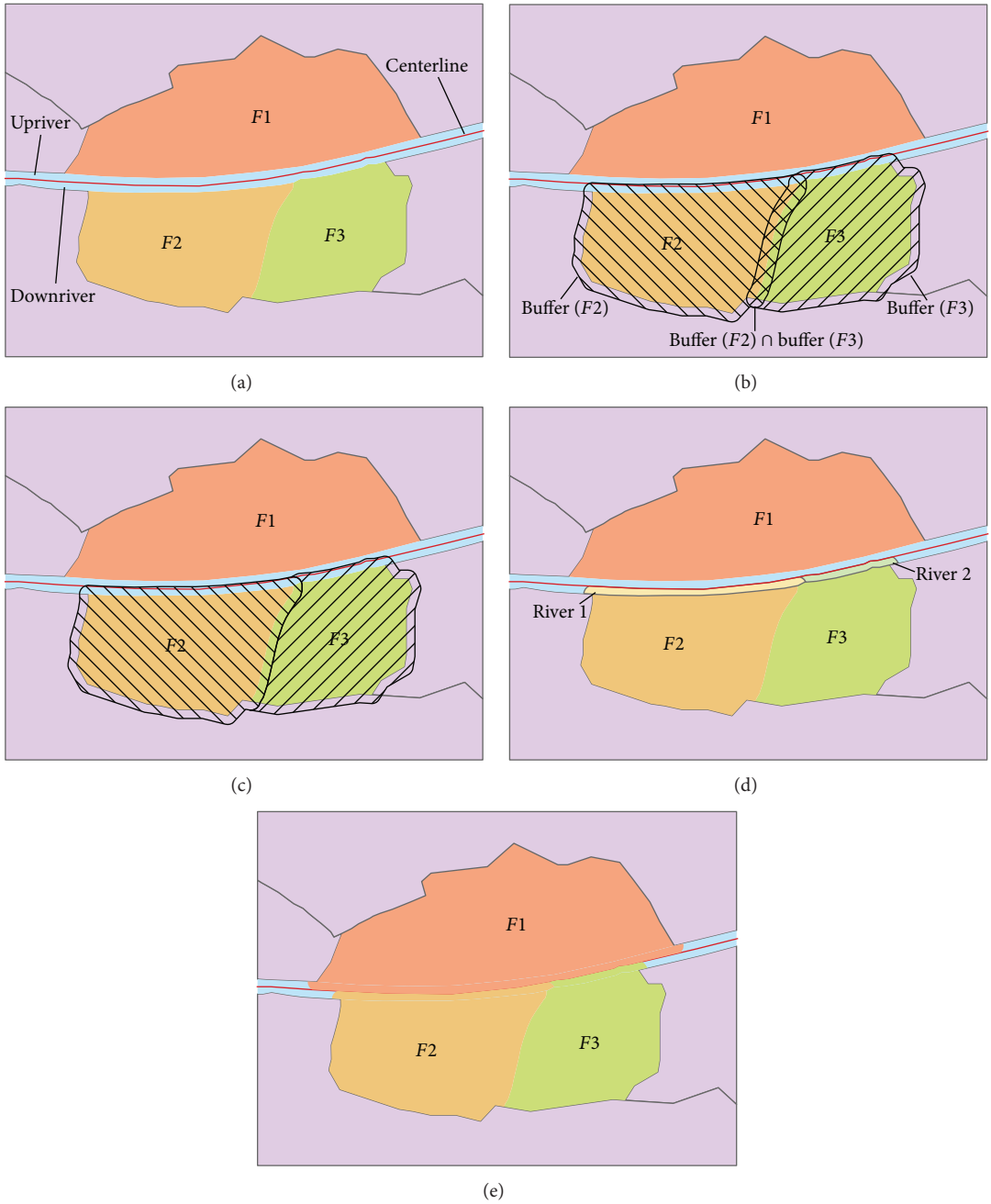


Figure 5: Process of handling strip feature.

ratio was 52.1%. The change rate of the total area of the important city and countryside construction land was 0.72%. The change rates of all land use types were less than 4%, except for agroland (19.36%) due to its scarcity and being highly dispersed. After generalization, some agro-land was integrated into other classes, and therefore changes in its area were larger than appropriate limits, which were considered as special circumstances. These generalization methods above must cause information loss as follows: the amalgamation of adjacent small area patches did not cause information loss; the aggregation of separate small patches with neighboring semantics caused area information loss, but its attribute and location information was preserved. Long and narrow terrain was simplified into lines, which maintained information and resulted in a very little loss. The most serious loss of information comes from the merging of isolated patches into other land types. In conclusion, methods based on semantic priority maintained the general characteristics of the original data, and thus the change in total area of each land type was very small. Microelements and the narrow area were

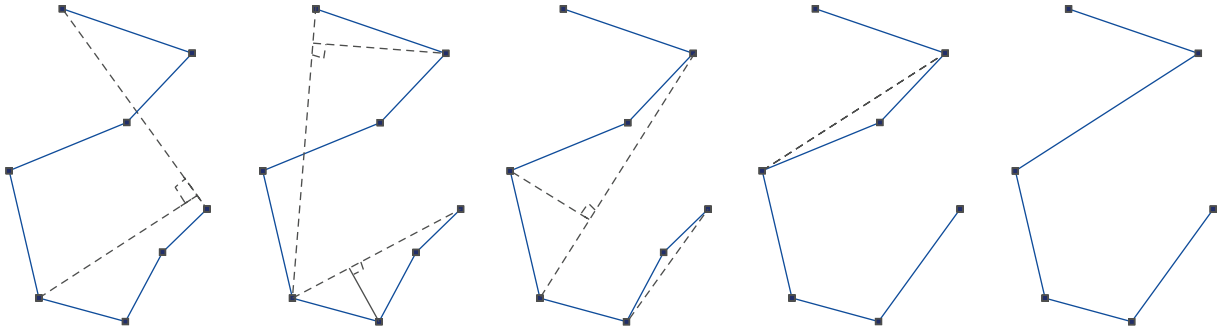
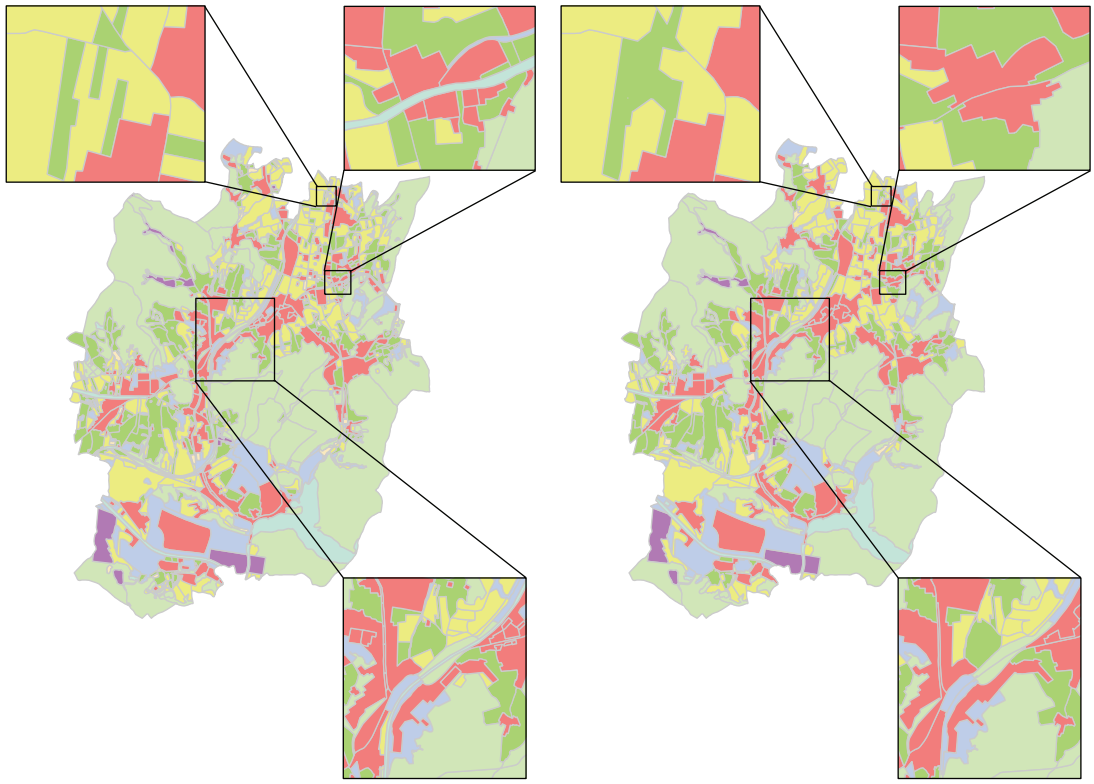


Figure 6: Douglas-Peucker algorithm procedure.



(a) Source data (b) Result data

Figure 7: Comparison of generalization.

managed effectively and reasonably. In addition, the buffer algorithm was simple and fast. However, because the division was not smooth when dividing narrow features using the buffer (see Figure 5(c)), there was a small raised area where the narrow feature absorbed by Feature $F3$ contacted $F2$, which will be the focus of future research.

Acknowledgments

The work described in this paper was substantially supported by the National Natural Science Foundation of China (nos. 40971299, 41171137) and Humanities Social Science Foundation of Ministry of Education (no. 09YJC790135).

References

[1] Y. Zongbo, "On mapping integration and compiling technique of image maps," *Scientia Geographica Sinica*, vol. 8, no. 1, pp. 87–93, 1988.

[2] R. Chithambaram, K. Beard, and R. Barrera, "Skeletonizing polygons for map generalization," *Technical Papers ACSM. Baltimore*, vol. 2, pp. 44–55, 1991.

[3] T. Ai and H. Wu, "Consistency correction of shared boundary between adjacent polygons," *Geomatics and Information Science of Wuhan University*, no. 5, pp. 426–442, 2000.

[4] T. H. Ai, R. Z. Guo, and X. D. Chen, "Simplification and aggregation of polygon object supported by delaunay triangulation structure," *Journal of Image and Graphics*, no. 7, pp. 93–99, 2001.

[5] L. Harrie, "Weight-setting and quality assessment in simultaneous graphic generalization," *The Cartographic Journal*, vol. 40, no. 3, pp. 221–233, 2003.

[6] L. Kulik, M. Duckham, and M. Egenhofer, "Ontology-driven map generalization," *Journal of Visual Languages & Computing*, vol. 16, no. 3, pp. 245–267, 2005.

[7] R. Zhao, J. Chen, D. Wang, Y. Shang, and T. Ai, "The design and implementation of geo-spatial database updating system based on digital map generalization," in *2nd International Conference on Space Information Technology*, Proceedings of SPIE, Huazhong University of Science and Technology; The Second Academy of China Aerospace Science and Industry Corporation; The National Natural Science Foundation of China; Chinese Academy of Space Technology; China Aerospace Science and Industry Corporation, Wuhan, China, November 2007.

[8] J. Li, D. Zhu, X. Song, Y. Chen, and Y. Yang, "A Polygon simplification algorithm with area-balance consideration," *Geography and Geo-Information Science*, no. 1, pp. 103–106, 2009.

[9] W. Huang, W. Dai, and S. Yu, "Using modified Douglas-Peucher algorithm based on area preservation to simplify polygons," *Science Technology and Engineering*, no. 24, pp. 7325–7328, 2009.

[10] J. Stoter, D. Burghardt, C. Duchêne et al., "Methodology for evaluating automated map generalization in commercial software," *Computers, Environment and Urban Systems*, vol. 33, no. 5, pp. 311–324, 2009.

[11] Q. Qiao and T. Zhang, "Automated map generalization in distributed environments," in *Proceedings of International Joint Conference on Computational Sciences and Optimization (CSO '09)*, pp. 181–183, IEEE Computer Society, Hainan, China, April 2009.

[12] A. Dilo, P. van Oosterom, and A. Hofman, "Constrained tGAP for generalization between scales: the case of Dutch topographic data," *Computers, Environment and Urban Systems*, vol. 33, no. 5, pp. 388–402, 2009.

[13] L. V. Stanislawski, "Feature pruning by upstream drainage area to support automated generalization of the United States National Hydrography Dataset," *Computers, Environment and Urban Systems*, vol. 33, no. 5, pp. 325–333, 2009.

[14] T. Foerster, L. Lehto, T. Sarjakoski, L. T. Sarjakoski, and J. Stoter, "Map generalization and schema transformation of geospatial data combined in a Web Service context," *Computers, Environment and Urban Systems*, vol. 34, no. 1, pp. 79–88, 2010.

[15] T. Ai, F. Yang, and J. Li, "Land-use data generalization for the database construction of the second land resource survey," *Geomatics and Information Science of Wuhan University*, vol. 35, no. 8, pp. 887–891, 2010.

[16] Y. Liu, H. Li, and C. Yang, "Ontology based land use data generalization," *Geomatics and Information Science of Wuhan University*, vol. 35, no. 8, pp. 883–886, 2010.

[17] Y. Zhu, S. Zhou, and T. Lu, "Rearch on spatial data line generalization algorithm in map generalization," *Journal of Software*, vol. 6, no. 2, pp. 241–248, 2011.

[18] T. Ai and Y. Liu, "Aggregation and amalgamation in land-use data generalization," *Geomatics and Information Science of Wuhan University*, vol. 27, no. 5, p. 486, 2002.

[19] J. Weng, Q. Guo, X. Wang, and P. Liu, "An improved algorithm for combination of land-use data," *Geomatics and Information Science of Wunan University*, no. 9, pp. 1116–1118, 2012.

[20] X. Liu, S. Li, and W. Huang, "Study of Douglas-Peucker algorithm controlling by the goniometry in generalization," *Geomatics & Spatial Information Technology*, vol. 29, no. 1, pp. 59–60, 2006.

[21] Q. Guo, "Study on progressive approach to graphic generalization of linear feature," *Geomatics and Information Science of Wuhan University*, no. 1, pp. 54–58, 1998.

[22] H. Z. Qian, F. Wu, B. Chen, J. H. Zhang, and J. Y. Wang, "Simplifying line with oblique dividing curve method," *Acta Geodaetica et Cartographica Sinica*, vol. 36, no. 4, pp. 443–456, 2007.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 975359, 7 pages
http://dx.doi.org/10.1155/2013/975359

Research Article

Spatiotemporal Simulation of Tourist Town Growth Based on the Cellular Automata Model: The Case of Sanpo Town in Hebei Province

Jun Yang,[1,2] Peng Xie,[1] Jianchao Xi,[1] Quansheng Ge,[1] Xueming Li,[2] and Fanqiang Kong[2]

[1] *Institute of Geographic Sciences and Natural Resources Research, CAS, Beijing 100101, China*
[2] *Liaoning Key Laboratory of Physical Geography and Geomatics, Dalian 116029, China*

Correspondence should be addressed to Peng Xie; xiepenggis@163.com

Received 31 January 2013; Accepted 26 March 2013

Academic Editor: Jianhong (Cecilia) Xia

Spatiotemporal simulation of tourist town growth is important for research on land use/cover change under the influence of urbanization. Many scholars have shown great interest in the unique pattern of driving urban development with tourism development. Based on the cellular automata (CA) model, we simulated and predicted the spatiotemporal growth of Sanpo town in Hebei Province, using the tourism urbanization growth model. Results showed that (1) average annual growth rate of the entire region was 1.5 Ha^2 per year from 2005 to 2010, 4 Ha^2 per year from 2010 to 2015, and 2.5 Ha^2 per year from 2015 to 2020; (2) urban growth rate increased yearly, with regional differences, and had a high degree of correlation with the Euclidean distance of town center, traffic route, attractions, and other factors; (3) Gougezhuang, an important village center in the west of the town, demonstrated traffic advantages and increased growth rate since 2010; (4) Magezhuang village has the largest population in the region, so economic advantages have driven the development of rural urbanization. It showed that CA had high reliability in simulating the spatiotemporal evolution of tourist town, which assists the study of spatiotemporal growth under urbanization and rational protection of tourism resources.

1. Introduction

Tourist towns have become increasingly important in tourism development and rural urbanization in recent years. Many researchers have focused on the practical use of tourism resources to achieve sustainable development of tourist towns [1, 2]. Such studies have conducted qualitative analysis on the role of various factors in sustainable tourist town development, so it is difficult to objectively predict the future development of these towns. Therefore, it is important to effectively simulate the development trend of tourist towns and suggest reasonable development plans.

The cellular automata (CA) model was first introduced by John von Neumann in the 1940s under the inspiration of mathematician and physicist Stanislaw Ulam during the "Manhattan Project." The cellular automata (CA) model is a mathematical framework discrete in time, space, and state and is a dynamic evolution system consisting of a large number of interacting cells. Wolfram [3] played a significant role in promoting early CA studies and laid the theoretical basis for such work. Because the CA model has strong spatial computing power, a complex and global pattern can be formed through simple local operations. This model has been successfully applied to environmental change, landscape pattern replacement, forest fire diffusion, urban expansion, and other simulation studies [4–11].

In urban geography, CA has been applied in the simulation of large-scale urban changes to explore the application of discrete dynamic models in urban land use change [12, 13] and in the simulation of urban systems [12, 14, 15]. Zhou and Chen [16] studied CA and elaborated on its principles, potential problems, and research significance, while Li and Yeh [17] used binding CA in the simulation of sustainable urban development patterns. The above studies have played a large role in promoting the development and application of CA theory in China.

In recent years, new developments in the urban CA model have been driven by progress in computer technology. Lauf et al. [18] studied an improved CA model, which included family and housing factors as the driving force through integrating system dynamics, to explore residential construction land expansion. García et al. [19] analyzed urban expansion in Northern Spain and Galicia by a variety of methods. Mitsova et al. [20] studied urban expansion and the protection of sensitive areas using CA based on land-use changes. Furthermore, a research has been conducted on the acquisition of CA conversion through combining neural network theory, data mining data and genetic algorithm theory, and on the enhanced simulation accuracy of the model [21–23].

The present paper applied the principle of CA to the simulation of tourist town urbanization, selected both suitable and limiting elements based on tourist town requirements for the development and protection of tourism resources, expanded the parameter system of traditional CA, and established the tourist town CA model to simulate Sanpo development from 2010 to 2020.

2. Model and Methods

2.1. Study Area. Sanpo town is located in the Taihang Mountain area, 28 km northwest of the Laishui County town of Baoding in Hebei Province, China. Sanpo is a national natural scenic area with significant regional advantages and is positioned 90 km east of Beijing, 170 km southeast of Tianjin and Langfang, and 90 and 190 km south of Baoding and Shijiazhuang, respectively. The town covers a total area of 200 km^2, is 180 to 1500 m above sea level, and has a total population of 11,887 people. The Beijing-Yuanping Railway, Baoye Highway, and 108 Highway run through the town. Sanpo is the administrative heart of the Yesanpo National Scenic Area, covering most of the scenic area and having many residents. For management purposes, Sanpo consists of a number of smaller villages, such as Gougezhuang and Magezhuang (Figure 1). With the development of tourism in Yesanpo since 1986, Sanpo has gradually evolved into a specialized tourist town. The past 25 years of development provide clear temporal and spatial evidence on the evolution of tourism development in this representative rural tourism town in China.

2.2. Tourism CA Model Framework. Tourism urbanization is a complex process. To simulate the spatio-temporal evolution of a tourist town, the driving factors and comprehensive mechanisms of tourism urbanization evolution must be studied. We first analyzed the type of tourism in the study area, clarified the development of the main tourism industry chains, and collected land use data, topographic maps, remote sensing data, and economic and social statistics of Sanpo over the years. We further investigated the integration of data compilation and multisource data through the area, studied the spatio-temporal evolution and relationship between driving factors using GIS spatial analysis and statistic functions, and combined analytic hierarchy processes (AHP). On this basis, we conducted spatial growth simulation through Python language programming using CA and urbanization growth models with ArcGIS software, as shown in Figure 2.

2.3. Model Structure and Index System. The model was established based on raster data and using the GRID raster data coding format of ArcGIS. The cellular space of the model covered the entire study area with 10 m × 10 m grids. To achieve multivariate data sharing, the model system used the WGS84 coordinate system. Cellular state space was divided into urban land, land that can be used for urbanization, and land that cannot be used for urbanization. Urban land information was provided by the land department of the Sanpo government, and land that cannot be used for urbanization was determined according to the requirements of protected areas and the constraining conditions of the terrain.

As external environment information of the cellular model, driving factor group data directly influences and controls the evolution of tourism urbanization. Tourism urbanization is a complex process with many influencing factors. By analyzing the comprehensive mechanisms of tourism and urbanization, we determined three major tourism urbanization factor groups and determined the weight coefficients of the driving factors by AHP (Table 1). The processing results of spatial data are shown in Figure 3. To achieve sustainable development and effectively protect tourism resources, natural reserves were taken as a spatial limiting factor, and the cell affected by spatial limiting factors could not be developed into a town.

2.4. Neighborhood Structure and Conversion Rules. The study area was a mountainous region with complex terrain. We considered two kinds of urbanization evolution power, specifically the influence of terrain and the influence of tourist factors. Firstly, the terrain on both sides of the valley watercourse was relatively flat with traffic arteries, so it was considered suitable for urban development. Secondly, the area was close to tourist attractions but beyond the safety distance of attraction protection to develop into a town.

2.4.1. Neighborhood Urbanization Function. The degree of urbanization was represented within the current cellular neighborhood, expressed by the proportion of urbanized cells within the scope of neighborhood, as follows:

$$P_{(x,y)} = \frac{1}{N} \sum_{i=1,j=1}^{\Omega} X_{(i,j)}, \tag{1}$$

where $P_{(x,y)}$ represents the percentage of urbanized cells within the neighborhood space, N represents the number of cells within the neighborhood space, $X_{(i,j)}$ represents urbanization cells, and Ω represents neighborhood space.

2.4.2. Cellular Conversion Probability. We determined the integrated value of the three major influencing factor groups

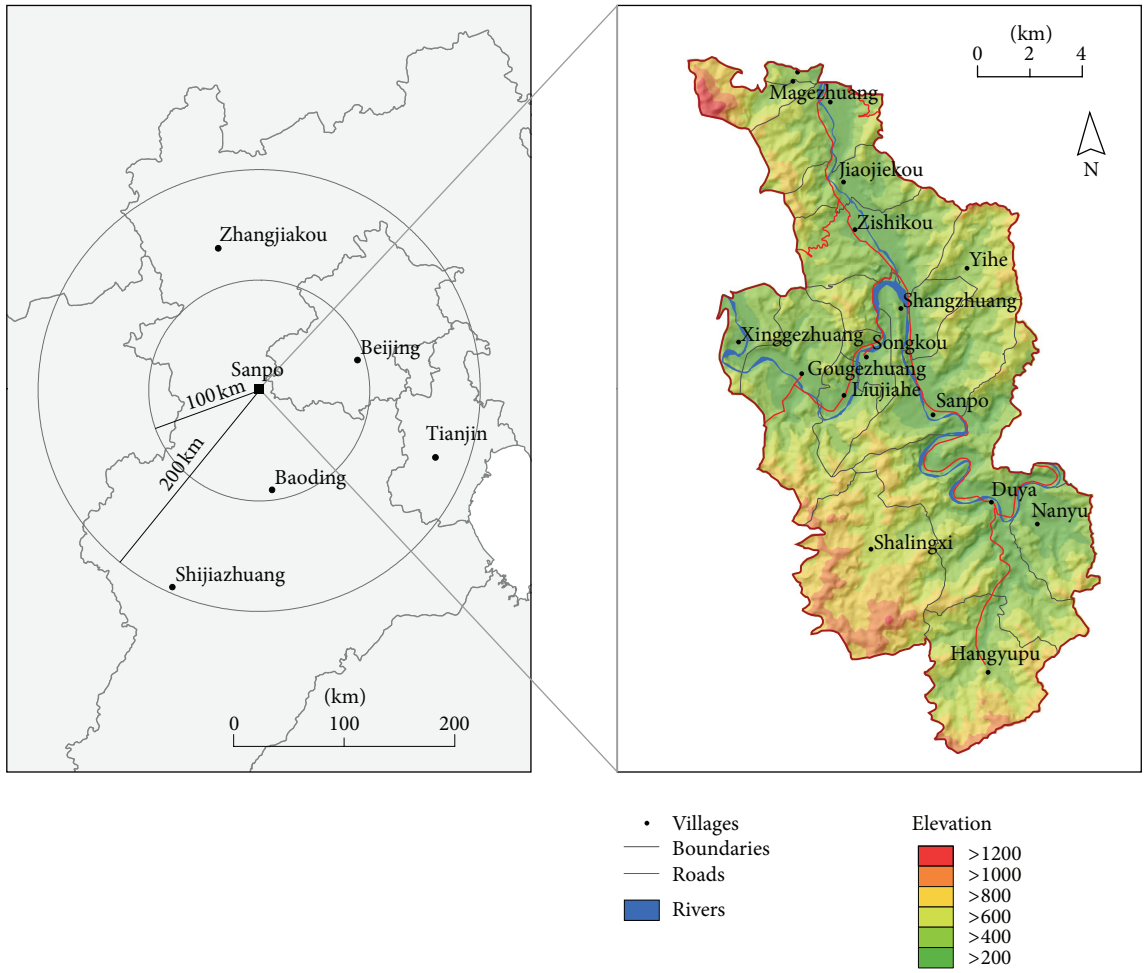


Figure 1: Location of study area.

and neighborhood urbanization function, representing the current cost value of cellular conversion, as follows:

$$p_{ij}^{t} = \phi\left(r_{ij}^{t}\right) = \exp\left[\alpha\left(\frac{r_{ij}^{t}}{r_{\max}} - 1\right)\right], \tag{2}$$

where α represents the diffusion coefficient and $r_{\max}$ represents the highest property value. The simple expression of r_{ij}^{t} is

$$r_{ij}^{t} = \left(\sum_{k=1}^{m} F_{ijk}^{t} W_k\right) \prod_{k=m+1}^{n} F_{ijk}^{t}, \tag{3}$$

when $1 \le k \le m$, r_{ij}^{t} represents the tourism urbanization driving factor, that is, the factors listed in Table 1; when $m < k$, r_{ij}^{t} represents the spatial limiting factor, referring to natural reserves and rivers, the probability for it to develop into a town is 0.

The simulation of CA was completed by multiple cycles. To express the uncertainty of tourism urbanization, p_{ij}^{t} (probability of developing into urban land) and $p_{\text{threshold}}$ (pregiven threshold) were added in the cycle for comparison to determine whether the current cell can develop into a town, that is,

$$\begin{cases} P_{ij}^{t} \ge P_{\text{threshold}} & \text{converted into urban land} \\ P_{ij}^{t} < P_{\text{threshold}} & \text{not converted into urban land.} \end{cases} \tag{4}$$

3. Results and Analysis

Based on the above models and methods, we conducted dynamic simulation of cellular automata for the spatiotemporal growth of Sanpo. Figure 4(a) represents the actual construction land in 2005 and 2010, and Figure 4(b) represents the simulated construction land in 2010, 2015, and 2020.

Because the study area was located in remote mountains on the north and south sides of the middle reaches of a river, the simulation results showed that the overall urban scale growth trend still did not develop in the flat area around the town. Within the scope of the study period, the towns and villages did not develop into an urban landscape and remained independent and decentralized geographical units.

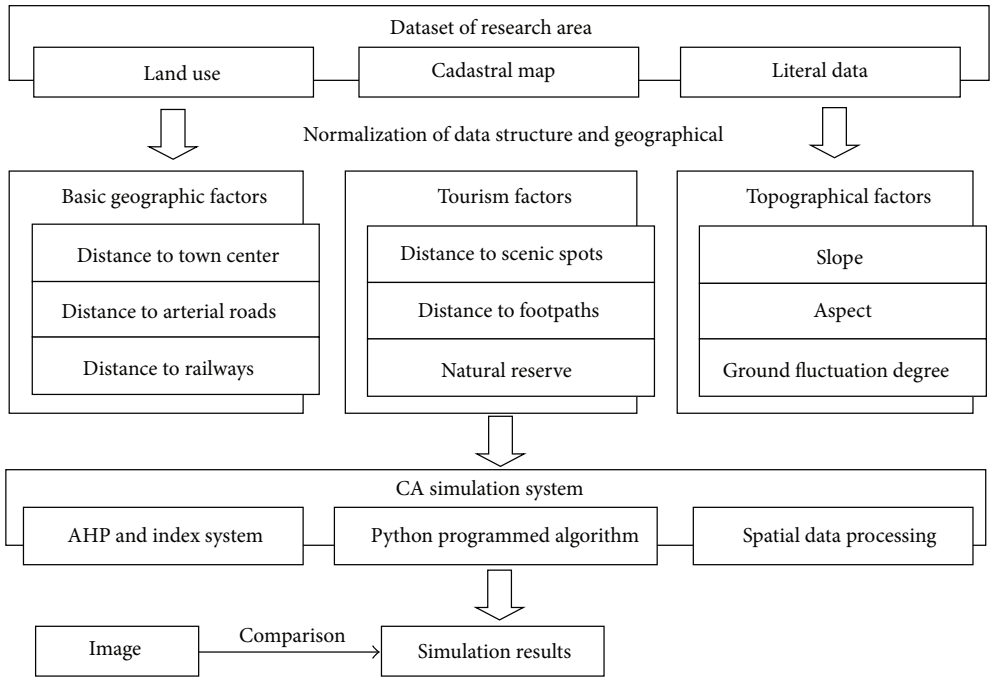


Figure 2: Frame diagram of spatio-temporal growth simulation of tourist town based on CA.

Table 1: Driving factors and weights of spatio-temporal growth of tourist town.

Influencing factor	Variable type	Weight coefficient	Variable grading
Basic geographic factors	Distance to town center	4	Grade 1: 0~500 m Grade 2: 501~1000 m Grade 3: 1001~3000 m Grade 4: >3000 m
	Distance to arterial roads	4	Grade 1: 0~100 m Grade 2: 101~500 m Grade 3: 501~1000 m Grade 4: >1000 m
	Distance to railway	2	Grade 1: 0~100 m Grade 2: 101~500 m Grade 3: 501~1000 m Grade 4: >1000 m
Geographic factors for tourism	Distance to footpath	3	Grade 1: 0~100 m Grade 2: 101~300 m Grade 3: 301~600 m Grade 4: >600 m
	Distance to scenic spots	3	Grade 1: 0~500 m Grade 2: 501~1000 m Grade 3: 1001~2000 m Grade 4: >2000 m
Topographical factors	Slope	4	Grade 1: <5 degrees Grade 2: 6 degrees~15 degrees Grade 3: 16 degrees~25 degrees Grade 4: >25 degrees
	Aspect	1	Grade 1: −45 degrees~45 degrees Grade 2: 45 degrees~135 degrees Grade 3: 135 degrees~225 degrees Grade 4: 225 degrees~315 degrees
	Ground fluctuation degree	2	Grade 1: <10 Grade 2: 10~20 Grade 3: 20~40 Grade 4: >40

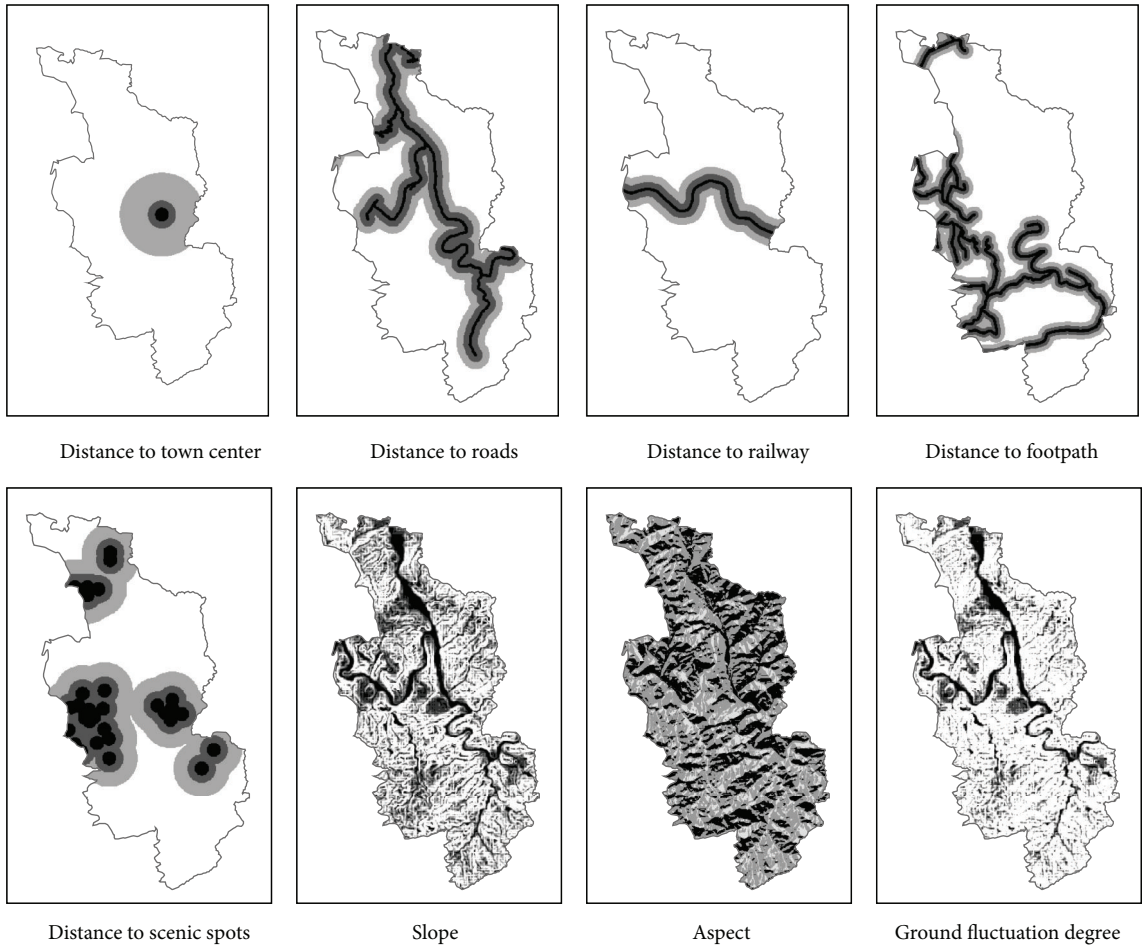


Figure 3: Spatio-temporal growth driving factors of tourist town.

We further analyzed the characteristics of urban growth and its driving mechanism in the Sanpo area regarding changes in time and space patterns.

Differences in urban development speed in different time periods were observed. Urban land increased by 7.5 Ha^2 from 2005 to 2010, with an average increase of 1.5 Ha^2 per year, and was expected to increase by 20 Ha^2 from 2010 to 2015, average increase rate of 4 Ha^2 per year, and by 17.520 Ha^2 from 2015 to 2020, average increase rate of 2.5 Ha^2 per year. These results were consistent with government development planning reports. Urban growth was mainly concentrated in the area near the town center, expanding at a rate of 1 Ha^2 per year, with the urban growth rate of other towns in the region being slow. This may be because the town center of Sanpo, which relies on the Yesanpo National Scenic Area, garnered significant support from development policies and funding in the early period of urban development. Furthermore, because urbanization adapts to the needs of local tourism development, tourism development accelerates urbanization in turn. Town urbanization accelerated after 2010 and was still the highest in the region with a growth rate of 1.3 Ha^2 per year. To the west of the town, Gougezhuang village developed rapidly, with an increase in construction land of 1.1 Ha^2 per year. Due to further demand for economic and tourism development in the Sanpo town center, contradictions existed in relation to sustainable development in the scenic area and land use restrictions, and other villages and towns (such as Gougezhuang and Magezhuang, Figure 4) showed tourism development potential with the support of resources and funding during this period. The simulated increase rate of urban land fell from 2015 to 2020, showing major growth near the town center because the growth area was topographically limited.

From the perspective of spatial patterns, urban growth regions in Sanpo were divided into three parts (right, Figure 4). The first was the northern area composed of South and North Chanfangzhuang and Magezhuang; the second was the western area, including Gougezhuang; and the third was Sanpo town center. Of these, the original urban town center covered the largest area and its growth rate was the fastest within the simulated period, demonstrating that the town center not only serves as a political, economical, and cultural center, but also as a tourist service base for the Yesanpo Scenic Area. The town center has developed at a fast rate due to the influx of people from surrounding towns and

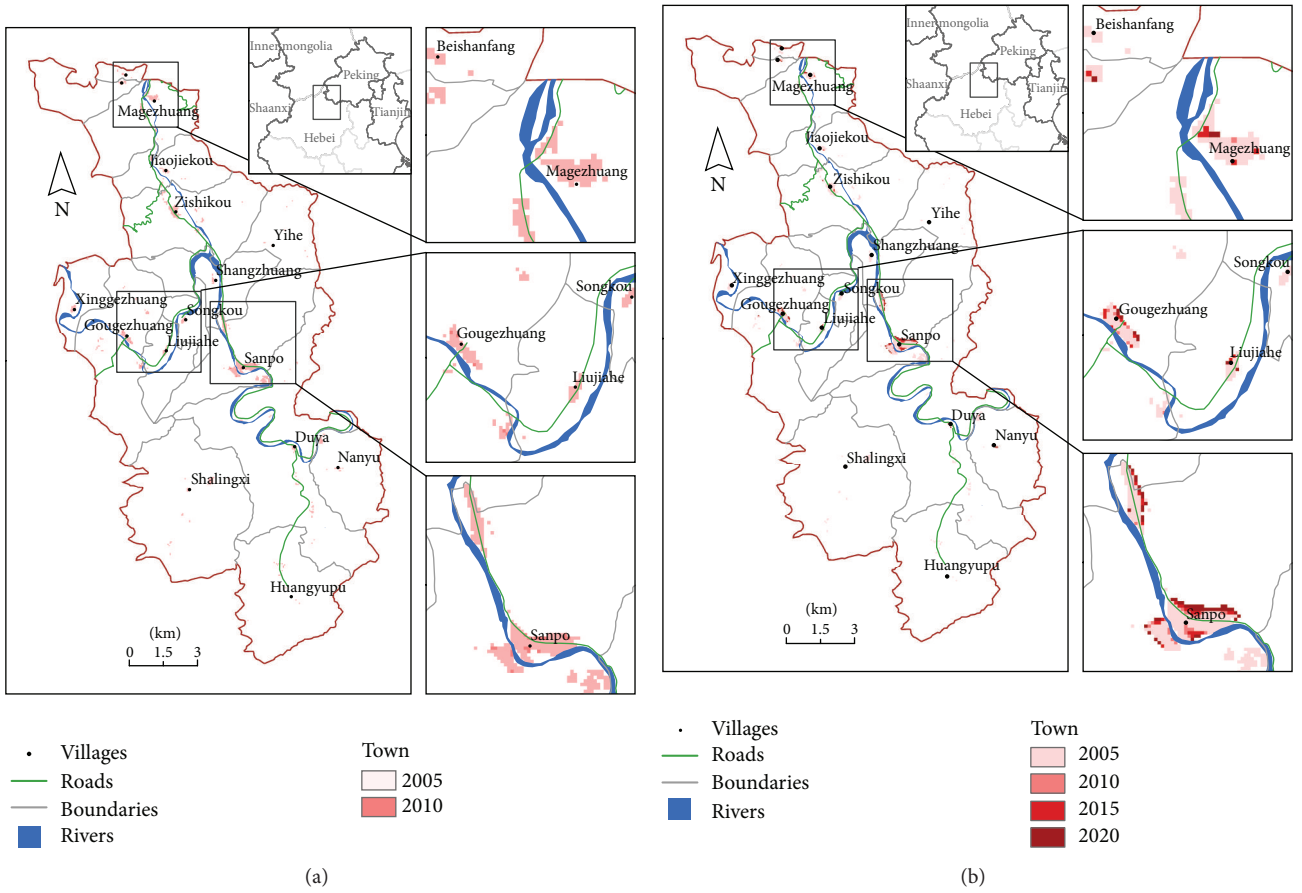


Figure 4: Spatio-temporal growth simulation results of Sanpo.

villages for employment, as well as the accommodation of temporary residents and visitors. Gougezhuang is an important central village to the west of the town, with an established Beijing-Yuanping Railway train station. Furthermore, it plays a role in radiating and driving the surrounding grass-roots villages, so there are unique advantages for tourism trade and service development in this village. In addition, with the largest population in this region, Magezhuang town growth will be inevitably achieved under the driving effect of the economy. For urban growth direction, most areas presented edge expansion; moreover, the urban growth rate of areas at the edge of town with traffic lines was faster than that of others, and the closer the area was to the tourist attraction, the faster its rate of urbanization was. The expansion of urban areas showed an obvious trend towards traffic and the tourism industry.

4. Conclusions and Discussion

This paper combined CA principles and GIS technology, established a tourism-type urban expansion model and simulated urban growth under the driving conditions of tourism factors using land use data of the Sanpo area from 2005 and 2010. The following was concluded.

(1) We analyzed the spatio-temporal growth of Sanpo using GIS technology and CA system principles, which reflected the role of tourism factors in regional urbanization and intuitively predicted the future trend of development of the town, with high reliability in solving spatio-temporal growth of the tourist city.

(2) By comprehensive analysis of the data and model, the Sanpo area was influenced by the natural environment, population growth, economic development, national policies, and other complex factors. The urban growth rate increased yearly, although with regional differences, and had a high degree of correlation with the Euclidean distance of town center, traffic route, attractions, and other factors.

(3) Because the Sanpo terrain was complex and tourism was a leading industry, the growth rates of towns surrounding the tourist attractions were relatively slow in the process of urban growth, achieving rational development, and utilization of tourism resources.

Because urbanization is a complex geographical process influenced by society, culture, and the economy, as well as by national economic policies and other factors, CA is still undergoing research and development and possesses a number of deficiencies [24]. Therefore, it is difficult to accurately simulate and predict urban growth. Some preliminary results have been made in researching the CA based model

and simulating the growth process of tourist towns under microscale conditions; however, due to basic data limitations, the simulation accuracy of this study needs to be enhanced and the tourism urbanization CA simulation system requires further improvement.

Acknowledgments

The work described in this paper was substantially supported by the National Natural Science Foundation of China (nos. 40971299, 41171137) and Humanities Social Science Foundation of Ministry of Education (no. 09YJC790135).

References

[1] L. Lu and J. Ge, "Reflection on the research progress of tourism urbanization," *Geographical Research*, vol. 25, no. 4, pp. 741–750, 2006.

[2] Z. Wang and H. Yu, "Study on coupling development between development of tourism industry and small town construction in Zhangjiajie City," *Economic Geography*, no. 7, pp. 165–171, 2012.

[3] S. Wolfram, "Cellular automata as models of complexity," *Nature*, vol. 311, no. 5985, pp. 419–424, 1984.

[4] M. H. Afshar, M. Shahidi, M. Rohani, and M. Sargolzaei, "Application of cellular automata to sewer network optimization problems," *Scientia Iranica*, vol. 18, no. 3, pp. 304–312, 2011.

[5] Y. Feng and Z. Han, "Impact of neighbor configurations on spatially-explicit modeling results," *Geographical Research*, vol. 30, no. 6, pp. 1055–1065, 2011.

[6] S. Kokubo, J. Tanimoto, and A. Hagishima, "A new cellular automata model including a decelerating damping effect to reproduce Kerner's three-phase theory," *Physica A*, vol. 390, no. 4, pp. 561–568, 2011.

[7] E. A. Silva and K. C. Clarke, "Calibration of the SLEUTH urban growth model for Lisbon and Porto, Portugal," *Computers, Environment and Urban Systems*, vol. 26, no. 6, pp. 525–552, 2002.

[8] V. Spicer, A. A. Reid, J. Ginther, H. Seifi, and V. Dabbaghian, "Bars on blocks: a cellular automata model of crime and liquor licensed establishment density," *Computers, Environment and Urban Systems*, vol. 36, no. 5, pp. 412–422, 2012.

[9] F. Wu, "Calibration of stochastic cellular automata: the application to rural-urban land conversions," *International Journal of Geographical Information Science*, vol. 16, no. 8, pp. 795–818, 2002.

[10] H. Yu, Z. He, and X. Pan, "Wetlands shrink simulation using cellular automata: a case study in Sanjiang Plain, China," *Procedia Environmental Sciences*, vol. 2, pp. 225–233, 2010.

[11] H. Zhang, Y. Zeng, X. Jin, C. Yin, and B. Zou, "Urban land expansion model based on multi-agent system and application," *Acta Geographica Sinica*, vol. 63, no. 8, pp. 869–881, 2008.

[12] M. Batty, H. Couclelis, and M. Eichen, "Urban systems as cellular automata," *Environment and Planning B*, vol. 24, no. 2, pp. 159–164, 1997.

[13] H. Couclelis, "Cellular worlds: a framework for modeling micro-macro dynamics," *Environment and Planning A*, vol. 17, no. 5, pp. 585–596, 1985.

[14] M. Batty and Y. Xie, "From cells to cities," *Environment and Planning B*, vol. 21, supplement 21, pp. 531–548, 1994.

[15] M. Batty and Y. Xie, "Possible urban automata," *Environment and Planning B*, vol. 24, no. 2, pp. 175–192, 1997.

[16] Y. Zhou and Y. Chen, "Cellular automata and simulation of spatial complexity of urban systems: history, present situation and future," *Economic Geography*, vol. 20, no. 3, pp. 35–39, 2000.

[17] X. Li and A. G. O. Yeh, "Constrained cellular automata for modelling sustainable urban forms," *Acta Geographica Sinica*, vol. 54, no. 4, pp. 289–298, 1999.

[18] S. Lauf, D. Haase, P. Hostert, T. Lakes, and B. Kleinschmit, "Uncovering land-use dynamics driven by human decision-making: a combined model approach using cellular automata and system dynamics," *Environmental Modelling & Software*, vol. 27-28, pp. 71–82, 2012.

[19] A. M. García, I. Santé, M. Boullón, and R. Crecente, "A comparative analysis of cellular automata models for simulation of small urban areas in Galicia, NW Spain," *Computers, Environment and Urban Systems*, vol. 36, no. 4, pp. 291–301, 2012.

[20] D. Mitsova, W. Shuster, and X. Wang, "A cellular automata model of land cover change to integrate urban growth with open space conservation," *Landscape and Urban Planning*, vol. 99, no. 2, pp. 141–153, 2011.

[21] X. Li, "Emergence of bottom-up models as a tool for landscape simulation and planning," *Landscape and Urban Planning*, vol. 100, no. 4, pp. 393–395, 2011.

[22] X. Li, Q. Yang, and X. Liu, "Discovering and evaluating urban signatures for simulating compact development using cellular automata," *Landscape and Urban Planning*, vol. 86, no. 2, pp. 177–186, 2008.

[23] X. Liu, X. Li, X. Shi, S. Wu, and T. Liu, "Simulating complex urban development using kernel-based non-linear cellular automata," *Ecological Modelling*, vol. 211, no. 1-2, pp. 169–181, 2008.

[24] P. M. Torrens and D. O'Sullivan, "Cellular automata and urban simulation: where do we go from here?" *Environment and Planning B*, vol. 28, no. 2, pp. 163–168, 2001.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 508315, 7 pages
http://dx.doi.org/10.1155/2013/508315

Research Article

Algorithms and Applications in Grass Growth Monitoring

Jun Liu,[1,2] Xi Yang,[2] Hao Long Liu,[1] and Zhi Qiao[3]

[1] *Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences, A11 Datun Road, Chaoyang District, Beijing 100101, China*
[2] *College of Geography and Tourism, Chongqing Normal University, No. 12 Tianchen Road, Shapingba District, Chongqing 400047, China*
[3] *State Key Laboratory of Water Environment Simulation, School of Environment, Beijing Normal University, No. 19 Xinjiekouwai Street, Beijing 100875, China*

Correspondence should be addressed to Zhi Qiao; george@mail.bnu.edu.cn

Received 25 February 2013; Accepted 29 March 2013

Academic Editor: Craig Caulfield

Monitoring vegetation phonology using satellite data has been an area of growing research interest in recent decades. Validation is an essential issue in land surface phenology study at large scale. In this paper, double logistic function-fitting algorithm was used to retrieve phenophases for grassland in North China from a consistently processed Moderate Resolution Spectrodiometer (MODIS) dataset. Then, the accuracy of the satellite-based estimates was assessed using field phenology observations. Results show that the method is valid to identify vegetation phenology with good success. The phenophases derived from satellite and observed on ground are generally similar. Greenup onset dates identified by Normalized Difference Vegetation Index (NDVI) and in situ observed dates showed general agreement. There is an excellent agreement between the dates of maturity onset determined by MODIS and the field observations. The satellite-derived length of vegetation growing season is generally consistent with the surface observation.

1. Introduction

Vegetation phenology is the study of periodic plant life cycle events, such as bud burst, leaf out, flower bloom, and leaf fall, and how these are influenced by seasonal and interannual changes in environment [1]. Because small fluctuations of climate can make a big difference in the timing of the vegetation phenological events, plant phenology is widely accepted as a robust indicator of the response of terrestrial ecosystems to climate change [2, 3]. Variations in the timing of vegetation phenophases are key components to identify and evaluate the effects of climatic change on terrestrial ecosystems [4, 5]. Plant phonological monitoring has been an area of growing research interest in recent decades.

Plant phenophases have been monitored by field observations for thousands of years [6, 7]. Long-term records of species-level phenophases are useful in monitoring local climatic changes [8, 9]. However, field phenological observations are difficult to extrapolate to large spatial scale and working intensive.

Satellite-based remotely sensed data provides the potential to scale from species-level observations to regional shifts of phenological patterns [10]. More than 150 vegetation indices (VIs) have been produced from satellite data to describe the information of vegetation. Each index is designed to accentuate a particular vegetation property. For instance, Normalized Difference Vegetation Index (NDVI) derived from reflectance data collected by the Moderate Resolution Spectroradiometers (MODIS) on Terra and Aqua provides an indication of the canopy greenness of vegetation communities. NDVI has been proved to be valid for retrieving land surface phenology [5]. A generalized VI temporal profile is theoretically smooth and continuous. However, due to the influences of the noise in the satellite data including the cloud cover, atmospheric effects, bidirectional effects, and snow cover, the time-series VI data are always with remarkable fluctuations [11].

A variety of methods are used to reduce noise in NDVI data and to reconstruct high-quality time-series VI data, for

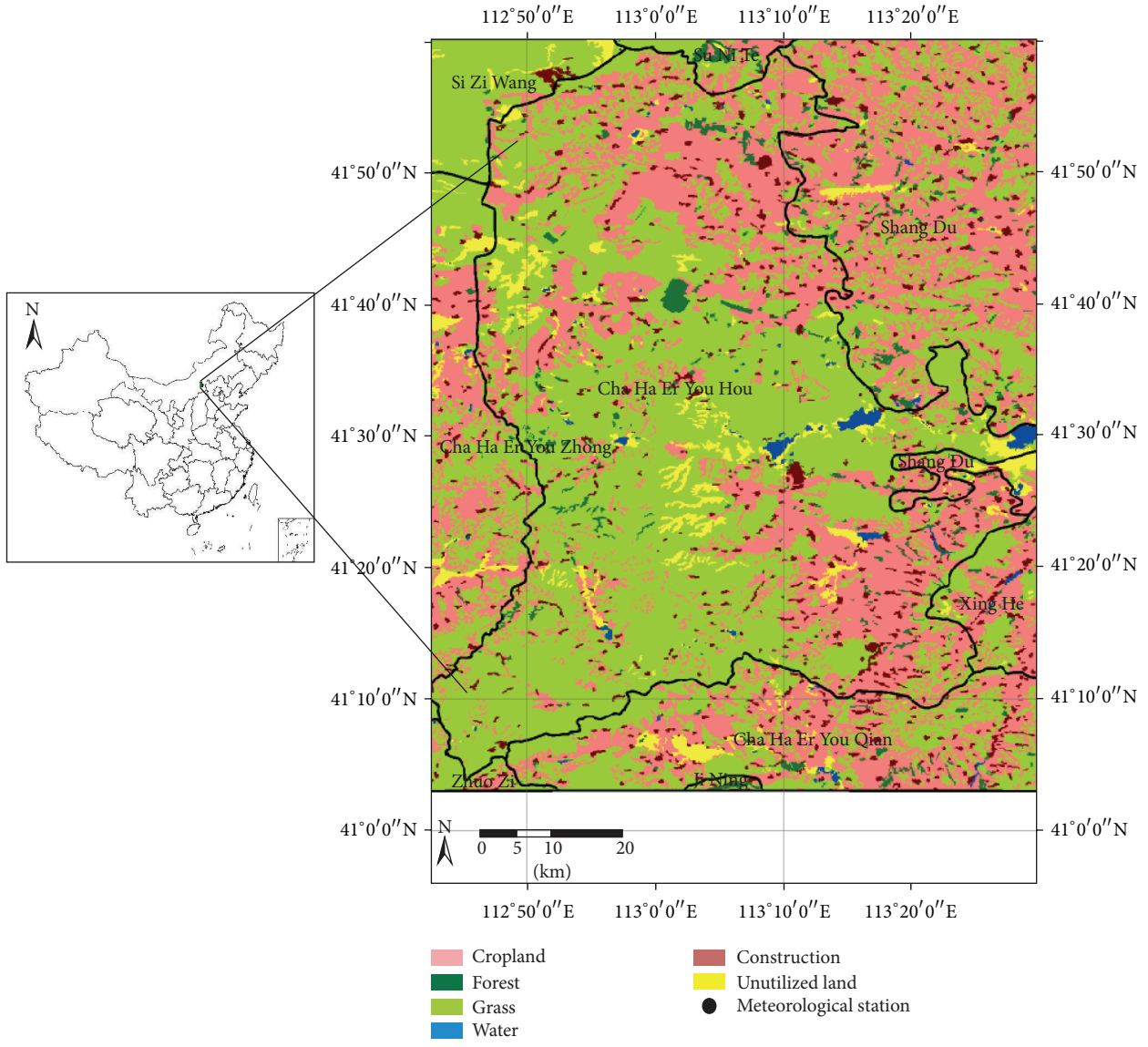


Figure 1: The location of study area.

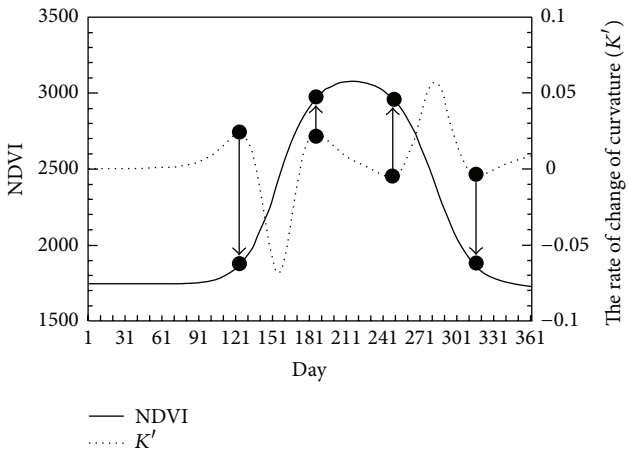


Figure 2: The reconstructed MOD13A2-NDVI time series and the rate of change of its curvature.

example, the asymmetric Gaussian function fitting, Savitzky-Golay filters, Fourier harmonic analysis, and piecewise logistic functions. After an empirically based comparison of different methods, Jennifer recommended the double logistic function-fitting algorithm due to robustness, scientific basis, and general applicability and some other desirable properties [12]. Based on the smoothed temporal VI data, Zhang et al. identified the phenological transition dates by the derivative of the curvature of the function [13].

However, it is often ambiguous what the satellite-retrieved phenological estimates actually track. For instance, the greatest temporal increase in the NDVI is due to snow melt rather than "start of the season" (SOS) [14, 15]. As a result, it is necessary to compare the satellite-measured land surface phenology with data observed at ground level. However, to date, measures of land surface phenology usually compare poorly with in situ observed phenology [16]. Therefore, researchers do not comprehensively understand how the

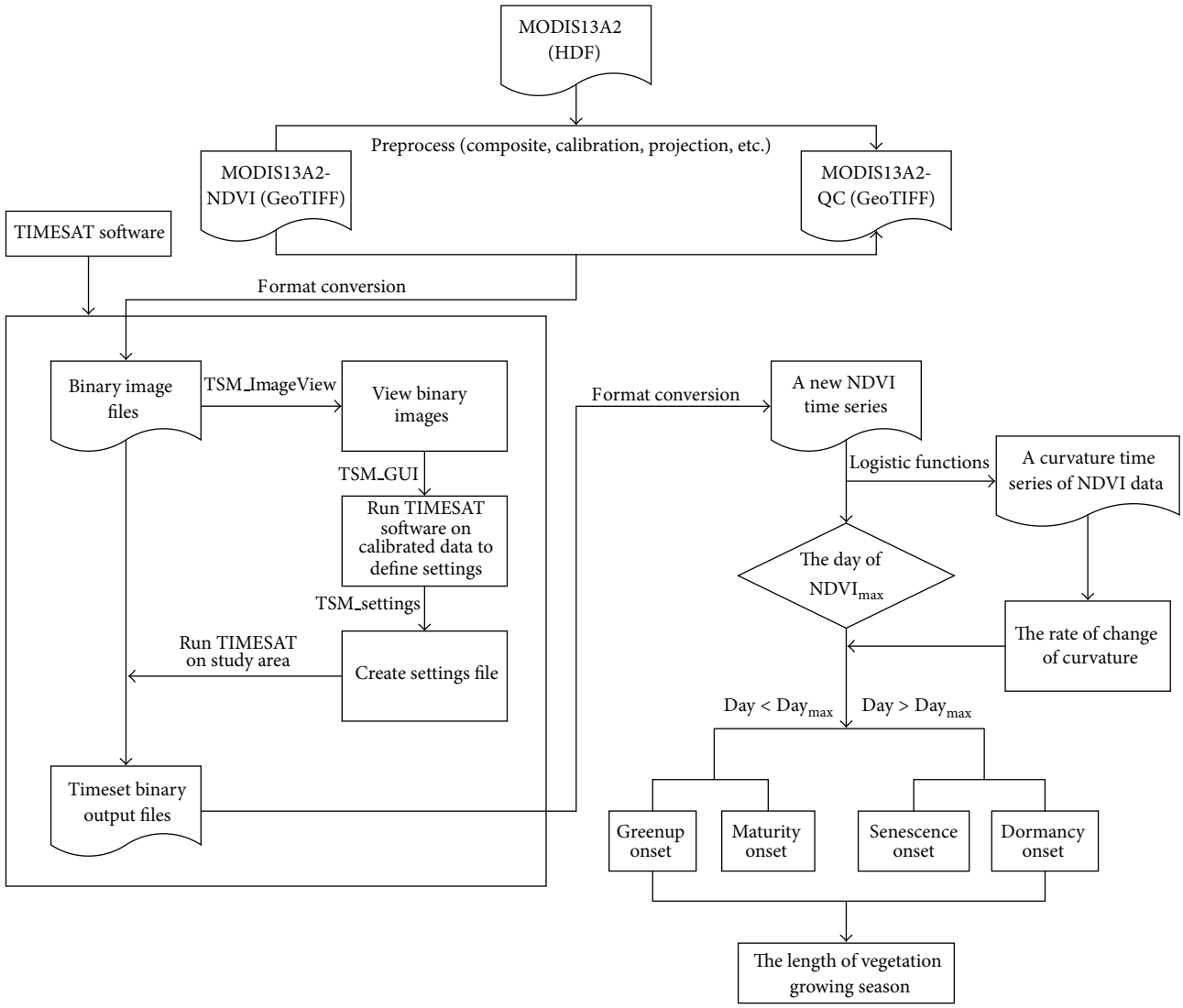


Figure 3: Flowchart of grassland growth monitoring process.

myriad definitions and methods are related to ground-based phenology. A primary hurdle is the incompatibility of spatial scales at which two types of observations are commonly obtained, in addition to different characteristic of the data. Such limitations have greatly impeded progress in the further application of time-series VI data.

Here, the double logistic function-fitting algorithm was used to retrieve phenophases for grassland in North China using a consistently processed MODIS dataset. Then, assessment of phenological dates was conducted using field observation phenology.

2. Study Area

The study was conducted on Leymus chinensis grassland in Inner Mongolia, one of the most representative grassland regions of China. Because the phenophases of Leymus chinensis are easily detectable from both field observations and remotely sensed data, it is well suited for phenological study based on remote data. Chahar Right Back Banner, the study area, is located in latitude 41°27′N and longitude 113°11′E. The area is characterized by a north temperate continental monsoon climate with average annual precipitation of 325.7 mm, average annual temperature of 4.17°C, and mean annual aridity index of 22.99. Grassland (mainly Leymus chinensis) is the dominant vegetation (Figure 1).

3. Material and Methodology

3.1. Material

3.1.1. MODIS Data. For this study, MOD13 A2 8-day NDVI and NDVI quality assurance products from Terra's Moderate Resolution Imaging Spectroradiometer (MODIS) were used. The dataset covered one-year time series, from January 1, 2007 to December 31, 2007 and included 46 images. The MODIS data have a spatial resolution of 1 km at nadir, which is well suited to monitor seasonal vegetation dynamics at the

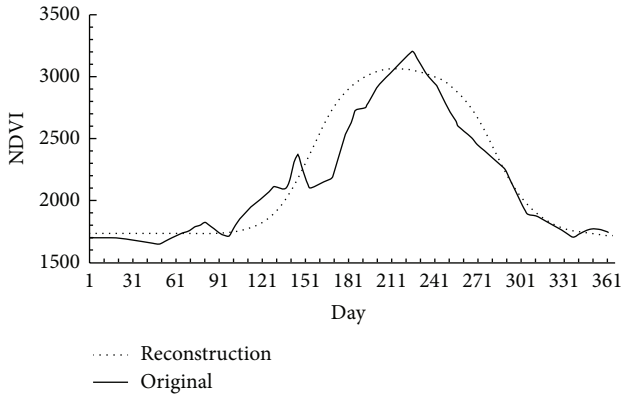


Figure 4: Comparison of original and reconstructed (double-logistic functions) MOD13A2-NDVI time series.

scale of the landscape. These data were acquired from the US Geological Survey.

The NDVI is calculated from these individual measurements as follows:

$$\text{NDVI} = \frac{(\text{NIR} - \text{VIS})}{(\text{NIR} + \text{VIS})}, \tag{1}$$

where VIS and NIR stand for the spectral reflectance measurements acquired in the visible (red) and near-infrared regions, respectively.

3.1.2. Field-Observed Data. Ground-observed grassland phenophases were obtained from Chen and Li [17]. The observations of the different phases of Leymus chinensis are carried at Chahar Right Back Banner stations of Inner Mongolia. The observation criterion of the grassland has been developed by China Meteorological Administration.

3.2. Methodology. The selected VI values were fitted with algorithm presented in Zhang et al. [13, 18] for reconstructing smoothing timeseries curves. This algorithm characterizes vegetation growth cycles using four transition dates derived from time series of MODIS VI data: (1) greenup, the date of onset of photosynthetic activity; (2) maturity, the date at which plant green leaf area is at maximum; (3) senescence, the date at which photosynthetic activity and green leaf area begin to rapidly decrease; (4) dormancy, the date at which physiological activity becomes near zero.

The annual change in satellite-derived VI data for a single growth or senescence cycle can be modeled using

$$y(t) = \frac{c}{1 + e^{a+bt}} + d, \tag{2}$$

where t is time in days, $y(t)$ is the NDVI value at time t, a and b are fitting parameters, $c + d$ is the maximum NDVI value, and d is the initial background NDVI value. The fitting parameters a and b were determined using least-square fitting.

The rate of the change in the curvature of the fitted logistic models is used to determine the four key transition dates (Figure 2).

A series of preprocessing steps were performed to smooth MODIS NDVI data products using TIMESAT software to identify the single growth and senescence cycle. The objective is to eliminate the abnormal value, that is, cloud and snow. Then, the VI data could be fit to logistic functions described by (2).

Zhang et al. [13] define the onset as the date when the second derivative gets from positive to negative values. The onset of four transition dates corresponds to the times at which the second derivative gets from positive to negative values. These transitions dates indicate when one phenological phase transitions from one approximately linear stage to another. The curvature K for (1) at time t can be computed by

$$K = \frac{d_a}{d_s} = -\frac{b^2 cz\,(1-z)\,(1+z)^3}{\left[(1+z)^4 + (bcz)^2\right]^{3/2}}, \tag{3}$$

where $z = e^{a+bt}$, a is the angle of the unit tangent vector at time t along a differentiable curve, and s is the unit length of the curve. The rate of change of curvature K' can be computed by

$$K' = b^3 cz \left\{ \frac{3z\,(1-z)\,(1+z)^3 \left[2(1+z)^3 + b^2 c^2 z\right]}{\left[(1+z)^4 + (bcz)^2\right]^{5/2}} - \frac{(1+z)^2 \left(1 + 2z - 5z^2\right)}{\left[(1+z)^4 + (bcz)^2\right]^{3/2}} \right\}. \tag{4}$$

The detailed flowchart of grassland growth monitoring process is illustrated in Figure 3. We first made a preprocess for the MOD13 A2 dataset taking MODIS Reprojection Tool (MRT) software as a platform. In order to eliminate the effect of cloud and snow, we smooth the dataset using TIMESAT software. For the corrected dataset, we simulated the phenophases for grassland through Arc Macro Language (AML) programming in accordance with the above algorithm.

4. Results

4.1. The Effect of Data Smoothing. The final smoothed curve was produced from the double logistic model. Figure 4 shows the effect of data smoothing on the time series for a grassland pixel. The reconstructed NDVI time series is smoother and the noise resulting from the atmospheric conditions is considered as outliers and removed. The new smooth and continuous time series fit to natural rules of grassland growth.

4.2. Phenophases of Grassland. The phenological pattern of grassland of Inner Mogolia during a single growth cycle is realistically identified using the method described in Section 3. Figures 5(a) to 5(e) present ecologically and geographically coherent patterns that are consistent with known phenological behavior in this area.

Greenup onset begins at the early April (124 day-of-year) in the station. Note that south areas exhibit earlier greenup.

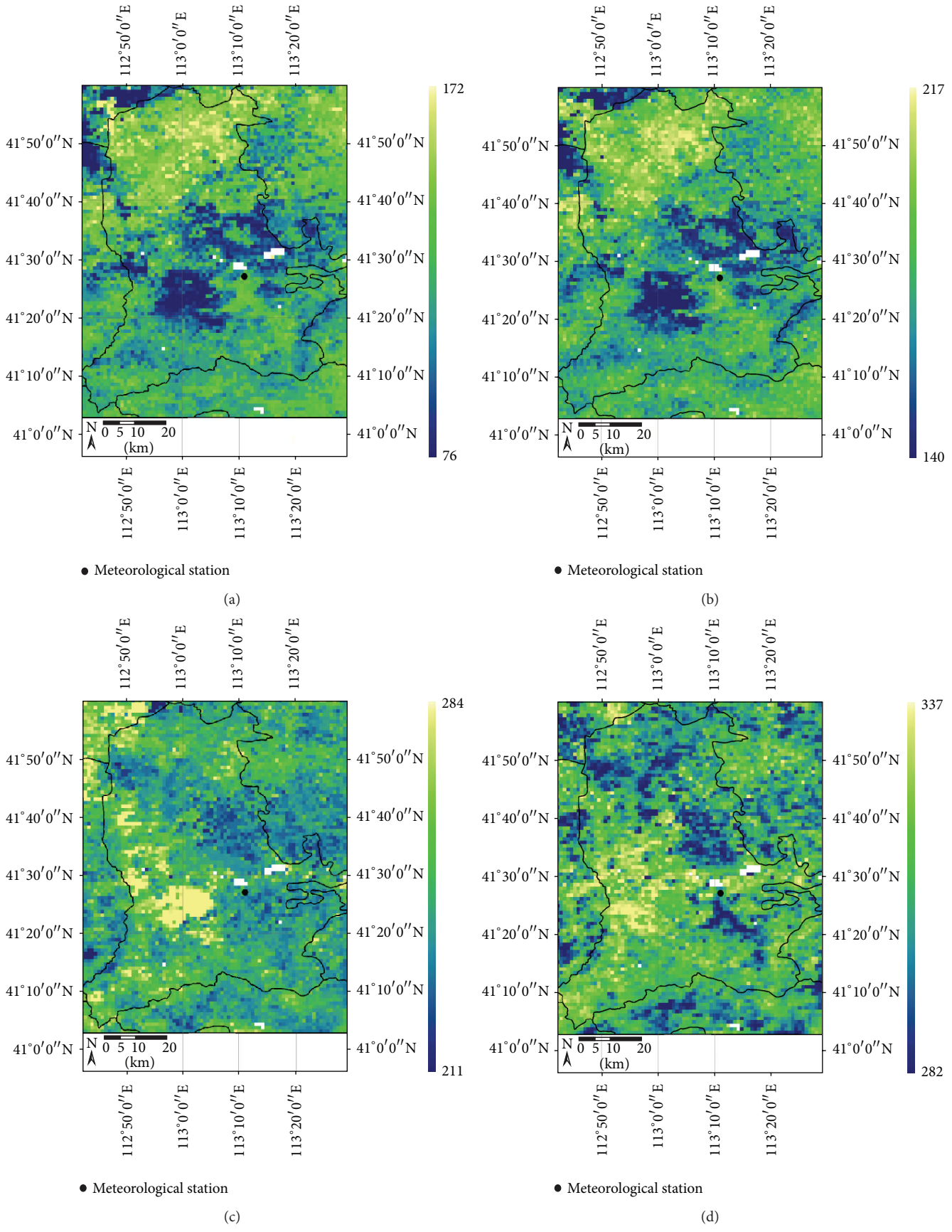
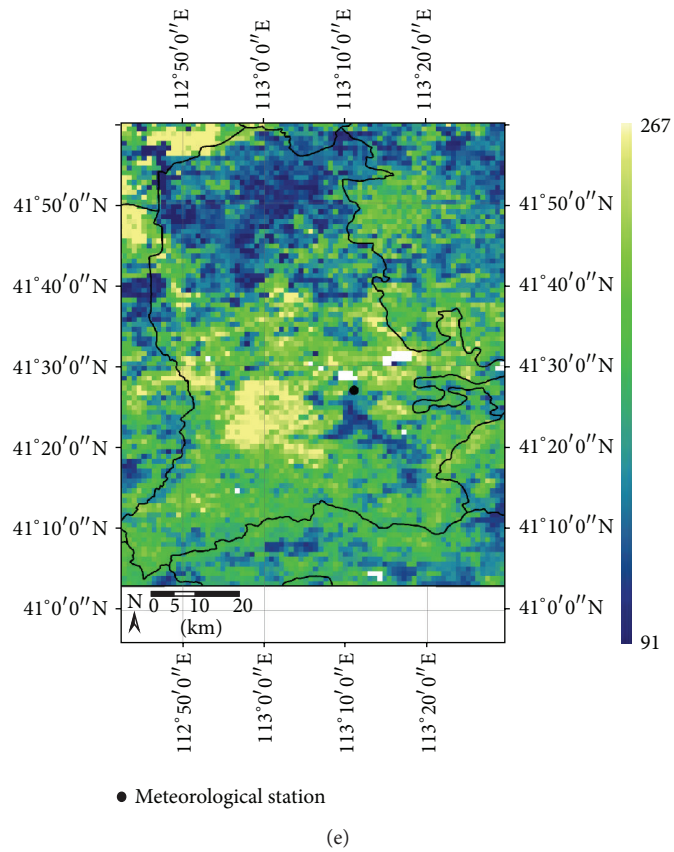


Figure 5: Continued.



(e)

Figure 5: (a) greenup onset, (b) maturity onset, (c) senescence onset, (d) dormancy onset, and (e) the length of vegetation growing season.

In most regions, about 62 days are required from greenup to reach the mature phase, with relatively late mature in more northern regions. Senescence occurs at the beginning of August (247 day-of-year) without strong spatial trend. Dormancy onset begins at the end of October (318 day-of-year). The length of the growing season is about 190 days.

4.3. Comparison with Field Observation. Table 1 presents the field-observed dates versus phenological transition dates derived from MODIS NDVI. In situ phenology observations are collected from Chen and Li [17]. Greenup onset dates identified by NDVI and in situ observed dates showed general agreement. There is an excellent agreement between the dates of maturity onset determined by MODIS and the field observations. However, values corresponding to the satellite onset estimate of senescence and dormancy date lag field-observed values by about 15–35 days. The satellite-derived length of vegetation growing season is generally consistent with the surface observation.

Table 1: The phenophases derived from MODIS NDVI and field observation.

Phenophases	VI-based land surface phenology	Field observation
Greenup onset	124	131
Maturity onset	186	187
Senescence onset	247	211
Dormancy onset	318	303
Length of growing season	194	183

* In situ phenology observations are collected from Chen and Li [17].

5. Conclusion and Discussion

This paper presents a valid methodology to identify grassland phenophases using remote sensing data. The double logistic model has been demonstrated as a flexible, repeatable, and realistic way to reconstruct time series. The phenophases derived from satellite and observed on ground are generally similar. However, values corresponding to the satellite onset estimate of senescence and dormancy date lag field-observed values by about 15–35 days. This may be explained by the field observation protocol adopted in describing phonological dates from maturity to senescence.

To improve accuracy, validation is an essential issue in land surface phenology study over large areas. This requires sufficient comparison between land surface phenology and in situ values, which is challenging because the location of field observation and MODIS pixels may not match [19].

Conflicts of Interests

The authors have declared that no conflict of interests exists.

Acknowledgments

This study was supported by the Key Project of National Natural Science Foundation of China (NSFC, no. 41030101) and the National Natural Science Foundation of China (NSFC, no. 41101115).

References

[1] A. Hudson Dunn and K. M. de Beurs, "Land surface phenology of North American mountain environments using moderate resolution imaging spectroradiometer data," *Remote Sensing of Environment*, vol. 115, no. 5, pp. 1220–1233, 2011.

[2] C. G. Rosenzweig, D. J. Casassa, A. Karoly et al., *Climate Change 2007: Impacts, Adaptation and Vulnerability: Contribution of Working Group II To the Fourth Assessment Report of the Intergovernmental Panel on Climate Change*, vol. 4, Cambridge University Press, 2007.

[3] T. R. Karl, J. M. Melillo, T. C. Peterson, and S. J. Hassol, *Global Climate Change Impacts in the United States*, Cambridge University Press, 2009.

[4] K. Soudani, G. le Maire, E. Dufrêne et al., "Evaluation of the onset of green-up in temperate deciduous broadleaf forests derived from Moderate Resolution Imaging Spectroradiometer (MODIS) data," *Remote Sensing of Environment*, vol. 112, no. 5, pp. 2643–2655, 2008.

[5] E. Ivits, M. Cherlet, G. Tóth et al., "Combining satellite derived phenology with climate data for climate change impact assessment," *Global and Planetary Change*, vol. 88-89, pp. 85–97, 2012.

[6] K. Zhu and M. Wan, *A Productive Science—Phenology, Public Science*, 1963.

[7] T. H. Sparks and P. D. Carey, "The responses of species to climate over two centuries: an analysis of the Marsham phenological record, 1736–1947," *Journal of Ecology*, vol. 83, no. 2, pp. 321–329, 1995.

[8] T. Rötzer and F.-M. Chmielewski, "Phenological maps of Europe," *Climate Research*, vol. 18, pp. 249–257, 2001.

[9] Q. Ge, J. Dai, J. Zheng et al., "Advances in first bloom dates and increased occurrences of yearly second blooms in eastern China since the 1960s: further phenological evidence of climate warming," *Ecological Research*, vol. 26, no. 4, pp. 713–723, 2011.

[10] I. Chuine, G. Cambon, and P. Comtois, "Scaling phenology from the local to the regional level: advances from species-specific phenological models," *Global Change Biology*, vol. 6, no. 8, pp. 943–952, 2000.

[11] Y. Huang, D. Jiang, D. Zhuang, H. Ren, and Z. Yao, "Filling gaps in vegetation index measurements for crop growth monitoring," *African Journal of Agricultural Research*, vol. 6, no. 12, pp. 2920–2930, 2011.

[12] J. N. Hird and G. J. McDermid, "Noise reduction of NDVI time series: an empirical comparison of selected techniques," *Remote Sensing of Environment*, vol. 113, no. 1, pp. 248–258, 2009.

[13] X. Y. Zhang, M. A. Friedl, C. B. Schaaf et al., "Monitoring vegetation phenology using MODIS," *Remote Sensing of Environment*, vol. 84, no. 3, pp. 471–475, 2003.

[14] B. C. Reed, J. F. Brown, D. VanderZee, T. R. Loveland, J. W. Merchant, and D. O. Ohlen, "Measuring phenological variability from satellite imagery," *Journal of Vegetation Science*, vol. 5, no. 5, pp. 703–714, 1994.

[15] N. Delbart, L. Kergoat, T. L. Toan, J. Lhermitte, and G. Picard, "Determination of phenological dates in boreal regions using normalized difference water index," *Remote Sensing of Environment*, vol. 97, no. 1, pp. 26–38, 2005.

[16] L. Liang, M. D. Schwartz, and S. Fei, "Validating satellite phenology through intensive ground observation and landscape scaling in a mixed seasonal forest," *Remote Sensing of Environment*, vol. 115, no. 1, pp. 143–157, 2011.

[17] X. Q. Chen and J. Li, "Relationships between Leymus chinensis phenology and meteorological factors in Inner Mongolia grasslands," *Acta Ecologica Sinica*, vol. 29, no. 10, pp. 5280–5290, 2009.

[18] X. Y. Zhang, M. A. Friedl, and C. B. Schaaf, "Global vegetation phenology from Moderate Resolution Imaging Spectroradiometer (MODIS): evaluation of global patterns and comparison with in situ measurements," *Journal of Geophysical Research*, vol. 111, no. 4, 2006.

[19] S. Ganguly, M. A. Friedl, B. Tan, X. Zhang, and M. Verma, "Land surface phenology from MODIS: characterization of the collection 5 global land cover dynamics product," *Remote Sensing of Environment*, vol. 114, no. 8, pp. 1805–1816, 2010.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 207987, 8 pages
http://dx.doi.org/10.1155/2013/207987

Research Article

Ecological Vulnerability Assessment Integrating the Spatial Analysis Technology with Algorithms: A Case of the Wood-Grass Ecotone of Northeast China

Zhi Qiao,[1] Xi Yang,[2] Jun Liu,[2,3] and Xinliang Xu[3]

[1] *State Key Laboratory of Water Environment Simulation, School of Environment, Beijing Normal University, No. 19, Xinjiekouwai Street, Beijing 100875, China*
[2] *College of Geography and Tourism, Chongqing Normal University, No.12, TianChen Road, ShaPingBa District, Chongqing 400047, China*
[3] *Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences, A11 DaTun Road, ChaoYang District, Beijing 100101, China*

Correspondence should be addressed to Jun Liu; liujun_igsnrr@yahoo.com.cn

Received 25 February 2013; Accepted 27 March 2013

Academic Editor: Jianhong (Cecilia) Xia

This study evaluates ecological vulnerability of the wood-grass ecotone of northeast China integrating the spatial analysis technology with algorithms. An assessment model of ecological vulnerability is developed applying the Analytical Hierarchy Process. The composite evaluation index system is established on the basis of the analysis of contemporary status and potential problems in the study area. By the application of the evaluation model, ecological vulnerability index is calculated between 1990 and 2005. The results show that ecological vulnerability was mostly at a medium level in the study area, however the ecological quality was deteriorating. Through the standard deviational ellipse, the variation of ecological vulnerability can be spatially explicated. It is extremely significative for the prediction of the regions that will easily deteriorate. The deterioration zone was concentrating in the area of Da Hinggan Ling Mountain, including Xingan League, Chifeng, Tongliao, and Chengde, whereas the improvement zone was distributing in the north-central of Hulunbeier.

1. Introduction

Ecosystem provides the essential material that is indispensable for human subsistence and development [1]. Nevertheless, the ecosystem has been taking a turn for the worse, which results from both the global change and population growth [2]. The ecosystem exhibits a significant amount of characters that are subjected to the exotic environment as the ecological vulnerability. Some investigators conceptualized ecological vulnerability as a function of exposure, sensitivity, and adaptive capacity [3, 4]. The definition is valuable as it embraces its own characters for ecosystem, that is, experiencing internal or external system disturbance [5], the ability of a system to adjust its behavior and characteristics in order to enhance its capacity versus external stress [6], and the establishing principle of ecological vulnerability index system [7].

The assessment of ecological vulnerability is progressively important as it enables us to ascertain the potential problem and to stimulate eco-environment protection [8]. The origins of vulnerability assessment are social sciences and economic field; however there has been an increasing interest in the ecosystem over the last decades [9]. Numerous approaches are employed for the assessment of ecological vulnerability, for example, comprehensive evaluation method [10], indices weight method (IWM) [11], analytical hierarchy process (AHP) [10, 12], and spatial principal component analysis model (SPCA) [2]. The integration of spatial analysis technology with algorithms provides a powerful means for ecological vulnerability assessment and forecast. Many spatially explicit indicators of sensitivity, exposure and adaptive capacity are available, all of which are essential to ecological vulnerability

assessment and forecast, through mining from geographic information system (GIS) and remote sensing (RS).

The wood-grass ecotone of Northeast China is initially proposed as a vulnerable area by the Ministry of environmental protection of the People's Republic of China in 2008 [13]. So far, experiments on ecological vulnerability of the wood-grass ecotone were insufficient. Specifically, we concentrate on the subsequent issues: (1) What was the spatial-temporal variation of ecological vulnerability in the region between 1990 and 2005? (2) How does integrate spatial analysis technology and algorithms for ecological vulnerability assessment and forecast?

2. Study Area

The wood-grass ecotone of Northeast China, lying in between the Da Hinggan Ling mountain and Yanshan mountain, extends about 14 degrees of latitude (39°30′~53°20′N) and 11 degrees of longitude (115°02′ ~ 126°04′E). It contains six cities: Hulunbeier, Xingan League, Tongliao, Chifeng, Chengde, Zhangjiakou, with the area about 5.24×10^5 km^2 and the altitude 89~2683 m (Figure 1). Ecological problems, for instance, ecological transition characteristics and heterogeneity, grassland degeneration, and soil erosion [13], have seriously affected sustainable development of the region. More significant is the truth that the entire area plays as an ecological security barrier for Beijing, which is the capital of China.

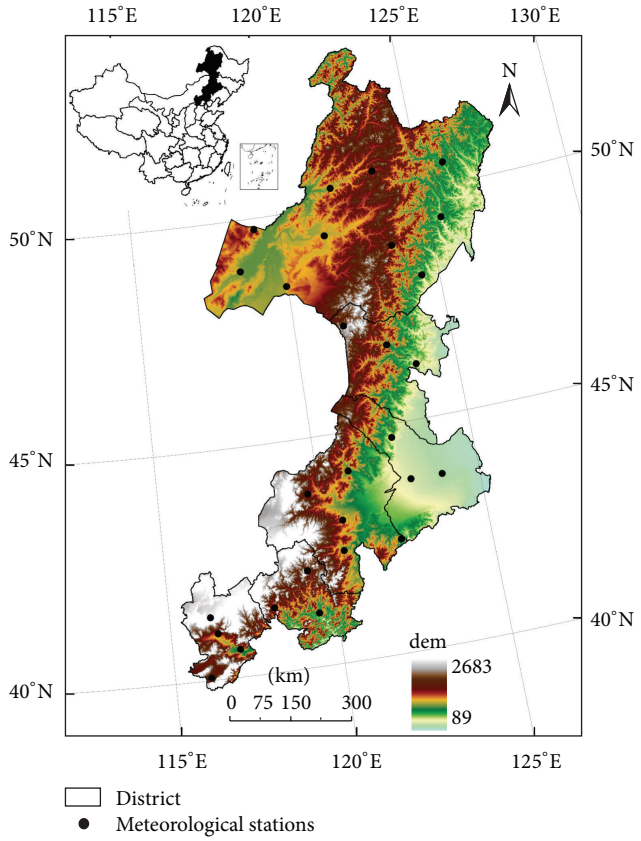


Figure 1: Distribution of the wood-grass ecotone of Northeast China.

3. Materials and Methods

The process of ecological vulnerability assessment involves the subsequent phases, that is, establishment of ecological vulnerability index system, modeling the assessment of ecological vulnerability, index calculation and standardization, ecological vulnerability assessment and classification, and variation analysis of ecological vulnerability (Figure 2).

3.1. Establishment of Ecological Vulnerability Index System. The study intends to measure the ecological vulnerability of the wood-grass ecotone of Northeast China. Through the investigation of contemporary status and potential problems of the study area, we summarize the characteristics as follows. Elevation and slope have a remarkable influence on the ecosystem because it lies in a mountainous area. The content of soil organic matter is 7%~10% in the east and middle regions; however it is only 0.5%~2% in the west. As a result of the infertile soil and thin soil layer, the soil is undoubtedly encroached by rain and wind. In addition, the overburdened agricultural activities seriously damage essentially vulnerable environment as a result of reclaiming forest and grass land. Considering the above investigations and combining with other research achievements carried out in the past few decades for the assessment of ecological vulnerability [1, 2, 14], eight elements are chosen in order to synthetically evaluate ecological vulnerability of the wood-grass ecotone of Northeast China. These components involve LUCC, DEM, soil texture, soil organic matter, precipitation, annual accumulated temperature (≥0°C), windy days in winter and spring (>6 m/s), and NDVI (Table 1). The above data used in the study are derived from Landsat thematic mapper (TM) image, digital elevation model (DEM), national soil field survey data, and meteorological data. The LUCC data is manually interpreted based on TM images, and there are six aggregated classes of land use, that is, cropland, forest, grassland, water body, bare land, and built-up land [15].

3.2. Modeling the Assessment of Ecological Vulnerability. Besides the establishment of ecological vulnerability index system, the additional key for the assessment of ecological vulnerability is determining the weight of individual evaluation indicator. In this paper, analytic hierarchy process (AHP) [16] is applied to generate a comprehensive decision for the assessment of ecological vulnerability.

The paper aims to make an assessment of ecological vulnerability; therefore the destination layer (level 1) is the ecological vulnerability index. The evaluation factors (level 2) for the criterion layer are *ecological suitability index*, *landscape pattern index*, and *land resources utilization degree index*. The corresponding quantitative indices (level 3) are *soil erosion sensitivity index* and *soil desertification sensitivity index*, *landscape unit plaques density* and *landscape evenness index*, and *land utilization degree composite index* (Table 1). The above system includes natural factors and anthropogenic

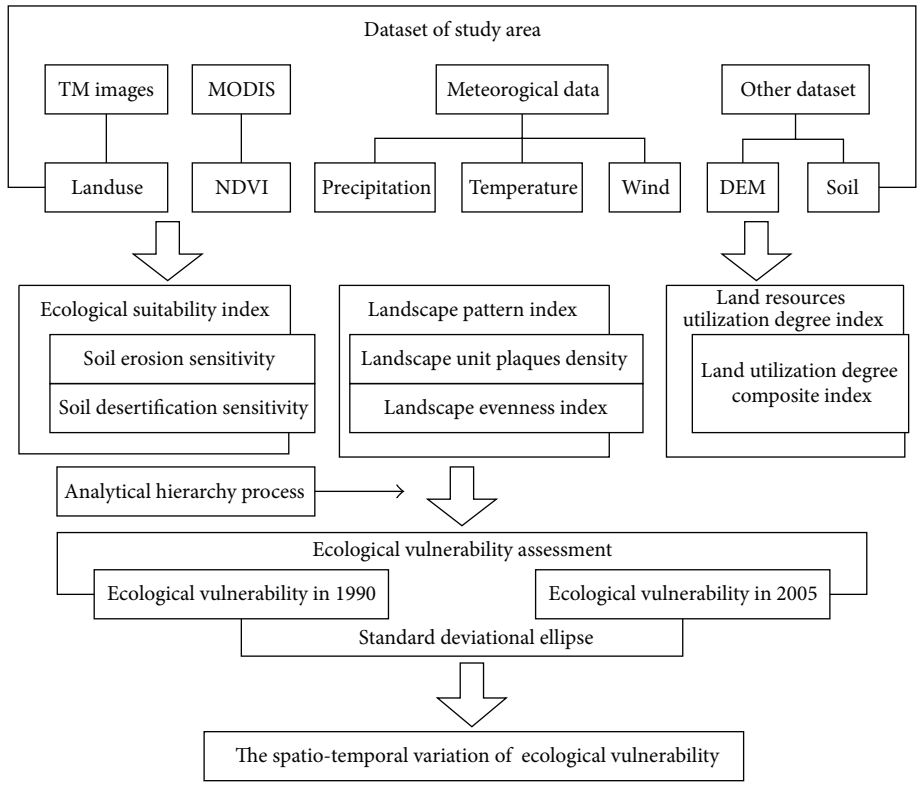


Figure 2: The workflow of ecological vulnerability assessment.

factors that directly relate with contemporary status and potential problems in the study area.

3.3. Index Calculation and Standardization

3.3.1. Ecological Suitability Index. Ecological suitability is determined by ecological sensitive degree of ecosystem. Sensitivity analysis is to distinguish the area which it is easily deteriorated by exotic environment. Sensitivity analysis is comprised of soil erosion sensitivity and soil desertification sensitivity on the basis of contemporary status and potential problems in the study area. Soil erosion sensitivity is determined by precipitation, DEM, soil, and LUCC. Soil desertification sensitivity is evaluated by average precipitation, annual accumulated temperature (≥0°C), windy days in winter and spring (>6 m/s), and NDVI.

(a) *Soil Erosion Sensitivity.* In the paper, we use the Wischmeier empirical formula to calculate the rainfall erosivity (R) [17]:

$$R = \sum_{i=1}^{i=12} 1.735 \times 10^{(1.5\log(P_i^2/P)-0.8188)}, \quad (1)$$

where R is the rainfall erosivity factor and its unit is MJ·mm·ha^{-1}·h^{-1}·year^{-1}, P_i is monthly rainfall in mm, and P is annual rainfall in mm.

Topographic factor (LS) is to estimate the relationship between topographical relief and soil erosion sensitivity.

Table 1: The ecological vulnerability index system of the wood-grass ecotone of Northeast China.

Destination layer	Criterion layer	Index layer	Data source
Ecological vulnerability	Ecological suitability index	Soil erosion sensitivity index	Precipitation
			DEM
			Soil texture
			Soil organic matter
			LUCC
		Soil desertification sensitivity index	Precipitation
			Annual accumulated Temperature (≥0°C)
			Windy days in winter and spring (>6 m/s)
			NDVI
	Landscape pattern index	Landscape unit plaques density	LUCC
		Landscape evenness index	
	Land resources utilization degree index	Land utilization degree composite index	

Table 2: Soil erosion sensitivity standard classification.

Index	Slight sensitivity	Light sensitivity	Medium sensitivity	Heavy sensitivity	Extreme sensitivity
R	≤25	25–100	100–400	400–600	>600
LS	≤20	20–50	51–100	101–300	>300
K	≤0.08	0.08–0.12	0.12–0.2	0.2-0.3	0.3–0.45
CM	Water, herb swamp, rice paddies	Broad-leaved forest, coniferous forest, grass, bush forest	Shrub grassland, double cropping crop	Desert, once cropping crop	No vegetation
Value	1	3	5	7	9
Classification value (SS)	1.0–2.0	2.1–4.0	4.1–6.0	6.1–8.0	>8.0

Table 3: Soil desertification sensitivity standard classification.

Index	Slight sensitivity	Light sensitivity	Medium sensitivity	Heavy sensitivity	Extreme sensitivity
Moisture degree index	>0.65	0.5–0.65	0.20–0.50	0.05–0.20	<0.05
Windy days in winter and spring (>6 m/s)	<15	15–30	30–45	45–60	>60
Soil texture	Pedestal rock	Viscidity	Gravel	Loamy texture	Sandiness
Vegetation cover	Dense	Moderate	Low	Thin	Bare land
Value	1	3	5	7	9
Classification value (DS)	1.0–2.0	2.1–4.0	4.1–6.0	6.1–8.0	>8.0

The paper uses the move window of 5 km × 5 km to extract the surface rolling on the basis of DEM.

The soil erodibility factor (K) is calculated through the following equation [18]:

$$K = \left\{0.2 + 0.3\exp\left[-0.0256S_a\left(1 - \frac{S_i}{100}\right)\right]\right\} \times \left(\frac{S_i}{C_l + S_i}\right)^{0.3} \times \left[1 - \frac{0.25C}{C + \exp(3.72 - 2.95C)}\right] \times \left[1 - \frac{0.7S_n}{S_n + \exp(-5.51 + 22.9S_n)}\right], \quad (2)$$

where K is soil erodibility factor and its unit is $t\cdot ha^{-1}\cdot h\cdot MJ^{-1}\cdot mm^{-1}\cdot ha^{-1}$. S_a, S_i and C_l is the percentage of sand, powder, and clay content in soil. $S_n = 1 - S_a/100$. C is carbon content in soil; it is the value of that organic content multiplied by *Bemmelen* (0.58 g C/g SOC) [19].

Cropping management factor (CM) is the factor used most often to compare the relative impacts of management options on conservation plans. It is then an estimate of the ratio of soil loss under actual conditions to losses experienced under the reference conditions. Because of the diversity of the type of the land cover and the administrative manner, the ability of the prevention of soil erosion is different (Table 2).

In summary, the calculation method of soil erosion sensitivity index is as follows:

$$SS = \sqrt[4]{\prod_{i=1}^{4} C_i}, \quad (3)$$

where SS is soil erosion sensitivity index, C_i is the value of each sensitivity factor, and i is the sensitivity factor (Table 2) [13].

(b) *Soil Desertification Sensitivity.* Similarly, the calculation method of soil desertification sensitivity is shown as follows:

$$DS = \sqrt[4]{\prod_{i=1}^{4} D_i}, \quad (4)$$

where DS is soil desertification sensitivity, D_i is the value of each sensitivity factor, and i is the sensitivity factor. In the paper, we use moisture degree index, soil texture, windy days in winter and spring, and vegetation cover to evaluate the soil desertification sensitivity (Table 3).

The subsequent equation is applied to calculate the moisture degree index, that is, the annual total precipitation (r) divides by annul accumulated temperature (≥0°C), $\sum\theta$,

$$K = \frac{r}{0.1 \times \sum\theta}. \quad (5)$$

Wind speed is took notes at each meteorological station; we interpolate those materials to count the windy days in winter and spring (>6 m/s) in the entire region.

According to the responce of soil texture to soil desertification sensitivity, the soil type is classified into five grades through international soil texture triangle table.

Vegetation cover shows the capability of preventing desertification. The greater the vegetation cover, the stronger the ability of prevention function. We use f to show vegetation cover

$$f = \frac{NDVI - NDVI_{min}}{NDVI_{max} - NDVI_{min}}, \quad (6)$$

where f is the vegetation cover, NDVI is normalized difference vegetation index, $\mathrm{NDVI}_{\min}$ is the minimum value of NDVI, and $\mathrm{NDVI}_{\max}$ is the maximum value of NDVI.

3.3.2. Landscape Pattern Index

(a) *Landscape Unit Patch Density (PD).* Patch density is a limited, but fundamental, aspect of landscape pattern. PD has the same basic utility as the number of patches as an index, except that it expresses the number of patches on a per unit area basis that facilitates comparisons among landscapes of varying size. We calculate PD with the following expression:

$$\mathrm{PD} = \frac{N}{A}, \tag{7}$$

where PD is landscape unit patch density, N is the patch number of each landscape, and A is the area of each landscape.

(b) *Landscape Evenness Index (SHEI).* SHEI equals minus the sum, across all patch types, of the proportional abundance of each patch type multiplied by that proportion, divided by the logarithm of the number of patch types. In other words, the observed Shannon's diversity index divided by the maximum Shannon's diversity index for that number of patch types. We calculate the landscape unit plaques density with the following expression:

$$\mathrm{SHEI} = \frac{-\sum_{i=1}^{m} \left(P_i \times \ln P_i\right)}{\ln m}, \tag{8}$$

where P_i is the proportion of the landscape occupied by patch type i. m is the number of patch types present in the landscape. SHEI = 0 when the landscape contains only 1 patch (i.e., no diversity) and approaches 0 as the distribution of area among the different patch types becomes increasingly uneven (i.e., dominated by 1 type). SHEI = 1 when distribution of area among patch types is perfectly even (i.e., proportional abundances are the same).

3.3.3. Land Resources Utilization Degree Index. Land resources utilization degree is expressed as the land utilization degree composite index. The connotation of land resources utilization degree that presents as the response indicator of ecological vulnerability is the limit of the land resource. The superior limit is the maximum degree of utilization of land resources, which means that it is not able to further develop the land resource, and vice versa. Where the value of bare land is 1, the value of forest, grass, and water body is 2, the value of cropland is 3, and the value of built-up land is 4, respectively.

3.3.4. Index Standardization. There are significant differences and diverse units for evaluation indicators. It is difficult to further evaluate ecological vulnerability through these heterogeneous indices indirectly. The value of evaluation indices must be standardized in a uniform measurement system

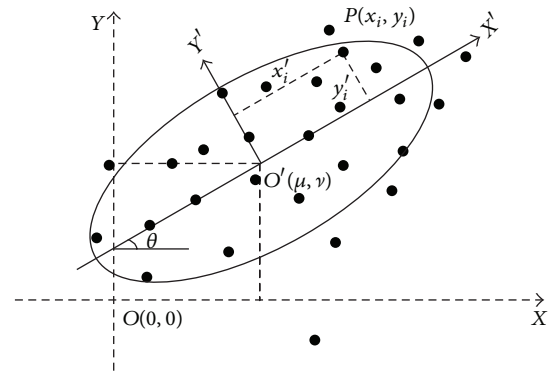


Figure 3: Standard deviational ellipse.

$$X'_{ij} = \frac{X_{ij} - X_{\min,j}}{X_{\max,j} - X_{\min,j}}, \tag{9}$$

where X'_{ij} represents the standardized value of factor j in grid i, X_{ij} represents the original value of factor j in grid i, $X_{\max,j}$, and $X_{\min,j}$ represent the maximum and minimum value of factor j in grid i, respectively.

3.4. Ecological Vulnerability Assessment and Classification. According to the evaluation factors and their weights calculated by analytical hierarchy process, ecological vulnerability index (EVI) is calculated as follows:

$$\mathrm{EVI} = \alpha \cdot \mathrm{SS} + \beta \cdot \mathrm{DS} + \gamma \cdot \mathrm{PD} + \omega \cdot \mathrm{SHEI} + \phi \cdot \mathrm{La}, \tag{10}$$

where EVI is ecological vulnerability index, SS, DS, PD, SHEI, and La are, respectively, soil erosion sensitivity index, soil desertification sensitivity index, landscape unit patch density, landscape evenness index, and land utilization degree composite index, and α, β, γ, ω, and ϕ are the weight values of each evaluative factor, respectively.

The scores of EVI scatter between 0 (the lowest grade) and 1 (the highest grade). To further reveal the characteristics of ecological vulnerability, we reclassify the grade of ecological vulnerability through the ArcGIS software and following natural breaks classification. The standard of ecological vulnerability is slight (0–0.2), light (0.2–0.4), medium (0.4–0.6), heavy (0.6–0.8), and extreme (0.8–1).

3.5. Variation Analysis of Ecological Vulnerability Based on the Standard Deviational Ellipse. Although there are numerous methods and cases of ecological vulnerability assessment, studies on forecasting ecological vulnerability are insufficient. The paper aims to forecast spatial variation of ecological vulnerability on the basis of the standard deviational ellipse.

The standard deviational ellipse aims to evaluate the trend of a set of points (Figure 3). The work includes two steps: ascertaining the mean center and determining the dispersion of the scattered points [20, 21].

Firstly, move the original coordinate system to the mean center (μ, ν) of the set of n units studied

$$\mu = \frac{\sum_{i=1}^{n} x_i}{n}, \qquad \nu = \frac{\sum_{i=1}^{n} y_i}{n}, \tag{11}$$

where μ, ν are the mean value of x_i, y_i in the original coordinate system (XOY), respectively.

Then calculate the standard deviation ($\sigma_{y'}$) of the y coordinates of the units,

$$\sigma_{y'} = \sqrt{\sum_{i=1}^{n} \frac{\left(y_i'\right)^2}{n}}, \tag{12}$$

where y_i' is the coordinate of the units in the transformed coordinate system $X'O'Y'$. Similarly, x_i' is calculated as the above method.

Finally, rotate the coordinate system XOY about the new origin (μ, ν) by angle θ $(0 < \theta \leq 2\pi)$ and calculate the standard deviation ($\sigma_{y'}$) of the Y' coordinates again,

$$\sigma_{y'} = \sqrt{\frac{\sum_{i=1}^{n} \overline{y_i}^2 \cos^2\theta - 2\sum_{i=1}^{n} \overline{x_i}\,\overline{y_i} \sin\theta\cos\theta + \sum_{i=1}^{n} \overline{x_i}^2 \sin^2\theta}{n}}, \tag{13}$$

where $\overline{x_i}$ and $\overline{y_i}$ are the coordinates of the units in the rotated coordinate system $X'O'Y$ and $\sigma_{y'}$ is the standard deviation of the Y' coordinates. A study of the foregoing equations shows that the locus of the $\sigma_{y'}$ value as the axis rotates about the mean center is an ellipse. In order to plot the ellipse on the map, it is necessary to locate the major and minor axes and to calculate the corresponding $\sigma_{y'}$ values.

When the $\sigma_{y'}$ value obtains the minimum value in the rotated coordinate system, the rotated angle θ is the direction of scattered points. Then it can get the minimum value through calculating the derivative of $\sigma_{y'}$ for (12)

$$\frac{d\sigma_{y'}}{d\theta} = \frac{1}{n\sigma_{y'}}\left[\sum_{i=1}^{n} \overline{x_i}^2 \sin\theta\cos\theta - \sum_{i=1}^{n} \overline{x_i}\,\overline{y_i}\left(\cos^2\theta - \sin^2\theta\right) - \sum_{i=1}^{n} \overline{y_i}^2 \sin\theta\cos\theta\right]. \tag{14}$$

Solving for θ,

$$\tan\theta = \frac{\left(\sum_{i=1}^{n} \overline{x_i}^2 - \sum_{i=1}^{n} \overline{y_i}^2\right) \pm \sqrt{\left(\sum_{i=1}^{n} \overline{x_i}^2 - \sum_{i=1}^{n} \overline{y_i}^2\right)^2 + 4\left(\sum_{i=1}^{n} \overline{x_i}\,\overline{y_i}\right)^2}}{2\sum_{i=1}^{n} \overline{x_i}\,\overline{y_i}}. \tag{15}$$

If θ value is substituted for the variable in (13) and (14), the maximum and minimum $\sigma_{y'}$ values are given. These are the semimajor and semiminor axes of the standard deviational ellipse. The θ value shows the distribution of scattered points.

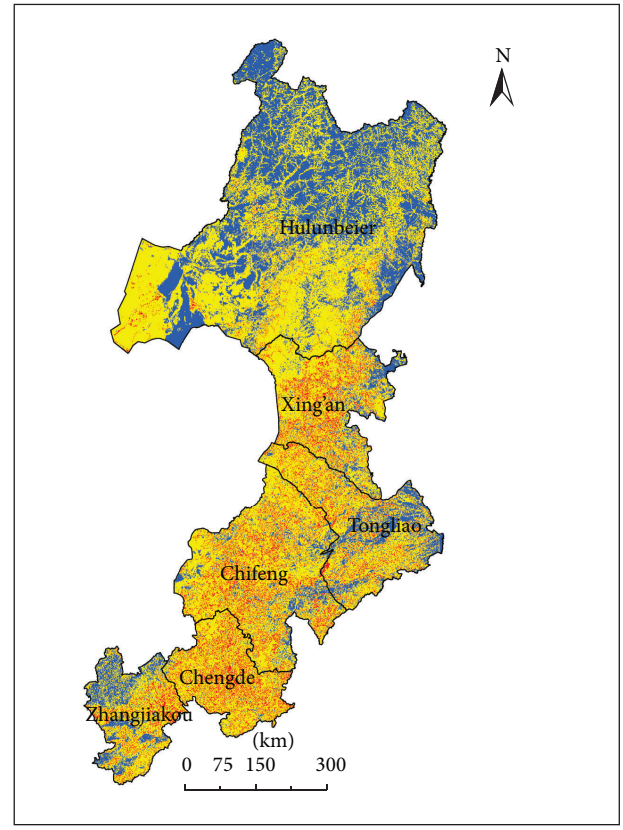


Figure 4: Spatial distribution of ecological vulnerability in 2005.

4. Result Analyses

4.1. The Assessment of Ecological Vulnerability in 2005. The ecological vulnerability of the wood-grass ecotone of northeast China is calculated according to the above method in Section 3.3. The largest area was medium-vulnerability zone, which accounted for 59.97% and mainly distributed in Hulunbeier and Chengde, followed by light-vulnerability zone accounted for 28.99%. Heavy-vulnerability zone accounted for 11.04%, and it was principally allocated in Chengde, Chifeng, Xingan League, and Zhangjiakou (Figure 4).

4.2. The Spatio-Temporal Variation of Ecological Vulnerability between 1990 and 2005. Statistical analysis indicates that areas of medium-vulnerability zone and heavy-vulnerability zone both increased between 1990 and 2005. Medium-vulnerability zone increased from 42.44% to 59.97%, which had the fastest growth among all zones. It is worthwhile to note that heavy-vulnerability zone increased by 5.64 $\times$ 10^4 km^2, and the percentage increased from 0.27% to 11.04%. Light vulnerability zone decreased from 56.98% to 28.99%, which had the fastest reduction among all zones. Taken altogether, ecological quality was deteriorating during

Table 4: Variation matrix of the ecological vulnerability between 1990 and 2005 (km^2).

Ecological vulnerability	Slight	Light	Medium	Heavy
Slight	0	1634	0	0
Light	0	117882	163287	17442
Medium	0	32411	150476	39531
Heavy	0	0	530	883

1990~2005. Transfer area from slight-vulnerability zone to light-vulnerability zone was 1634 km^2. Light-vulnerability zones converted into medium-vulnerability zone by 1.63×10^5 km^2, and light-vulnerability zones converted into heavy-vulnerability zone by 1.74×10^4 km^2, which accounted for 55% and 6% of the area of light vulnerability in 1990, respectively (Table 4).

Transfer zones can be explored through the standard deviational ellipse. There were six kinds of variations in the study area. Hence we use different colors to display those variations (Figure 5). There were four kinds of variations that indicate deterioration for ecological environment and other two kinds of variations that illustrate improvement for ecological environment. For the four kinds of deterioration zones, the zone changing from slight-vulnerability to light-vulnerability zone appeared in the southwest of Hulunbeier. And the other three deterioration zones concentrated in the Da Hinggan Ling mountain, including Xingan League, Chifeng, Tongliao, and Chengde. It showed that ecological circumstance was the absence of stabilization; both anthropogenic activity further endangered the environment in these regions. Corresponding measures must be taken to protect those deterioration zones. The improvement zone predominantly distributed in the north central of Hulunbeier, simultaneously.

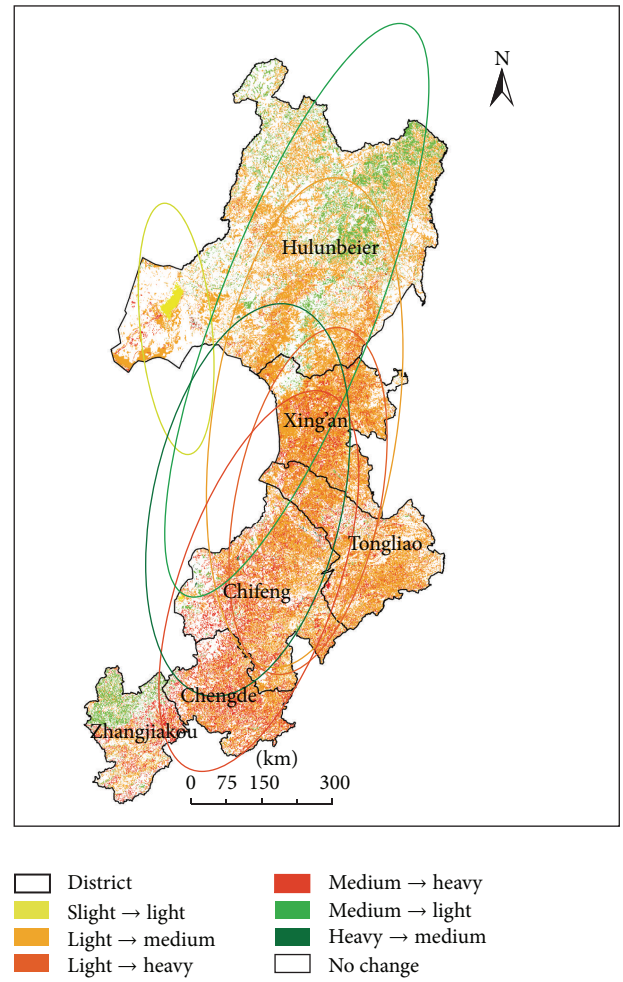


Figure 5: Spatial distribution of ecological vulnerability changes.

5. Discussions and Conclusions

Comparing spatio-temporal variation of the ecological vulnerability, the area of medium-vulnerability zone and heavy-vulnerability zone increased, simultaneously; the area of slight-vulnerability zone and light-vulnerability zone decreased. It was emphasized that the ecological quality became worse between 1990 and 2005 in the wood-grass ecotone of northeast China. There were numerous reasons impacting the spatio-temporal variation of ecological vulnerability, that is, global climate change, vegetation degradation, soil erosion, topographical relief, and rapid population growth.

The standard deviational ellipse is positive to spatially explore and forecast the deterioration zones. In the middle and south of the study area, the ecological environment deteriorated. At the same time, the ecological environment got better in the north of the study area. The method is also important as it is of significant benefit to identify areas at risk that will threaten sustainable development. Of course, the method can similarly discover areas where the quality of ecological environment is improving in virtue of the implementation of environmental protection. This study indicates that the assessment of ecological vulnerability can be improved with high spatio-temporal resolution through integrating GIS and RS with algorithms. It is more likely to be accepted by the local governments who would implement the recommended policies.

Acknowledgments

This study was supported by the National Natural Science Foundation of China (NSFC, no. 41101115) and Key Technologies R&D Program of China (2013BAC04B03).

References

[1] P. H. Qiu, S. J. Xu, G. Z. Xie, B. N. Tang, H. Bi, and L. S. Yu, "Analysis of the ecological vulnerability of the western Hainan Island based on its landscape pattern and ecosystem sensitivity," *Acta Ecologica Sinica*, vol. 27, no. 4, pp. 1257–1264, 2007.

[2] S. Y. Wang, J. S. Liu, and C. J. Yang, "Eco-environmental vulnerability evaluation in the Yellow River Basin, China," *Pedosphere*, vol. 18, no. 2, pp. 171–182, 2008.

[3] H. Eakin and A. L. Luers, "Assessing the vulnerability of social-environmental systems," *Annual Review of Environment and Resources*, vol. 31, pp. 365–394, 2006.

[4] G. C. Gallopín, "Linkages between vulnerability, resilience, and adaptive capacity," *Global Environmental Change*, vol. 16, no. 3, pp. 293–303, 2006.

[5] J. J. MaCarthy, O. F. Canziani, N. A. Leary, D. J. Dokken, and K. S. White, *Climate Change 2001: Impact, Adaptation, and Vulnerability*, Cambridge University Press, Cambridge. UK, 2001.

[6] N. Brook, "Vulnerability, risk and adaptation: a conceptual framework," Working Paper 38, Tyndall Centre for Climate Change Research, University of East Anglia, Norwich, UK, 2003.

[7] J. A. David, J. D. Andrew, and C. S. Lindsay, "Using principal component analysis for information-rich socio-ecological vulnerability mapping in Southern Africa," *Applied Geography*, vol. 35, no. 1-2, pp. 515–524, 2012.

[8] X. Ying, G. M. Zeng, G. Q. Chen, L. Tang, K. L. Wang, and D. Y. Huang, "Combining AHP with GIS in synthetic evaluation of eco-environment quality—a case study of Hunan Province, China," *Ecological Modelling*, vol. 209, no. 2–4, pp. 97–109, 2007.

[9] H. J. de Lange, S. Sala, M. Vighi, and J. H. Faber, "Ecological vulnerability in risk assessment—a review and perspectives," *Science of the Total Environment*, vol. 408, no. 18, pp. 3871–3879, 2010.

[10] X. M. Li, M. Min, and C. F. Tan, "The functional assessment of agricultural ecosystems in Hubei Province, China," *Ecological Modelling*, vol. 187, no. 2-3, pp. 352–360, 2005.

[11] X. J. Li, J. Peterson, G. J. Liu, and L. Qian, "Assessing regional sustainability: the case of land use and land cover change in the middle Yiluo catchment of the Yellow River Basin, China," *Applied Geography*, vol. 21, no. 1, pp. 87–106, 2001.

[12] P. Klungboonkrong and M. A. P. Taylor, "A microcomputer-based system for multicriteria environmental impacts evaluation of urban road networks," *Computers, Environment and Urban Systems*, vol. 22, no. 5, pp. 425–446, 1998.

[13] Z. Qiao and X. L. Xu, "Assessment of soil erosion sensitivity and key factors identification in the Wood-Grass ecotone of northeast China," *Journal of Natural Resources*, vol. 27, no. 8, pp. 1349–1361, 2012.

[14] G. B. Song, Y. Chen, and M. R. Tian, "The ecological vulnerability evaluation in southwestern mountain region of China based on GIS and AHP method," *Procedia Environmental Sciences*, vol. 2, pp. 465–475, 2010.

[15] J. Liu, M. Liu, H. Tian et al., "Spatial and temporal patterns of China's cropland during 1990–2000: an analysis based on Landsat TM data," *Remote Sensing of Environment*, vol. 98, no. 4, pp. 442–456, 2005.

[16] M. R. Rahman, Z. H. Shi, and C. Chongfa, "Soil erosion hazard evaluation—an integrated use of remote sensing, GIS and statistical approaches with biophysical parameters towards management strategies," *Ecological Modelling*, vol. 220, no. 13-14, pp. 1724–1734, 2009.

[17] K. Meusburger, N. Konz, M. Schaub, and C. Alewell, "Soil erosion modelled with USLE and PESERA using QuickBird derived vegetation parameters in an alpine catchment," *International Journal of Applied Earth Observation and Geoinformation*, vol. 12, no. 3, pp. 208–215, 2010.

[18] J. R. Williams, S. L. Neitsch, and J. G. Arnold, *Soil and Water Assessment Tool—User's Manuam*, Blackland Research Center, Texas Agricultural Experiment Station, Texas, Tex, USA, 1999.

[19] S. Q. Wang and C. H. Zhou, "Estimating soil carbon reservoir of terrestrial ecosystem in China," *Geographical Research*, vol. 18, no. 4, pp. 349–356, 1999.

[20] D. W. Lefever, "Measuring geographic concentration by means of the standard deviational ellipse," *The American Journal of Sociology*, vol. 32, no. 1, pp. 88–94, 1926.

[21] J. X. Gong, "Clarifying the standard deviational ellipse," *Geographical Analysis*, vol. 34, no. 2, pp. 155–167, 2002.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 951312, 10 pages
http://dx.doi.org/10.1155/2013/951312

Research Article

Nonstationary INAR(1) Process with qth-Order Autocorrelation Innovation

Kaizhi Yu,[1] Hong Zou,[2] and Daimin Shi[1]

[1] *Statistics School, Southwestern University of Finance and Economics, Chengdu, Sichuan 611130, China*
[2] *School of Economics, Southwestern University of Finance and Economics, Chengdu, Sichuan 611130, China*

Correspondence should be addressed to Kaizhi Yu; kaizhiyu.swufe@gmail.com

Received 7 January 2013; Accepted 17 February 2013

Academic Editor: Fuding Xie

This paper is concerned with an integer-valued random walk process with qth-order autocorrelation. Some limit distributions of sums about the nonstationary process are obtained. The limit distribution of conditional least squares estimators of the autoregressive coefficient in an auxiliary regression process is derived. The performance of the autoregressive coefficient estimators is assessed through the Monte Carlo simulations.

1. Introduction

In many practical settings, one often encounters integer-valued time series, that is, counts of events or objects at consecutive points in time. Typical examples include daily large transactions of Ericsson B, monthly number of ongoing strikes in a particular industry, number of patients treated each day in an emergency department, and daily counts of new swine flu cases in Mexico. Since most traditional representations of dependence become either impossible or impractical; see Silva et al. [1], this area of research did not attract much attention until the early 1980s. During the last three decades, a number of time series models have been developed for discrete-valued data. It has become increasingly important to gain a better understanding of the probabilistic properties and to develop new statistical techniques for integer-valued time series.

The existing models can be broadly classified into two types: thinning operator models and regression models. Recently the thinning operator models have been greatly developed; see a survey by Weiß [2]. A number of innovations have been made to model various integer-valued time series. For instance, Ferland et al. [3] proposed an integer-valued GARCH model to study overdispersed counts, and Fokianos and Fried [4], Weiß [5], and Zhu and Wang [6–8] made further studies. Bu and McCabe [9] considered the lag-order selection for a class of integer autoregressive models, while Enciso-Mora et al. [10] discussed model selection for general integer-valued autoregressive moving-average processes. Silva et al. [1] addressed a forecasting problem in an INAR(1) model. McCabe et al. [11] derived efficient probabilistic forecasts of integer-valued random variables. Several other models and thinning operators were proposed as well.

Random coefficient INAR models were studied by Zheng et al. [12, 13] and Gomes and e Castro [14], while random coefficient INMA models were proposed by Yu et al. [15, 16]. Kachour and Yao [17] introduced a class of autoregressive models for integer-valued time series using the rounding operator. Kim and Park [18] proposed an extension of integer-valued autoregressive INAR models by using a signed version of the thinning operator. The signed thinning operator was developed by Zhang et al. [19] and Kachour and Truquet [20].

However, these models focus exclusively on stationary processes, whereas nonstationary processes are often encountered in reality. Some studies have examined asymptotic properties of nonstationary INAR models for nonstatinaory time series. Hellström [21] focused on the testing of unit root in INAR(1) models and provided small sample distributions for the Dickey-Fuller test statistic. Ispány et al. [22] considered a nearly nonstationary INAR(1) process. It was shown that the limiting distribution of the conditional least squares estimator for the coefficient is normal. Györfi et al. [23]

proposed a nonstationary inhomogeneous INAR(1) process, where the autoregressive type coefficient slowly converges to one. Kim and Park [18] considered a process called integer-valued autoregressive process with signed binomial thinning to handle a nonstationary integer-valued time series with a large dispersion. Drost et al. [24] studied the asymptotic properties of this "near unit root" situation. They found that the limit experiment is Poissonian. Barczy et al. [25] proved that the sequence of appropriately scaled random step functions formed from an unstable INAR(p) process that converges weakly towards a squared Bessel process.

However, to our knowledge, few effort has been devoted to studying nonstationary INAR(1) models with an INMA innovation. In this paper, we aim to fill in this gap in the literature. We consider a new nonstatioary INAR(1) process in which the innovation follows a qth-order moving average process (NSINARMA$(1, q)$). Similar to the studies of nonstationary processes for continuous time series models, we need to accommodate two stochastic processes, one as the "true process" and the other as an "auxiliary regression process." In this paper, we study particularly statistical properties of the conditional least squares (CLS) estimators of the autoregressive coefficient in the auxiliary integer-valued autoregression process, when the "true process" is nonstationary integer-valued autoregression with an innovation which has an integer-valued moving average part.

The rest of this paper is organized as follows. In Section 2, our nonstationary thinning model with INMA(q) innovation is described, and some statistical properties are established. In Section 3, the limiting distribution of the autoregressive coefficient in the auxiliary regression process is derived. In Section 4, simulation results for the CLS estimator are presented. Finally, concluding remarks are made in Section 5.

2. Definition and Properties of the NSINARMA$(1, q)$ Process

In this section, we consider a nonstationary INAR(1) process which can be used to deal with the autocorrelation of innovation process.

Definition 1. An integer-valued stochastic process $\{X_t\}$ is said to be the NSINARMA$(1, q)$ process if it satisfies the following recursive equations:

$$\begin{aligned} X_t &= X_{t-1} + u_t, \\ u_t &= \varepsilon_t + \theta_1 \circ \varepsilon_{t-1} + \theta_2 \circ \varepsilon_{t-2} + \cdots + \theta_q \circ \varepsilon_{t-q}, \end{aligned} \tag{1}$$

where $\{\varepsilon_t\}$ is a sequence of i.i.d. nonnegative integer-valued random variables with finite mean $\mu_\varepsilon < \infty$, variance $\sigma_\varepsilon^2 < \infty$, and probability mass function f_ε. All counting series $\theta_k \circ \varepsilon_{t-k}$ are mutually independent, and $\theta_k \in [0, 1]$, $k = 1, \ldots, q$. For the sake of convenience, we suppose that the process starts from zero, more precisely, $X_0 = 0$.

It is easy to see the following results of X_t hold.

Proposition 2. *For $t \geq 1$, one has*

(i) $E(X_t X_{t-1}) = X_{t-1} + (1 + \theta_1 + \cdots + \theta_q)\mu_\varepsilon$,

(ii) $E(X_t) = t(1 + \theta_1 + \cdots + \theta_q)\mu_\varepsilon$,

(iii) $\mathrm{Var}(X_t X_{t-1}) = \mu_\varepsilon \sum_{k=1}^{q} \theta_k(1 - \theta_k) + \sigma_\varepsilon^2(1 + \sum_{k=1}^{q} \theta_k^2)$,

(iv) $\mathrm{Var}(X_t) = t(\mu_\varepsilon \sum_{k=1}^{q} \theta_k(1 - \theta_k) + \sigma_\varepsilon^2(1 + \sum_{k=1}^{q} \theta_k^2))$.

Proof. It is straightforward to get (i) to (iii). We prove (iv) by induction with

$$\begin{aligned} &\mathrm{Var}\left(X_t\right) \\ &\quad = \mathrm{Var}\left(E\left(X_t \mid X_{t-1}\right)\right) + E\left(\mathrm{Var}\left(X_t \mid X_{t-1}\right)\right) \\ &\quad = \mathrm{Var}\left(X_{t-1}\right) + \left(\mu_\varepsilon \sum_{k=1}^{q} \theta_k\left(1 - \theta_k\right) + \sigma_\varepsilon^2\left(1 + \sum_{k=1}^{q} \theta_k^2\right)\right) \end{aligned} \tag{2}$$

and the initial value $X_0 = 0$. □

3. Estimation Methods

Similar to the continuous time series process with unit roots, we focus on the properties of the autoregressive coefficient estimator in an auxiliary regression process when the true process follows the nonstatioary INAR(1) process defined as Definition 1.

Suppose that the auxiliary regression process X_t is an INAR(1) model,

$$X_t = \alpha \circ X_{t-1} + \nu_t, \tag{3}$$

where $\{\nu_t\}$ is a sequence of i.i.d. nonnegative integer-valued random variables with finite mean μ_ν and variance σ_ν^2. We are interested in the properties of α when the true process is a nonstatioary INAR(1) model with moving average components. In this paper, we consider a conditional least squares (CLSs) estimator. An advantage of this method is that it does not require specifying the exact family of distributions for the innovations.

Let

$$Q\left(\beta\right) = \sum_{t=1}^{T}\left(X_t - \alpha \circ X_{t-1} - \mu_\nu\right)^2, \tag{4}$$

with $\beta = (\alpha, \mu_\nu)$, be the CLS criterion function. The CLS estimators of α and μ_ν are obtained by minimizing Q and are given by

$$\widehat{\alpha} = \frac{T\sum_{t=1}^{T} X_{t-1}X_t - \left(\sum_{t=1}^{T} X_{t-1}\right)\left(\sum_{t=1}^{T} X_t\right)}{T\sum_{t=1}^{T} X_{t-1}^2 - \left(\sum_{t=1}^{T} X_{t-1}\right)^2}, \tag{5}$$

$$\widehat{\mu}_\nu = T^{-1}\left(\sum_{t=1}^{T} X_t - \widehat{\alpha}\sum_{t=1}^{T} X_{t-1}\right). \tag{6}$$

Similar to studying the unit root in continuous time series, we are only concerned with statistical properties of

the autoregressive coefficient estimator. For the nonstationary of continuous-valued time series, we often need to examine whether the characteristic polynomial of AR(1) process has a unit root. Thus, we want to see if we can find the limiting distribution of the autoregressive coefficient estimator. Let us present a result that is needed later on.

Lemma 3. *Suppose that Z_t follows a random walk without drift,*

$$Z_t = Z_{t-1} + w_t, \tag{7}$$

where $Z_0 = 0$ and $\{w_t\}$ is an i.i.d. sequence with mean zero and variance $\sigma_w^2 > 0$. Let "$\Rightarrow$" denote converges in distribution, and let $W(r)$ denote the standard Brownian motion. Then, one has the following properties:

(i) $T^{-1/2}\sum_{t=1}^{T} w_t \Rightarrow \sigma_w W(1)$,

(ii) $T^{-1}\sum_{t=1}^{T} Z_{t-1}w_t \Rightarrow (1/2)\sigma_w^2[W^2(1)-1]$,

(iii) $T^{-3/2}\sum_{t=1}^{T} tw_t \Rightarrow \sigma_w W(1) - \sigma_w \int_0^1 W(r)dr$,

(iv) $T^{-3/2}\sum_{t=1}^{T} Z_{t-1} \Rightarrow \sigma_w \int_0^1 W(r)dr$,

(v) $T^{-5/2}\sum_{t=1}^{T} tZ_{t-1} \Rightarrow \sigma_w \int_0^1 rW(r)dr$,

(vi) $T^{-2}\sum_{t=1}^{T} Z_{t-1}^2 \Rightarrow \sigma_w^2 \int_0^1 W^2(r)dr$,

(vii) $T^{-(n+1)}\sum_{t=1}^{T} t^n \to 1/(n+1)$, *for* $n = 0, 1, 2, \ldots$.

Proof. See Proposition 17.1 in Hamilton [26]. □

Theorem 4. *If all the assumptions of Definition 1 hold, then one has*

(i) $T^{-1}\sum_{t=1}^{T} \varepsilon_t(\theta_k \circ \varepsilon_{t-k})$ *converges in mean square to* $\theta_k\mu_\varepsilon^2$, *for* $k = 1, \ldots, q$,

(ii) $T^{-1}\sum_{t=1}^{T}(\theta_i \circ \varepsilon_{t-i})(\theta_j \circ \varepsilon_{t-j})$ *converges in mean square to* $\theta_i\theta_j\mu_\varepsilon^2$, *for* $1 \le i < j \le q$.

Proof. (i) First, we prove that $T^{-1}\sum_{t=1}^{T} \varepsilon_t \cdot (\theta_k \circ \varepsilon_{t-k})$ is integrable in mean square. Using the well-known results:

$$\begin{aligned} E\left(X\left(\beta \circ Y\right)\right) &= E\left(X\right)E\left(\beta \circ Y\right) = \beta E\left(X\right)E\left(Y\right), \\ E(\beta \circ Y)^2 &= E\left(E\left((\beta \circ Y)^2 \mid Y\right)\right) \\ &= \beta\left(1-\beta\right)E\left(Y\right) + \beta^2 E\left(Y^2\right), \end{aligned} \tag{8}$$

where X and Y are independent and $\beta \in [0, 1]$, we can derive that

$$\begin{aligned} &E\left(T^{-1}\sum_{t=1}^{T}\varepsilon_t\left(\theta_k \circ \varepsilon_{t-k}\right)\right)^2 \\ &\quad = T^{-2}\left(\sum_{t=1}^{T}E\left[\varepsilon_t^2(\theta_k \circ \varepsilon_{t-k})^2\right]\right. \\ &\qquad \left. +2\sum_{1\le m<n\le T}E\left(\varepsilon_m \cdot \theta_k \circ \varepsilon_{m-k}\right)\left(\varepsilon_n \cdot \theta_k \circ \varepsilon_{n-k}\right)\right) \\ &\quad = T^{-2}\left(T\left(\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\left(\theta_k\left(1-\theta_k\right)\mu_\varepsilon\right.\right.\right. \\ &\qquad \left.\left.\left. +\theta_k^2\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\right)\right)+\left(T^2-T\right)\theta_k^2\mu_\varepsilon^4\right) \\ &\quad = T^{-1}\left(\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\left(\theta_k\left(1-\theta_k\right)\mu_\varepsilon+\theta_k^2\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\right)\right) \\ &\qquad +\left(1-T^{-1}\right)\theta_k^2\mu_\varepsilon^4 \\ &\quad \le \left(\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\left(\theta_k\left(1-\theta_k\right)\mu_\varepsilon+\theta_k^2\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\right)\right) \\ &\qquad +\theta_k^2\mu_\varepsilon^4 < \infty. \end{aligned} \tag{9}$$

Therefore, $T^{-1}\sum_{t=1}^{T}\varepsilon_t \cdot (\theta_k \circ \varepsilon_{t-k})$ is integrable.

Next, we show that it converges to $\theta_k\mu_\varepsilon^2$ in mean square. In fact,

$$\begin{aligned} &E\left(T^{-1}\sum_{t=1}^{T}\varepsilon_t \cdot \theta_k \circ \varepsilon_{t-k} - \theta_k\mu_\varepsilon^2\right)^2 \\ &\quad = E\left(T^{-1}\sum_{t=1}^{T}\varepsilon_t \cdot \theta_k \circ \varepsilon_{t-k} - T^{-1}\sum_{t=1}^{T}E(\varepsilon_t \cdot \theta_k \circ \varepsilon_{t-k})\right)^2 \\ &\quad = T^{-2}\sum_{t=1}^{T}\left(E(\varepsilon_t \cdot \theta_k \circ \varepsilon_{t-k})^2 - \left(E\left(\varepsilon_t \cdot \theta_k \circ \varepsilon_{t-k}\right)\right)^2\right) \\ &\qquad + 2T^{-2}\sum_{t=k+1}^{T-k}\operatorname{cov}\left(\theta_k \circ \varepsilon_{t-k} \cdot \varepsilon_t, \theta_k \circ \varepsilon_t \cdot \varepsilon_{t+k}\right) \\ &\quad = T^{-1}\left(\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\left(\theta_k\left(1-\theta_k\right)\mu_\varepsilon\right.\right. \\ &\qquad \left.\left. +\theta_k^2\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\right)-\theta_k^2\mu_\varepsilon^4\right) \\ &\qquad + 2T^{-2}\left(T-2k\right)\theta_k^2\mu_\varepsilon^2\sigma_\varepsilon^2. \end{aligned} \tag{10}$$

From the assumptions $\mu_\varepsilon < \infty$, $\sigma_\varepsilon^2 < \infty$, and $\theta_k \in [0, 1]$, we get $\lim_{T\to\infty}E(T^{-1}\sum_{t=1}^{T}\varepsilon_t \cdot (\theta_k \circ \varepsilon_{t-k}) - \theta_k\mu_\varepsilon^2)^2 = 0$.

This completes the proof for (i).

(ii) Without loss of generality, we assume that $1 \le i < j \le q$. We have

$$
\begin{aligned}
&E\left(T^{-1}\sum_{t=1}^{T}(\theta_i\circ\varepsilon_{t-i})\cdot(\theta_j\circ\varepsilon_{t-j})\right)^2\\
&\quad=T^{-2}\sum_{t=j}^{T}E\left((\theta_i\circ\varepsilon_{t-i})\left(\theta_j\circ\varepsilon_{t-j}\right)\right)^2\\
&\qquad+2T^{-2}\sum_{1\le m<n\le T}E\Big((\theta_i\circ\varepsilon_{m-i})\\
&\qquad\qquad\cdot\left(\theta_j\circ\varepsilon_{m-j}\right)(\theta_i\circ\varepsilon_{n-i})\\
&\qquad\qquad\cdot\left(\theta_j\circ\varepsilon_{n-j}\right)\Big)\\
&\quad=T^{-1}\left(\theta_i\left(1-\theta_i\right)\mu_\varepsilon+\theta_i^2\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\right)\\
&\qquad\times\left(\theta_j\left(1-\theta_j\right)\mu_\varepsilon+\theta_j^2\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\right)\\
&\qquad+T^{-2}\frac{1}{2}\left(T-j-1\right)\left(T-j\right)\theta_i^2\theta_j^2\mu_\varepsilon^4\\
&\quad\le\left(\theta_i\left(1-\theta_i\right)\mu_\varepsilon+\theta_i^2\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\right)\\
&\qquad\times\left(\theta_j\left(1-\theta_j\right)\mu_\varepsilon+\theta_j^2\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\right)\\
&\qquad+\frac{1}{2}\theta_i^2\theta_j^2\mu_\varepsilon^4<\infty.
\end{aligned}
\tag{11}
$$

Thus, $T^{-1}\sum_{t=1}^{T}(\theta_i\circ\varepsilon_{t-i})\cdot(\theta_j\circ\varepsilon_{t-j})$ is integrable.

Next, we prove that the limit holds $\lim_{T\to\infty}E\left(T^{-1}\sum_{t=1}^{T}(\theta_i\circ\varepsilon_{t-i})\cdot(\theta_j\circ\varepsilon_{t-j})-\theta_i\theta_j\mu_\varepsilon^2\right)^2=0$,

$$
\begin{aligned}
&E\left(T^{-1}\sum_{t=1}^{T}\left(\theta_i\circ\varepsilon_{t-i}\right)\cdot\left(\theta_j\circ\varepsilon_{t-j}\right)-\theta_i\theta_j\mu_\varepsilon^2\right)^2\\
&\quad=E\left(T^{-1}\sum_{t=1}^{T}\left(\theta_i\circ\varepsilon_{t-i}\right)\cdot\left(\theta_j\circ\varepsilon_{t-j}\right)\right.\\
&\qquad\left.-T^{-1}\sum_{t=1}^{T}E\left(\left(\theta_i\circ\varepsilon_{t-i}\right)\cdot\left(\theta_j\circ\varepsilon_{t-j}\right)\right)\right)^2\\
&\quad=T^{-2}\sum_{t=1}^{T}\mathrm{Var}\left(\left(\theta_i\circ\varepsilon_{t-i}\right)\left(\theta_j\circ\varepsilon_{t-j}\right)\right)\\
&\qquad+2T^{-2}\sum_{1\le m<n\le T}\mathrm{cov}\left(\left(\theta_i\circ\varepsilon_{m-i}\right)\cdot\left(\theta_j\circ\varepsilon_{m-j}\right),\right.\\
&\qquad\qquad\left.\left(\theta_i\circ\varepsilon_{n-i}\right)\cdot\left(\theta_j\circ\varepsilon_{n-j}\right)\right)\\
&\quad=T^{-2}\sum_{t=1}^{T}\mathrm{Var}\left(\left(\theta_i\circ\varepsilon_{t-i}\right)\left(\theta_j\circ\varepsilon_{t-j}\right)\right)\\
&\qquad+T^{-2}\sum_{k=\max\{1,1-i+j\}}^{T-i+j}\mathrm{cov}\left(\left(\theta_i\circ\varepsilon_{k-i}\right)\cdot\left(\theta_j\circ\varepsilon_{k-j}\right),\right.\\
&\qquad\qquad\left.\left(\theta_i\circ\varepsilon_{k-2i+j}\right)\cdot\left(\theta_j\circ\varepsilon_{k-i}\right)\right)\\
&\quad=T^{-1}\left(\theta_i\left(1-\theta_i\right)\mu_\varepsilon+\theta_i^2\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\right)\\
&\qquad\times\left(\theta_j\left(1-\theta_j\right)\mu_\varepsilon+\theta_j^2\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right)\right)\\
&\qquad+T^{-2}\left(\left(T-i+j-\max\left\{1,1-i+j\right\}\right)\right.\\
&\qquad\qquad\left.\times\theta_i^2\theta_j\left(1-\theta_j\right)\mu_\varepsilon^3+\theta_i^2\theta_j^2\mu_\varepsilon^2\sigma_\varepsilon^2\right).
\end{aligned}
\tag{12}
$$

By using $\mu_\varepsilon<\infty$, $\sigma_\varepsilon^2<\infty$, and $\theta_k\in[0,1]$, with the above arguments, we get

$$
\lim_{T\to\infty}E\left(T^{-1}\sum_{t=1}^{T}(\theta_i\circ\varepsilon_{t-i})(\theta_j\circ\varepsilon_{t-j})-\theta_i\theta_j\mu_\varepsilon^2\right)^2=0. \tag{13}
$$

Then, $T^{-1}\sum_{t=1}^{T}(\theta_i\circ\varepsilon_{t-i})(\theta_j\circ\varepsilon_{t-j})$ converges in mean square to $\theta_i\theta_j\mu_\varepsilon^2$. □

Theorem 5. *If the process X_t is defined as in Definition 1, then one has*

(i) $T^{-1}\sum_{t=1}^{T}u_t\Rightarrow\mu_\varepsilon(1+\sum_{k=1}^{q}\theta_k)$,

(ii) $T^{-2}\sum_{t=1}^{T}X_{t-1}u_t\Rightarrow(1/2)\mu_\varepsilon^2(1+\sum_{k=1}^{q}\theta_k)^2$,

(iii) $T^{-2}\sum_{t=1}^{T}tu_t\Rightarrow(1/2)\mu_\varepsilon(1+\sum_{k=1}^{q}\theta_k)$,

(iv) $T^{-2}\sum_{t=1}^{T}X_{t-1}\Rightarrow(1/2)\mu_\varepsilon(1+\sum_{k=1}^{q}\theta_k)$,

(v) $T^{-3}\sum_{t=1}^{T}tX_{t-1}\Rightarrow(1/3)\mu_\varepsilon(1+\sum_{k=1}^{q}\theta_k)$,

(vi) $T^{-3}\sum_{t=1}^{T}X_{t-1}^2\Rightarrow(1/3)\mu_\varepsilon^2(1+\sum_{k=1}^{q}\theta_k)^2$.

Proof. Let $\varepsilon_t^*=\varepsilon_t-\mu_\varepsilon$, $\varepsilon_{k,t}^*=\theta_k\circ\varepsilon_{t-k}-\theta_k\mu_\varepsilon$, $k=1,\ldots,q$.

Then, we have the means and variances of ε_t^* and $\varepsilon_{k,t}^*$ given by

$$
\begin{aligned}
&E\left(\varepsilon_t^*\right)=0,\qquad\sigma_{\varepsilon_t^*}=\sqrt{\mathrm{Var}\left(\varepsilon_t^*\right)}=\sigma_\varepsilon,\\
&E\left(\varepsilon_{k,t}^*\right)=0,\quad\sigma_{\varepsilon_{k,t}^*}=\sqrt{\mathrm{Var}\left(\varepsilon_{k,t}^*\right)}=\sqrt{\theta_k\left(1-\theta_k\right)\mu_\varepsilon+\theta_k^2\sigma_\varepsilon^2}.
\end{aligned}
\tag{14}
$$

It is easy to see that $\{\varepsilon_t^*\}$ is a sequence of i.i.d. random variables. For a fixed k, $\{\varepsilon_{k,t}^*\}$ is also a sequence of i.i.d. random variables.

(i) Straightforward using the law of large numbers.

(ii) One has

$$
\begin{aligned}
\sum_{t=1}^{T}X_{t-1}u_t&=X_0u_1+X_1u_2+\cdots+X_{T-1}u_T\\
&=0+u_1u_2+\left(u_1+u_2\right)u_3\\
&\quad+\cdots+\left(u_1+\cdots+u_{T-1}\right)u_T\\
&=u_1\left(u_2+\cdots+u_T\right)+u_2\left(u_3+\cdots+u_T\right)\\
&\quad+\cdots+u_{T-1}u_T\\
&=\sum_{i<j}u_iu_j=\frac{1}{2}\left(\left(\sum_{t=1}^{T}u_t\right)^2-\sum_{t=1}^{T}u_t^2\right).
\end{aligned}
\tag{15}
$$

From the conclusion in (i), we get

$$\left(T^{-1}\sum_{t=1}^{T}u_t\right)^2 \Longrightarrow \mu_\varepsilon^2\left(1+\sum_{k=1}^{q}\theta_k\right)^2. \tag{16}$$

Note that

$$\begin{aligned}\sum_{t=1}^{T}u_t^2 &= \sum_{t=1}^{T}\varepsilon_t^2 + \sum_{k=1}^{q}\left(\sum_{t=1}^{T}(\theta_k\circ\varepsilon_{t-k})^2\right)\mu_X \\ &\quad + \sum_{k=1}^{q}\left(\sum_{t=1}^{T}\varepsilon_t\cdot(\theta_k\circ\varepsilon_{t-k})\right) \\ &\quad + 2\sum_{1\le i<j\le q}\left(\sum_{t=1}^{T}(\theta_i\circ\varepsilon_{t-i})\cdot(\theta_j\circ\varepsilon_{t-j})\right).\end{aligned} \tag{17}$$

Using the law of large numbers, we obtain

$$\begin{aligned}T^{-1}\sum_{t=1}^{T}\varepsilon_t^2 &\Longrightarrow E\left(\varepsilon_t^2\right) = \left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right), \\ T^{-1}\sum_{t=1}^{T}(\theta_k\circ\varepsilon_{t-k})^2 &\Longrightarrow E(\theta_k\circ\varepsilon_{t-k})^2 \\ &= \theta_k(1-\theta_k)\mu_\varepsilon + \theta_k^2\left(\mu_\varepsilon^2+\sigma_\varepsilon^2\right) \\ &= \sigma_{\varepsilon_{k,t}^*}^2, \quad k=1,\ldots,q.\end{aligned} \tag{18}$$

Recall Theorem 4, where $T^{-1}\sum_{t=1}^{T}\varepsilon_t(\theta_k\circ\varepsilon_{t-k})$ converges in mean square to $\theta_k\mu_\varepsilon^2$ and $T^{-1}\sum_{t=1}^{T}(\theta_i\circ\varepsilon_{t-i})(\theta_j\circ\varepsilon_{t-j})$ converges in mean square to $\theta_i\theta_j\mu_\varepsilon^2$, and thus

$$\begin{aligned}T^{-1}\sum_{t=1}^{T}\varepsilon_t\cdot(\theta_k\circ\varepsilon_{t-k}) &\Longrightarrow \theta_k\mu_\varepsilon^2, \\ T^{-1}\sum_{t=1}^{T}(\theta_i\circ\varepsilon_{t-i})\left(\theta_j\circ\varepsilon_{t-j}\right) &\Longrightarrow \theta_i\theta_j\mu_\varepsilon^2.\end{aligned} \tag{19}$$

Then, we get $T^{-1}\sum_{t=1}^{T}u_t^2 \Rightarrow (\mu_\varepsilon^2+\sigma_\varepsilon^2)+\sum_{k=1}^{q}\sigma_{\varepsilon_{k,t}^*}^2 + 2(\sum_{k=1}^{q}\theta_k+\sum_{1\le i<j\le q}\theta_i\theta_j)\mu_\varepsilon^2$.

Therefore,

$$\begin{aligned}T^{-2}\sum_{t=1}^{T}X_{t-1}u_t &= \frac{1}{2}\left(\left(T^{-1}\sum_{t=1}^{T}u_t\right)^2 - T^{-1}\left(T^{-1}\sum_{t=1}^{T}u_t^2\right)\right) \\ &\Longrightarrow \frac{1}{2}\left(\mu_\varepsilon^2\left(1+\sum_{k=1}^{q}\theta_k\right)^2 - 0\right) \\ &= \frac{1}{2}\mu_\varepsilon^2\left(1+\sum_{k=1}^{q}\theta_k\right)^2.\end{aligned} \tag{20}$$

(iii) Morever,

$$\begin{aligned}T^{-2}\sum_{t=1}^{T}tu_t &= T^{-2}\sum_{t=1}^{T}t\left(\varepsilon_t+\theta_1\circ\varepsilon_{t-1}+\cdots+\theta_q\circ\varepsilon_{t-q}\right) \\ &= T^{-2}\sum_{t=1}^{T}t\left(\varepsilon_t^*+\mu_\varepsilon\right) + \sum_{k=1}^{q}\left(T^{-2}\sum_{t=1}^{T}t\left(\varepsilon_{k,t}^*+\theta_k\mu_\varepsilon\right)\right) \\ &= T^{-1/2}\left(T^{-3/2}\sum_{t=1}^{T}t\varepsilon_t^* + \sum_{k=1}^{q}\left(T^{-3/2}\sum_{t=1}^{T}t\varepsilon_{k,t}^*\right)\right) \\ &\quad + \mu_\varepsilon\left(1+\sum_{k=1}^{q}\theta_k\right)\left(T^{-2}\sum_{t=1}^{T}t\right).\end{aligned} \tag{21}$$

From (iii) and (vii) of Lemma 3, we have

$$\begin{aligned}T^{-3/2}\sum_{t=1}^{T}t\varepsilon_t^* &\Longrightarrow \sigma_{\varepsilon_t^*}W(1) - \sigma_{\varepsilon_t^*}\int_0^1 W(r)\,dr, \\ T^{-3/2}\sum_{t=1}^{T}t\varepsilon_{k,t}^* &\Longrightarrow \sigma_{\varepsilon_{k,t}^*}W(1) - \sigma_{\varepsilon_{k,t}^*}\int_0^1 W(r)\,dr, \\ &\qquad k=1,\ldots,q, \\ T^{-2}\sum_{t=1}^{T}t &\longrightarrow \frac{1}{2}.\end{aligned} \tag{22}$$

Therefore, $T^{-2}\sum_{t=1}^{T}tu_t \Rightarrow 0+0+(1/2)\mu_\varepsilon(1+\sum_{k=1}^{q}\theta_k) = (1/2)\mu_\varepsilon(1+\sum_{k=1}^{q}\theta_k)$. Consider the following:

(iv)

$$\begin{aligned}T^{-2}\sum_{t=1}^{T}X_{t-1} &= T^{-2}\sum_{t=1}^{T-1}(T-t)\,u_t \\ &= T^{-2}\sum_{t=1}^{T-1}(T-t)\left(\varepsilon_t+\theta_1\circ\varepsilon_{t-1}+\cdots+\theta_q\circ\varepsilon_{t-q}\right) \\ &= T^{-2}\sum_{t=1}^{T}(T-t)\left(\varepsilon_t^*+\mu_\varepsilon\right) \\ &\quad + T^{-2}\sum_{k=1}^{q}\sum_{t=1}^{T}(T-t)\left(\varepsilon_{k,t}^*+\theta_k\mu_\varepsilon\right) \\ &= T^{-2}\sum_{t=1}^{T}(T-t)\,\varepsilon_t^* + \sum_{k=1}^{q}\left(T^{-2}\sum_{t=1}^{T}(T-t)\,\varepsilon_{k,t}^*\right) \\ &\quad + \mu_\varepsilon\left(1+\sum_{k=1}^{q}\theta_k\right)\left(T^{-2}\sum_{t=1}^{T}(T-t)\right)\end{aligned}$$

$$= T^{-1}\sum_{t=1}^{T}\varepsilon_t^* - T^{-1/2}\left(T^{-3/2}\sum_{t=1}^{T}t\varepsilon_t^*\right) + \sum_{k=1}^{q}\left(T^{-1}\sum_{t=1}^{T}\varepsilon_{k,t}^* - T^{-1/2}\left(T^{-3/2}\sum_{t=1}^{T}t\varepsilon_{k,t}^*\right)\right) + \mu_\varepsilon\left(1+\sum_{k=1}^{q}\theta_k\right)\left(T^{-2}\sum_{t=1}^{T}(T-t)\right). \tag{23}$$

By the law of large numbers, we have

$$T^{-1}\sum_{t=1}^{T}\varepsilon_t^* \Longrightarrow 0, \qquad T^{-1}\sum_{t=1}^{T}\varepsilon_{k,t}^* \Longrightarrow 0, \quad k=1,\ldots,q. \tag{24}$$

From the (iii) and (vii) of Lemma 3, we get

$$T^{-3/2}\sum_{t=1}^{T}t\varepsilon_t^* \Longrightarrow \sigma_{\varepsilon_t^*}W(1) - \sigma_{\varepsilon_t^*}\int_0^1 W(r)\,dr,$$
$$T^{-3/2}\sum_{t=1}^{T}t\varepsilon_{k,t}^* \Longrightarrow \sigma_{\varepsilon_{k,t}^*}W(1) - \sigma_{\varepsilon_{k,t}^*}\int_0^1 W(r)\,dr, \quad k=1,\ldots,q, \tag{25}$$
$$T^{-2}\sum_{t=1}^{T}(T-t) \longrightarrow \frac{1}{2}.$$

Then, we have

$$T^{-2}\sum_{t=1}^{T}X_{t-1} \Longrightarrow 0+0+0+\frac{1}{2}\mu_\varepsilon\left(1+\sum_{k=1}^{q}\theta_k\right) = \frac{1}{2}\mu_\varepsilon\left(1+\sum_{k=1}^{q}\theta_k\right). \tag{26}$$

(v) Elementary algebra gives us that

$$X_t = \sum_{k=1}^{t}u_k = \sum_{k=1}^{t}\left(\varepsilon_k+\theta_1\circ\varepsilon_{k-1}+\cdots+\theta_q\circ\varepsilon_{k-q}\right) = \sum_{k=1}^{t}\varepsilon_k^* + \sum_{k=1}^{t}\varepsilon_{1,k}^* + \cdots + \sum_{k=1}^{t}\varepsilon_{q,k}^* + \left(1+\sum_{k=1}^{q}\theta_k\right)\mu_\varepsilon t = \xi_t + \sum_{i=1}^{q}\eta_{i,t} + \left(1+\sum_{k=1}^{q}\theta_k\right)\mu_\varepsilon t, \tag{27}$$

where $\xi_t = \sum_{k=1}^{t}\varepsilon_k^*$, $\eta_{i,t} = \sum_{k=1}^{t}\varepsilon_{i,k}^*$, $i=1,\ldots,q$ and, as assumed, $\xi_0 = \eta_{1,0} = \cdots = \eta_{q,0} = 0$. It is easy to see that ξ_t and $\eta_{i,t}$, $i=1,\ldots,q$, follow a random walk process without drift. Using (v) and (vii) of Lemma 3, we get

$$T^{-5/2}\sum_{t=1}^{T}t\xi_{t-1} \Longrightarrow \sigma_{\varepsilon_t^*}\int_0^1 rW(r)\,dr,$$
$$T^{-5/2}\sum_{t=1}^{T}t\eta_{i,t-1} \Longrightarrow \sigma_{\varepsilon_{i,t}^*}\int_0^1 rW(r)\,dr, \quad i=1,\ldots,q, \tag{28}$$
$$T^{-3}\sum_{t=1}^{T}t(t-1) \longrightarrow \frac{1}{3}.$$

Then, we get

$$\begin{aligned} T^{-3}\sum_{t=1}^{T}tX_{t-1} &= T^{-3}\sum_{t=1}^{T}t\xi_{t-1} + \sum_{i=1}^{q}\left(T^{-3}\sum_{t=1}^{T}t\eta_{i,t-1}\right) + \left(1+\sum_{k=1}^{q}\theta_k\right)\mu_\varepsilon\left(T^{-3}\sum_{t=1}^{T}t(t-1)\right) \\ &= T^{-1/2}\left(T^{-5/2}\sum_{t=1}^{T}t\xi_{t-1}\right) + T^{-1/2}\sum_{i=1}^{q}\left(T^{-5/2}\sum_{t=1}^{T}t\eta_{i,t-1}\right) + \left(1+\sum_{k=1}^{q}\theta_k\right)\mu_\varepsilon\left(T^{-3}\sum_{t=1}^{T}t(t-1)\right) \\ &\Longrightarrow 0+0+\frac{1}{3}\left(1+\sum_{k=1}^{q}\theta_k\right)\mu_\varepsilon = \frac{1}{3}\left(1+\sum_{k=1}^{q}\theta_k\right)\mu_\varepsilon. \end{aligned} \tag{29}$$

(vi) One has

$$\begin{aligned} &T^{-3}\sum_{t=1}^{T}X_{t-1}^2 \\ &= T^{-3}\sum_{t=1}^{T}\left(\xi_{t-1}+\sum_{i=1}^{q}\eta_{i,t-1}+\left(1+\sum_{k=1}^{q}\theta_k\right)\mu_\varepsilon(t-1)\right)^2 \\ &= T^{-3}\sum_{t=1}^{T}\xi_{t-1}^2 + T^{-3}\sum_{i=1}^{q}\sum_{t=1}^{T}\eta_{i,t-1}^2 + \left(1+\sum_{k=1}^{q}\theta_k\right)^2\mu_\varepsilon^2\left(T^{-3}\sum_{t=1}^{T}(t-1)^2\right) \\ &\quad + 2\sum_{i=1}^{q}\left(T^{-3}\sum_{t=1}^{T}\xi_{t-1}\eta_{i,t-1}\right) \\ &\quad + 2\left(1+\sum_{k=1}^{q}\theta_k\right)\mu_\varepsilon\left(T^{-3}\sum_{t=1}^{T}\left(\xi_{t-1}+\sum_{i=1}^{q}\eta_{i,t-1}\right)(t-1)\right). \end{aligned} \tag{30}$$

Firstly, we prove $T^{-3}\sum_{t=1}^{T}(\xi_{t-1}+\sum_{i=1}^{q}\eta_{i,t-1})(t-1)\Rightarrow 0$. By (iii) and (v) of Lemma 3, we have

$$T^{-3/2}\sum_{t=1}^{T}t\varepsilon_t^*\Longrightarrow\sigma_{\varepsilon_t^*}W(1)-\sigma_{\varepsilon_t^*}\int_0^1 W(r)\,dr,$$
$$T^{-5/2}\sum_{t=1}^{T}t\xi_{t-1}\Longrightarrow\sigma_{\varepsilon_t^*}\int_0^1 rW(r)\,dr. \tag{31}$$

It is easy to see that $T^{-2}\xi_{T-1}\Rightarrow 0$ and $T^{-2}\varepsilon_T^*\Rightarrow 0$. Thus,

$$\begin{aligned}
&T^{-3}\sum_{t=1}^{T}(t-1)\,\xi_{t-1}\\
&\quad=T^{-3}\sum_{t=1}^{T-1}t\xi_t=T^{-3}\sum_{t=1}^{T-1}t\left(\xi_{t-1}+\varepsilon_t^*\right)\\
&\quad=T^{-3}\sum_{t=1}^{T}t\left(\xi_{t-1}+\varepsilon_t^*\right)-T^{-2}\xi_{T-1}-T^{-2}\varepsilon_T^*\\
&\quad=T^{-1/2}\left(T^{-5/2}\sum_{t=1}^{T-1}t\xi_{t-1}\right)+T^{-3/2}\left(T^{-3/2}\sum_{t=1}^{T-1}t\varepsilon_t^*\right)\\
&\qquad-T^{-2}\xi_{T-1}-T^{-2}\varepsilon_T^*\Longrightarrow 0.
\end{aligned}\tag{32}$$

By using a similar approach, we get $T^{-3}\sum_{t=1}^{T}(t-1)\eta_{i,t-1}\Rightarrow 0$, $i=1,\ldots,q$.

Therefore,

$$\begin{aligned}
&T^{-3}\sum_{t=1}^{T}\left(\xi_{t-1}+\sum_{i=1}^{q}\eta_{i,t-1}\right)(t-1)\\
&\quad=T^{-3}\sum_{t=1}^{T}(t-1)\,\xi_{t-1}\\
&\qquad+\sum_{i=1}^{q}\left(T^{-3}\sum_{t=1}^{T}(t-1)\,\eta_{i,t-1}\right)\Longrightarrow 0+0=0.
\end{aligned}\tag{33}$$

Secondly, we prove that the limit $T^{-3}\sum_{t=1}^{T}\xi_{t-1}\eta_{i,t-1}\Rightarrow 0$, $i=1,\ldots,q$, holds.

Using the well-known inequality $|\xi_{t-1}\eta_{i,t-1}|\le(1/2)(\xi_{t-1}^2+\eta_{i,t-1}^2)$, we find that

$$\begin{aligned}
\left|T^{-3}\sum_{t=1}^{T}\xi_{t-1}\eta_{i,t-1}\right|&\le T^{-3}\sum_{t=1}^{T}\left|\xi_{t-1}\eta_{i,t-1}\right|\\
&\le\frac{1}{2}\left(T^{-3}\sum_{t=1}^{T}\xi_{t-1}^2+T^{-3}\sum_{t=1}^{T}\eta_{i,t-1}^2\right).
\end{aligned}\tag{34}$$

From (vi) of Lemma 3, we get

$$T^{-2}\sum_{t=1}^{T}\xi_{t-1}^2\Longrightarrow\sigma_{\varepsilon_t^*}^2\int_0^1 W^2(r)\,dr,$$
$$T^{-2}\sum_{t=1}^{T}\eta_{i,t-1}^2\Longrightarrow\sigma_{\varepsilon_{i,t}^*}^2\int_0^1 W^2(r)\,dr. \tag{35}$$

The two limits imply that

$$T^{-3}\sum_{t=1}^{T}\xi_{t-1}^2\Longrightarrow 0,\qquad T^{-3}\sum_{t=1}^{T}\eta_{i,t-1}^2\Longrightarrow 0. \tag{36}$$

By the Cauchy-Schwarz theorem, we obtain $T^{-3}\sum_{t=1}^{T}\xi_{t-1}\eta_{i,t-1}\Rightarrow 0$, $i=1,\ldots,q$.

From (iii), (vi), and (vii) of Lemma 3, we obtain

$$T^{-2}\sum_{t=1}^{T}\xi_{t-1}^2\Longrightarrow\sigma_{\varepsilon_t^*}^2\int_0^1 W^2(r)\,dr,$$
$$T^{-2}\sum_{t=1}^{T}\eta_{i,t-1}^2\Longrightarrow\sigma_{\varepsilon_{i,t}^*}^2\int_0^1 W^2(r)\,dr,$$
$$T^{-3}\sum_{t=1}^{T}(t-1)^2\longrightarrow\frac{1}{3},$$
$$T^{-3/2}\sum_{t=1}^{T}t\varepsilon_t^*\Longrightarrow\sigma_{\varepsilon_t^*}W(1)-\sigma_{\varepsilon_t^*}\int_0^1 W(r)\,dr,$$
$$T^{-3/2}\sum_{t=1}^{T}t\varepsilon_{i,t}^*\Longrightarrow\sigma_{\varepsilon_{i,t}^*}W(1)-\sigma_{\varepsilon_{i,t}^*}\int_0^1 W(r)\,dr. \tag{37}$$

Therefore, $T^{-3}\sum_{t=1}^{T}X_{t-1}^2\Rightarrow 0+0+(1/3)(1+\sum_{k=1}^{q}\theta_k)^2\mu_\varepsilon^2+0+0=(1/3)(1+\sum_{k=1}^{q}\theta_k)^2\mu_\varepsilon^2$.

The proof of this theorem is complete. □

Theorem 6. *The conditional least squares estimators of α given by* (5) *converges in distribution to constant 1, when the true process is a nonstationary INAR*(1) *model with INMA*(q) *innovation.*

Proof. We first derive the numerator limit of (5):

$$\begin{aligned}
&T^{-4}\left(T\sum_{t=1}^{T}X_{t-1}X_t-\left(\sum_{t=1}^{T}X_{t-1}\right)\left(\sum_{t=1}^{T}X_t\right)\right)\\
&\quad=T^{-3}\sum_{t=1}^{T}X_{t-1}X_t-\left(T^{-2}\sum_{t=1}^{T}X_{t-1}\right)\left(T^{-2}\sum_{t=1}^{T}X_t\right)\\
&\quad=T^{-3}\sum_{t=1}^{T}X_{t-1}^2+T^{-1}\left(T^{-2}\sum_{t=1}^{T}X_{t-1}u_t\right)\\
&\qquad-\left(T^{-2}\sum_{t=1}^{T}X_{t-1}\right)\left(T^{-2}\sum_{t=1}^{T}X_t\right).
\end{aligned}\tag{38}$$

Table 1: Bias and MSE results of α for NSINARMA(1,1) model.

	CLS					
	$\lambda = 0.3$			$\lambda = 5$		
Sample size	100	300	800	100	300	800
$\theta_1 = 0.1$						
Bias(α)	$-2.3411e-03$	$-1.3253e-04$	$-2.4503e-05$	$-9.7709e-05$	$1.2879e-05$	$9.7185e-07$
MSE(α)	$1.6443e-03$	$5.2694e-06$	$1.8012e-07$	$2.8641e-06$	$4.9761e-08$	$2.8335e-10$
$\theta_1 = 0.4$						
Bias(α)	$-1.8837e-03$	$-6.2751e-05$	$-1.9073e-05$	$-9.3204e-05$	$1.1088e-05$	$2.1430e-06$
MSE(α)	$1.0645e-03$	$1.1813e-06$	$1.0913e-07$	$2.6061e-06$	$3.6884e-08$	$1.3778e-09$
$\theta_1 = 0.7$						
Bias(α)	$-1.3353e-03$	$-8.5133e-05$	$-1.3719e-05$	$-7.6709e-05$	$2.1019e-05$	$2.5092e-06$
MSE(α)	$5.3488e-04$	$2.1743e-06$	$5.6460e-08$	$1.7653e-06$	$1.3254e-07$	$1.8889e-09$
$\theta_1 = 0.9$						
Bias(α)	$-4.6077e-04$	$-6.1094e-05$	$6.5116e-06$	$-2.1260E-04$	$1.8841E-05$	$1.3932E-06$
MSE(α)	$6.3694e-05$	$1.1197e-06$	$1.2720e-08$	$1.3559E-05$	$1.0649E-07$	$5.8233E-10$

By the (ii), (iv), and (vi) of Theorem 5, we have

$$\begin{aligned}
T^{-2}\sum_{t=1}^{T} X_{t-1}u_t &\Longrightarrow \frac{1}{2}\mu_\varepsilon^2\left(1+\sum_{k=1}^{q}\theta_k\right)^2,\\
T^{-2}\sum_{t=1}^{T} X_{t-1} &\Longrightarrow \frac{1}{2}\mu_\varepsilon\left(1+\sum_{k=1}^{q}\theta_k\right),\\
T^{-2}\sum_{t=1}^{T} X_t = T^{-2}\sum_{t=1}^{T} X_{t-1} + T^{-1}\left(T^{-1}\sum_{t=1}^{T} u_t\right) &\qquad (39)\\
&\Longrightarrow \frac{1}{2}\mu_\varepsilon\left(1+\sum_{k=1}^{q}\theta_k\right),\\
T^{-3}\sum_{t=1}^{T} X_{t-1}^2 &\Longrightarrow \frac{1}{3}\mu_\varepsilon^2\left(1+\sum_{k=1}^{q}\theta_k\right)^2.
\end{aligned}$$

Then, we get

$$\begin{aligned}
&T^{-4}\left(T\sum_{t=1}^{T} X_{t-1}X_t - \left(\sum_{t=1}^{T} X_{t-1}\right)\left(\sum_{t=1}^{T} X_t\right)\right)\\
&\Longrightarrow \frac{1}{3}\mu_\varepsilon^2\left(1+\sum_{k=1}^{q}\theta_k\right)^2 + 0 - \left(\frac{1}{2}\mu_\varepsilon\left(1+\sum_{k=1}^{q}\theta_k\right)\right)^2\\
&= \frac{1}{12}\mu_\varepsilon^2\left(1+\sum_{k=1}^{q}\theta_k\right)^2. \qquad (40)
\end{aligned}$$

Similarly, we have the denominator limit of (5):

$$\begin{aligned}
&T^{-4}\left(T\sum_{t=1}^{T} X_{t-1}^2 - \left(\sum_{t=1}^{T} X_{t-1}\right)^2\right)\\
&\Longrightarrow \frac{1}{12}\mu_\varepsilon^2\left(1+\sum_{k=1}^{q}\theta_k\right)^2. \qquad (41)
\end{aligned}$$

Using the Slutsky theorem, we obtain

$$\begin{aligned}
\widehat{\alpha} &= \frac{T\sum_{t=1}^{T} X_{t-1}X_t - \left(\sum_{t=1}^{T} X_{t-1}\right)\left(\sum_{t=1}^{T} X_t\right)}{T\sum_{t=1}^{T} X_{t-1}^2 - \left(\sum_{t=1}^{T} X_{t-1}\right)^2}\\
&= \frac{T^{-4}\left(T\sum_{t=1}^{T} X_{t-1}X_t - \left(\sum_{t=1}^{T} X_{t-1}\right)\left(\sum_{t=1}^{T} X_t\right)\right)}{T^{-4}\left(T\sum_{t=1}^{T} X_{t-1}^2 - \left(\sum_{t=1}^{T} X_{t-1}\right)^2\right)} \qquad (42)\\
&\Longrightarrow \frac{(1/12)\,\mu_\varepsilon^2\left(1+\sum_{k=1}^{q}\theta_k\right)^2}{(1/12)\,\mu_\varepsilon^2\left(1+\sum_{k=1}^{q}\theta_k\right)^2} = 1.
\end{aligned}$$

This completes the proof. □

4. Simulation Study

To study the empirical performance of the CLS estimator of the autoregressive coefficient for an auxiliary regression process, while the true process is NSINARMA$(1, q)$ process, a brief simulation study is conducted.

Consider the true process,

$$\begin{aligned}
X_t &= X_{t-1} + u_t,\\
u_t &= \varepsilon_t + \theta_1 \circ \varepsilon_{t-1} + \cdots + \theta_q \circ \varepsilon_{t-q}, \qquad (43)
\end{aligned}$$

Table 2: Bias and MSE results of α for NSINARMA(1,2) model.

	CLS					
	$\lambda = 0.3$			$\lambda = 5$		
Sample size	100	300	800	100	300	800
$(\theta_1, \theta_2) = (0.1, 0.1)$						
Bias(α)	$-9.8012e-04$	$-8.0314e-05$	$-1.2149e-05$	$1.6030e-05$	$-2.6319e-05$	$-7.9892e-06$
MSE(α)	$2.8819e-04$	$1.9351e-06$	$4.4280e-08$	$7.7085e-07$	$2.0781e-07$	$1.9148e-08$
$(\theta_1, \theta_2) = (0.1, 0.6)$						
Bias(α)	$-9.0876e-04$	$-1.4515e-04$	$3.6936e-06$	$-4.5047e-05$	$-3.6633e-05$	$-1.7588e-06$
MSE(α)	$2.4775e-04$	$6.3203e-06$	$4.0928e-09$	$6.0877e-07$	$4.0259e-07$	$9.2803e-10$
$(\theta_1, \theta_2) = (0.2, 0.3)$						
Bias(α)	$-5.8582e-04$	$-8.2571e-05$	$-1.1720e-05$	$4.7792e-05$	$-2.2042e-05$	$-7.8901e-06$
MSE(α)	$1.0296e-04$	$2.0454e-06$	$4.1208e-08$	$6.8522e-07$	$1.4575e-07$	$1.8676e-08$
$(\theta_1, \theta_2) = (0.3, 0.4)$						
Bias(α)	$-1.0548e-03$	$-1.3550e-04$	$-2.1872e-06$	$-6.0986e-05$	$-2.1873e-05$	$-1.6658e-08$
MSE(α)	$3.3379e-04$	$5.5082e-06$	$1.4352e-09$	$1.1158e-06$	$1.4353e-07$	$8.3243e-14$
$(\theta_1, \theta_2) = (0.4, 0.4)$						
Bias(α)	$-8.9921e-04$	$-1.4099e-04$	$-8.5553e-07$	$-6.6568e-05$	$-2.2816e-05$	$1.0669e-07$
MSE(α)	$2.4257e-04$	$5.9633e-06$	$2.1958e-10$	$1.3294e-06$	$1.5617e-07$	$3.4148e-12$

Table 3: Bias and MSE results of α for NSINARMA(1,3) model.

	CLS					
	$\lambda = 0.3$			$\lambda = 5$		
Sample size	100	300	800	100	300	800
$(\theta_1, \theta_2, \theta_3) = (0.1, 0.1, 0.1)$						
Bias(α)	$-1.3742e-03$	$-1.4097e-04$	$-3.3819e-05$	$-4.4965e-05$	$4.5510e-06$	$-5.1844e-06$
MSE(α)	$5.6651e-04$	$5.9616e-06$	$3.4312e-07$	$6.0655e-07$	$6.2135e-08$	$8.0635e-09$
$(\theta_1, \theta_2, \theta_3) = (0.1, 0.2, 0.4)$						
Bias(α)	$-2.0218e-03$	$-9.3641e-05$	$8.2901e-06$	$-1.4846e-04$	$-8.0926e-06$	$-8.4087e-07$
MSE(α)	$1.2263e-03$	$2.6306e-06$	$2.0618e-08$	$6.6121e-06$	$1.9647e-08$	$2.1212e-10$
$(\theta_1, \theta_2, \theta_3) = (0.2, 0.1, 0.2)$						
Bias(α)	$-1.0825e-03$	$-1.1218e-04$	$-3.0775e-05$	$-2.0846e-05$	$7.8758e-06$	$-4.3565e-06$
MSE(α)	$3.5154e-04$	$3.7751e-06$	$2.8413e-07$	$1.3037e-07$	$1.8608e-08$	$5.6936e-09$
$(\theta_1, \theta_2, \theta_3) = (0.3, 0.3, 0.3)$						
Bias(α)	$-1.6766e-03$	$-1.3506e-04$	$8.8713e-06$	$-1.3902e-04$	$-2.6374e-06$	$-2.3441e-07$
MSE(α)	$8.4330e-04$	$5.4723e-06$	$2.3610e-08$	$5.7980e-06$	$2.0868e-09$	$1.6485e-11$
$(\theta_1, \theta_2, \theta_3) = (0.5, 0.3, 0.1)$						
Bias(α)	$-1.5676e-03$	$-1.3084e-04$	$7.6103e-06$	$-1.5146e-04$	$-2.5173e-06$	$-2.9431e-07$
MSE(α)	$7.3722e-04$	$5.1355e-06$	$1.7375e-08$	$6.8819e-06$	$1.9010e-09$	$2.5985e-11$

where $\{\varepsilon_t\}$ is an an i.i.d. sequence of the Poisson random variables with parameter $\lambda = 0.3, 5$. The lag orders and coefficient parameters values considered are

(i) for $q = 1$, $\theta_1 \in \{0.1, 0.4, 0.7, 0.9\}$,

(ii) for $q = 2$, $(\theta_1, \theta_2) \in \{(0.1, 0.1), (0.2, 0.3), (0.1, 0.6), (0.3, 0.4), (0.4, 0.4)\}$,

(iii) for $q = 3$, $(\theta_1, \theta_2, \theta_3) \in \{(0.1, 0.1, 0.1), (0.1, 0.2, 0.4), (0.2, 0.1, 0.2), (0.3, 0.3, 0.3), (0.5, 0.3, 0.1)\}$.

The auxiliary regression process is an INAR(1) process,

$$X_t = \alpha \circ X_{t-1} + v_t, \tag{44}$$

where $\{v_t\}$ is a sequence of i.i.d. nonnegative integer-valued random variables. This simulation study is conducted to indicate the large sample performances of the CLS estimator of α when the true process is NSINARMA$(1, q)$ process. In the simulation, we use $X_0 = \varepsilon_0 = \varepsilon_{-1} = \cdots = \varepsilon_{-q} = 0$. The study is based on 300 replications. For each replication, we estimate the model parameter α and calculate the bias and MSE of the parameter estimates. The sample size is varied to be $T = 100$, 300, and 800.

From the results reported in Tables 1, 2, and 3, we can see that CLS is a good estimation method. The estimates' bias and MSE values are all small. Most of the biases are negative.

All the bias and MSE values decrease with an increasing sample size T. When the coefficient θ of innovation process and the sample size are fixed, we find that the absolute values of bias and MSE become smaller with an increasing λ. When the sample size is increased, the MSE and bias values both converge to zero. For example, the smallest bias and MSE values in the simulation showed in Table 1 are $9.7185e-07$ and $2.8335e-10$. This illustrates that the CLS estimator $\widehat{\alpha}$ given by (5) converges to a constant.

5. Conclusions

In this paper, we have proposed a nonstationary INAR model for nonstationary integer-valued data. We have used an extended structure of the innovation process to allow the innovation with correlation to follow an INMA(q) process. We have presented the moments and conditional moments for this model, proposed a CLS estimator for the autoregressive coefficient in auxiliary regression model, and obtained the asymptotic distribution for the estimator of coefficient. The simulation results indicate that our CLS method produces good estimates for large samples.

Acknowledgments

The authors are grateful to two anonymous referees for helpful comments and suggestions which greatly improved the paper. This work is supported by the National Natural Science Foundation of China (no. 71171166 and no. 71201126), the Science Foundation of Ministry of Education of China (no. 12XJC910001), and the Fundamental Research Funds for the Central Universities (no. JBK120405).

References

[1] N. Silva, I. Pereira, and M. E. Silva, "Forecasting in *INAR*(1) model," *REVSTAT Statistical Journal*, vol. 7, no. 1, pp. 119–134, 2009.

[2] C. H. Weiß, "Thinning operations for modeling time series of counts—a survey," *Advances in Statistical Analysis*, vol. 92, no. 3, pp. 319–341, 2008.

[3] R. Ferland, A. Latour, and D. Oraichi, "Integer-valued GARCH process," *Journal of Time Series Analysis*, vol. 27, no. 6, pp. 923–942, 2006.

[4] K. Fokianos and R. Fried, "Interventions in INGARCH processes," *Journal of Time Series Analysis*, vol. 31, no. 3, pp. 210–225, 2010.

[5] C. H. Weiß, "The INARCH(1) model for overdispersed time series of counts," *Communications in Statistics—Simulation and Computation*, vol. 39, no. 6, pp. 1269–1291, 2010.

[6] F. Zhu, "A negative binomial integer-valued GARCH model," *Journal of Time Series Analysis*, vol. 32, no. 1, pp. 54–67, 2011.

[7] F. Zhu, "Modeling overdispersed or underdispersed count data with generalized Poisson integer-valued GARCH models," *Journal of Mathematical Analysis and Applications*, vol. 389, no. 1, pp. 58–71, 2012.

[8] F. Zhu and D. Wang, "Diagnostic checking integer-valued *ARCH*(p) models using conditional residual autocorrelations," *Computational Statistics & Data Analysis*, vol. 54, no. 2, pp. 496–508, 2010.

[9] R. Bu and B. McCabe, "Model selection, estimation and forecasting in *INAR*(p) models: a likelihood-based Markov Chain approach," *International Journal of Forecasting*, vol. 24, no. 1, pp. 151–162, 2008.

[10] V. Enciso-Mora, P. Neal, and T. Subba Rao, "Efficient order selection algorithms for integer-valued ARMA processes," *Journal of Time Series Analysis*, vol. 30, no. 1, pp. 1–18, 2009.

[11] B. P. M. McCabe, G. M. Martin, and D. Harris, "Efficient probabilistic forecasts for counts," *Journal of the Royal Statistical Society B*, vol. 73, no. 2, pp. 253–272, 2011.

[12] H. Zheng, I. V. Basawa, and S. Datta, "Inference for pth-order random coefficient integer-valued autoregressive processes," *Journal of Time Series Analysis*, vol. 27, no. 3, pp. 411–440, 2006.

[13] H. Zheng, I. V. Basawa, and S. Datta, "First-order random coefficient integer-valued autoregressive processes," *Journal of Statistical Planning and Inference*, vol. 137, no. 1, pp. 212–229, 2007.

[14] D. Gomes and L. C. e Castro, "Generalized integer-valued random coefficient for a first order structure autoregressive (RCINAR) process," *Journal of Statistical Planning and Inference*, vol. 139, no. 12, pp. 4088–4097, 2009.

[15] K. Yu, D. Shi, and P. Song, "First-order random coefficient integer-valued moving average process," *Journal of Zhejiang University—Science Edition*, vol. 37, no. 2, pp. 153–159, 2010.

[16] K. Yu, D. Shi, and H. Zou, "The random coefficient discrete-valued time series model," *Statistical Research*, vol. 28, no. 4, pp. 106–112, 2011 (Chinese).

[17] M. Kachour and J. F. Yao, "First-order rounded integer-valued autoregressive (*RINAR*(1)) process," *Journal of Time Series Analysis*, vol. 30, no. 4, pp. 417–448, 2009.

[18] H.-Y. Kim and Y. Park, "A non-stationary integer-valued autoregressive model," *Statistical Papers*, vol. 49, no. 3, pp. 485–502, 2008.

[19] H. Zhang, D. Wang, and F. Zhu, "Inference for *INAR*(p) processes with signed generalized power series thinning operator," *Journal of Statistical Planning and Inference*, vol. 140, no. 3, pp. 667–683, 2010.

[20] M. Kachour and L. Truquet, "A p-order signed integer-valued autoregressive (*SINAR*(p)) model," *Journal of Time Series Analysis*, vol. 32, no. 3, pp. 223–236, 2011.

[21] J. Hellström, "Unit root testing in integer-valued AR(1) models," *Economics Letters*, vol. 70, no. 1, pp. 9–14, 2001.

[22] M. Ispány, G. Pap, and M. C. A. van Zuijlen, "Asymptotic inference for nearly unstable *INAR*(1) models," *Journal of Applied Probability*, vol. 40, no. 3, pp. 750–765, 2003.

[23] L. Györfi, M. Ispány, G. Pap, and K. Varga, "Poisson limit of an inhomogeneous nearly critical *INAR*(1) model," *Acta Universitatis Szegediensis*, vol. 73, no. 3-4, pp. 789–815, 2007.

[24] F. C. Drost, R. van den Akker, and B. J. M. Werker, "The asymptotic structure of nearly unstable non-negative integer-valued AR(1) models," *Bernoulli*, vol. 15, no. 2, pp. 297–324, 2009.

[25] M. Barczy, M. Ispány, and G. Pap, "Asymptotic behavior of unstable *INAR*(p) processes," *Stochastic Processes and their Applications*, vol. 121, no. 3, pp. 583–608, 2011.

[26] J. D. Hamilton, *Time Series Analysis*, Princeton University Press, Princeton, NJ, USA, 1994.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 183410, 7 pages
http://dx.doi.org/10.1155/2013/183410

Research Article

A Dynamic Fuzzy Cluster Algorithm for Time Series

Min Ji,[1] Fuding Xie,[2] and Yu Ping[3]

[1] *School of Computer Science, Liaoning Normal University, Dalian, Liaoning 116081, China*
[2] *School of Urban and Environmental Science, Liaoning Normal University, Dalian, Liaoning 116029, China*
[3] *The School of Electronics and Information Engineering, Tongji University, Shanghai 201804, China*

Correspondence should be addressed to Fuding Xie; xiefd@lnnu.edu.cn

Received 19 December 2012; Accepted 25 March 2013

Academic Editor: Jianhong (Cecilia) Xia

This paper presents an efficient algorithm, called dynamic fuzzy cluster (DFC), for dynamically clustering time series by introducing the definition of key point and improving FCM algorithm. The proposed algorithm works by determining those time series whose class labels are vague and further partitions them into different clusters over time. The main advantage of this approach compared with other existing algorithms is that the property of some time series belonging to different clusters over time can be partially revealed. Results from simulation-based experiments on geographical data demonstrate the excellent performance and the desired results have been obtained. The proposed algorithm can be applied to solve other clustering problems in data mining.

1. Introduction

Time series clustering problems arise when we observe a sample of time series data and want to group them into different categories or clusters. This is an important area of research for different disciplines because time series is a very popular type of data which exists in many domains, such as environmental monitoring, market research, and quality control. It is well known that the goal of time series clustering is to discover the natural grouping(s) of a set of patterns. An operational definition of time series clustering can be stated as follows. Given a representation of n series, find k groups based on a measure of similarity such that the similarities between objects in the same group are high while the similarities between objects in different groups are low.

Traditionally, the clustering methods are divided into two parts: crisp clustering and fuzzy clustering. Generally, each sample will belong to one cluster or the other by the crisp clustering methods, such as k-means and spectral method. Instead of assigning each sample a cluster label, fuzzy partition methods allow each sample to have different membership degrees.

However, time series often display dynamic behavior in their evolution over time. From Figure 1, one can see that during a certain period, a time series might belong to a certain cluster; afterwards its dynamics might be closer to that of another cluster. This switch from one time state to another is a typically dynamic behavior of time series over time. Thus, this dynamic behavior should be taken into account when attempting to cluster time series. In this case, the traditional clustering approaches are unlikely to locate and effectively represent the underlying structure for the given time series. D'Urso and Maharaj [1, 2] pointed out the existence of switching time series and studied it by autocorrelation-based and wavelets-based methods, respectively. That is to say that, the cluster labels of switching series are varied over time. Therefore, it is worthwhile to further investigate how the cluster of the switching series is changed over time. Motivated by their work, our proposal investigates the problem of evolutionary clustering and proposes a dynamic fuzzy cluster algorithm based on improved FCM algorithm and key points. Some properties of switching time series is further detected over time.

The rest of the paper is organized as follows. Related works are reviewed in the next section. In Section 3, we introduce the definition of key point, improve the FCM algorithm, and conclude by proposing a dynamic fuzzy cluster algorithm. In Section 4, we provide experimental results to validate the proposal, and finally in Section 5 we give discussion and conclusions.

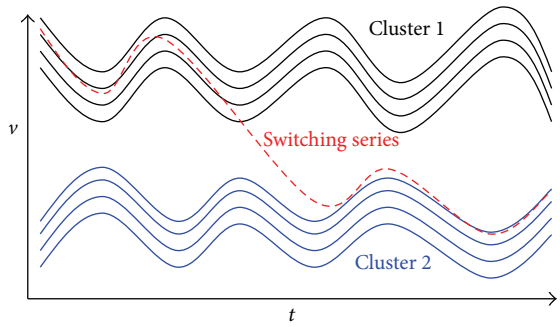


Figure 1: The dynamic behavior of switching time series over time.

2. Related Works

There are some previous references in the literature that have considered the problem of clustering time series. In 2005, Liao [3] presented a survey of clustering of time series data. He introduced and summarized previous works that investigated the clustering of time series data in various application domains, including general-purpose clustering algorithms, the criteria for evaluating the performance of the clustering results, and the measures to determine the similarity/dissimilarity between two time series being compared. Chakrabarti et al. [4] first presented a generic framework for evolutionary clustering and discussed evolutionary versions of two widely used clustering algorithms within this framework: k-means and agglomerative hierarchical clustering. To fulfill evolutionary clustering, a measure of temporal smoothness was integrated in the overall measure of clustering quality and two frameworks that incorporated temporal smoothness in evolutionary spectral clustering were also proposed [5]. Corduas and Piccolo investigated time series clustering and classification by the autoregressive metric [6]. Xiong and Yeung studied the clustering of data patterns that are represented as sequences or time series possibly of different lengths by using mixtures of autoregressive moving average (ARMA) models [7]. Without assuming any parametric model for the true autoregressive structure of the series, a general class of nonparametric autoregressive models was studied [8]. Combined the tools of symbolic time series analysis with the nearest neighbor single linkage clustering algorithm, Brida et al. [9] introduced a new method to describe dynamic patterns of the real exchange rate comovements time series and to analyze their influence in currency crises. By using the principle of complex network, a novel algorithm for shape-based time series clustering was proposed by Zhang et al. [10]. It can reduce the size of data and improve the efficiency of the algorithm. An efficient pattern reduction algorithm for reducing the computation time of k-means and k-means-based clustering algorithms was proposed and applied to cluster time series in [11]. E. Keogh [12–15] and his panel do a lot of work on time series classification and clustering and provide many useful datasets and benchmarks for testing time series classification and clustering. Furthermore, they also declared that clustering of time series subsequences, extracted via a sliding window, is meaningless. The latest development of cluster time series refers to Fu's work [16].

Recently, Jain [17] undertook a review of the 50-year existence of K-means algorithm and pointed out some of the emerging and useful research directions, including semisupervised clustering, ensemble clustering, simultaneous feature selection during data clustering, and large-scale data clustering. Fuzzy c-means, proposed by Dunn [18] and later improved by many authors, is an extension of K-means, where each data point can be a member of multiple clusters with a membership value. By modifying the FCM algorithm, Höppner and Klawonn [19] proposed a cross-correlation clustering (CCC) algorithm to solve the problem of clustering unaligned time series. It can be applied not only to short time series (whole series clustering) but also to time series subsequence (STS) clustering. More works related to fuzzy clustering can be found in [20, 21]. Data reduction by replacing group examples with their centroids before clustering them was used to speed up K-means and fuzzy c-means [22]. To incorporate the fuzziness in the clustering procedure, the so-called membership degree of each time series to different groups is considered as a means of evaluating the fuzziness in the assignment procedure. Möller-Levet et al. [23] introduced a new algorithm in the fuzzy c-means family, which is designed to cluster time series and is particularly suited for short time series and those with unevenly spaced sampling points. Considering the dynamic behavior of time series, D'Urso and Maharaj proposed a fuzzy clustering approach based on the autocorrelation functions of time series, in which each time series is not assigned exclusively to only one cluster, but it is allowed to belong to different clusters with various membership degrees [1]. In the evaluation of the time series of labels, fuzzy c-means clustering method is performed on merged dataset and time series of labels of each dataset are derived [24]. In order to deal with the more complicated data, Kannan et al. [25] proposed an alternative generalization of FCM clustering techniques called quadratic entropy based fuzzy c-means.

Various applications of time series have also been investigated in geography. Since voluminous time series have been, and continue to be, collected with modern data acquisition techniques, there is an urgent need for effective and efficient methods to extract unknown and unexpected information from spatial datasets of unprecedentedly large size, high dimensionality, and complexity. To address these challenges, spatial data mining and geographic knowledge discovery have emerged as an active research field [26]. Obviously, it is a basic and important problem to cluster spatial data in geography. Some published examples of cluster analysis in time series have been based on environmental data, where we have time series from different locations and wish to group locations which show similar behavior. See, for instance, Macchiato et al. [27] for a spatial clustering of daily ambient temperature, or Cowpertwait and Cox [28] for an application to a rainfall problem. Other examples can be found in medicine, economics, engineering, and so forth. A method for clustering multidimensional nonstationary meteorological time series was presented by Horenko [29]. The approach was based on optimization of the regularized

averaged clustering functional describing the quality of data representation in terms of several regression models and a metastable hidden process switching between them. Wang and Chen [30] presented a new method to predict the temperature and the Taiwan Futures Exchange, based on automatic clustering techniques and two-factors high-order fuzzy time series. Change-point analysis was used to detect changes in variability within GOMOS hindcast timeseries for significant wave heights of storm peak events across the Gulf of Mexico for the period 1900–2005. The change-point procedure can be readily applied to other environmental timeseries [21]. They presented a statistical approach based on the k-means clustering technique to manage environmental sampled data to evaluate and forecast the energy deliverable by different renewable sources in a given site. Clustering of industrialized countries according to historical data of CO_2 emissions was investigated by Alonso et al. [31]. Other examples can be found in medicine, economics, engineering, and so forth.

3. Dynamic Fuzzy Cluster Algorithm

A detailed description of the proposed algorithm is presented in this section.

3.1. Key Point. Mathematically, time series is defined as a set of observations x_i, each one being recorded at a specified time t_i. Generally, we can denote a n-dimensional time series as follows:

$$\mathrm{TS} = \left\{\left(x_1, t_1\right), \left(x_2, t_2\right), \ldots, \left(x_n, t_n\right)\right\}. \tag{1}$$

If all the time intervals $t_i - t_{i-1}$ are equal, that is, $t_i - t_{i-1} = \Delta t$ $(i = 2, 3, \ldots, n)$, then TS can be simply written as $\{x_1, x_2, \ldots, x_n\}$.

Definition 1. A point (x_i, t_i) is a change point of time series TS, if it satisfies the following conditions:

$$\cos(\theta) = \frac{\vec{\alpha} \cdot \vec{\beta}}{\|\alpha\| \cdot \|\beta\|} < 0 \tag{2}$$

or

$$\cos(\theta) = \frac{\vec{\alpha} \cdot \vec{\beta}}{\|\alpha\| \cdot \|\beta\|} \leq \tau, \quad \text{if } \left(x_i - x_{i-1}\right)\left(x_{i+1} - x_1\right) > 0, \tag{3}$$

where $\vec{\alpha} = (x_i - x_{i-1}, t_i - t_{i-1})$, $\vec{\beta} = (x_{i+1} - x_i, t_{i+1} - t_i)$, and τ is a parameter.

Definition 2. A change point (x_i, t_i) is called key point of time series TS if the following condition is satisfied:

$$\left|t_i - t_j\right| \geq \gamma \Delta t, \tag{4}$$

where γ is a parameter, and (x_j, t_j) is a neighbor change point of (x_i, t_i).

For convenience, we set (x_1, t_1) and (x_n, t_n) to be key points of time series TS and denote the set of key point of time series TS by $\{\mathrm{KP}_1, \mathrm{KP}_2, \ldots, \mathrm{KP}_s\}$.

Obviously, the key point is a typical point that describes, implicitly, how a series changes in a certain time. These points usually represent the special moments, such as the start or the end in a tendency of upward or downward, the peak or the bottom of the series. The key point clearly reveals the dynamic aspect of a given time series. Thus much more attention should be paid to them. The other advantage of key point is that it can effectively avoid the impact of singular points for clustering result.

3.2. Improved FCM Algorithm. In this subsection, we suggest a fuzzy clustering model for classifying time series. Fuzzy clustering is an overlapping clustering method which allows cases to belong to more than one cluster simultaneously as opposed to traditional clustering which results in mutually exclusive clusters. In the proposed fuzzy clustering model, we take into account the fuzziness and the key point. Our clustering model incorporates the information on the key point to the time series which obviously contains much more information than ordinary points in a series, and incorporates fuzziness which represents the uncertainty associated with the assignment of time series to different clusters. In what follows, it is apparent that in order to incorporate fuzziness into the clustering process, the so-called membership degree of each time series to each cluster should be considered.

The term of weighted-matrix should be introduced firstly before we start to describe the modified FCM model. The weighted-matrix is defined as

$$W = \begin{pmatrix} w_{11}, w_{12}, \ldots, w_{1n} \\ w_{21}, w_{22}, \ldots, w_{2n} \\ \vdots \\ w_{M1}, w_{12}, \ldots, w_{Mn} \end{pmatrix}, \tag{5}$$

where w_{ij} indicates the weight of ith time series at jth observation and refers to following criteria:

$$w_{ij} = \begin{cases} 1, & \text{if } \left(x_j, t_j\right) \text{ is not a key point of } i\text{th time series,} \\ 9 * \tanh\left|x_{ij} - \dfrac{x_{ij-1} + x_{ij+1}}{2}\right| + 1, & \text{if } \left(x_j, t_j\right) \text{ is a key point of } i\text{th time series and } j \neq 1,\ j \neq n, \\ 5, & \text{if } j = 1, \\ 10, & \text{if } j = n. \end{cases} \tag{6}$$

The importance of each key point is described initially by the nonnegative difference between the value of key point and the average of two adjacent points (ordinary points) of key point. Logically, the larger difference represents the more important status of key point. We map w_{ij} into 1 to 10 levels to understand and compute the importance of key point by function $\tanh(x) = (e^x - e^{-x})/(e^x + e^{-x})$ which is monotonically increasing function with range from −1 to 1. Doing so, we can easily measure the importance of each key point in terms of the value in the interval [1, 10]. Consider the fact that, in most cases, start points ($j = 1$) are less important than the end points ($j = n$) which represent the last status of a time series, and they are assigned the average value 5 and the maximal value 10, respectively.

The modified fuzzy clustering model based on the key point can be formalized as follows:

$$\text{min: } J_m = \sum_{i=1}^{M}\sum_{c=1}^{C} u_{ic}^m d_{ic}^2 = \sum_{i=1}^{M}\sum_{c=1}^{C} u_{ic}^m \sum_{j=1}^{n}\left[w_{ij}\left(x_{ij} - x_{cj}\right)\right]^2 \tag{7}$$

$$\text{with the constraints: } \sum_{c=1}^{C} u_{ic} = 1, \quad u_{ic} \geq 0,$$

where u_{ic} is the membership degree of the ith time series to the cth cluster, $d_{ic}^2 = \sum_{j=1}^{n}[w_{ij}(x_{ij} - x_{cj})]^2$ is the squared Euclidean distance measure between the ith time series and the centroid time series of the cth cluster based on weighted w_{ij}, and $m > 1$ is a parameter that controls the fuzziness of the partition. Usually $m \in [2, 2.5]$, hence we take $m = 2$ in our later experiments.

Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, with the update of membership degree u_{ic} and the cluster centers TS_c. This procedure converges to a local minimum or a saddle point of J_m.

The updating steps are defined as

$$u_{ic} = \left[\sum_{c'=1}^{C}\left(\frac{\sum_{j=1}^{n}\left[w_{ij}\left(x_{ij} - x_{cj}\right)\right]^2}{\sum_{j=1}^{n}\left[w_{ij}\left(x_{ij} - x_{c'j}\right)\right]^2}\right)^{1/(m-1)}\right]^{-1}, \tag{8}$$

$$\mathrm{TS}_c = \frac{\sum_{i=1}^{M} u_{ic}^m \mathrm{TS}_i}{\sum_{i=1}^{M} u_{ic}^m}.$$

3.3. The Switching Time Series. Owing to the reason that the switching time series may belong to the different clusters over time, their dynamic property cannot be revealed sufficiently if we simply point out that they belong to a cluster with probability u_{ic}. Instead, we should further determine their cluster in a certain time period.

If there exists an $u_{ik} \geq 0.7$, then we can say that TS_i is stable and belongs to kth cluster. If $u_{ik} \leq 0.2$, it is known that TS_i should not belong to kth cluster. When $0.2 < \max\{u_{ik}\} < 0.7$ ($1 \leq k \leq C$), it is necessary to further judge the relation between TS_i and kth cluster. That is to say, TS_i is to be a switch series.

It is easy to find these switch series by modified FCM algorithm. Suppose that TS_i is a switch series with membership degrees $(u_{i1}, u_{i2}, \ldots, u_{iC})$ and the set of maintaining key point $\{\mathrm{KP}_1, \mathrm{KP}_2, \ldots, \mathrm{KP}_s\}$. We compute the distance TS_i and the centroid time series of the kth ($0.2 < u_{ik} < 0.7$) cluster in interval $[\mathrm{KP}_h, \mathrm{KP}_{h+1}]$ by $D_{ik} = \sum(x_{ij} - x_{kj})^2$. If $D_{ij} = \min\{D_{ig}\}$, the series TS_i is assigned cluster label j.

3.4. The Description of Dynamic Fuzzy Cluster Algorithm. Given a set of TS, the set of the key points for each TS_i is first determined by using Definitions 1 and 2. Considering the importance of key point, we improve the FCM algorithm by proposing a novel measurement of dissimilarity. The traditional distances in the FCM model would be replaced by the suggested dissimilarity measurement, leading to an enhancement of the FCM model.

The dynamic fuzzy cluster algorithm is presented as follows.

Input: the set of time series (TS).

Output: the result of cluster and the switching time series.

Step 1: For each TS_i, compute the set of change points by Definition 1 and the set of the maintaining key points by Definition 2.

Step 2: To calculate the weighted-matrix W by formulating W_{ij}.

Step 3: Cluster time series (TS) by the modified FCM algorithm.

Step 4: If there exist the switching time series, further determine their cluster in a certain time period in terms of principle stated in Section 3.3, otherwise, output the cluster results.

4. Experiment

To test the validity of our proposal, we conducted experiments using real dataset and simulation studies. The reason that we choose the daily temperature data of three states in America lies in the following: (1) There exists distinct differences among its average temperature because these three states are far from each other in geography. (2) The temperature data recorded by these stations is relatively complete. (3) It is easy to understand the cluster result. Obviously, the temperature series recorded by the stations which located in the same state should group in the same cluster. The clustering results obtained here exactly show this fact. The second dataset called Beef was created by Keogh and is usually employed to test the result of classification or clustering for time series. To some extent, our results may explain the reason that some series are misclassified or misclustered when testing this dataset.

Example A (daily temperature data of the United States). The daily temperature data (http://www.ncdc.noaa.gov/IPS/coop/coop.html) from various stations of New Mexico (NM), Montana (MO), Hawaii (HA), Wyoming (WY), and South

Table 1: The cluster result of 24 stable time series.

Cluster 1 (HA)	Cluster 2 (NM)	Cluster 3 (MO)
Honolulu Observ 702.2	Artesia	Chinook
Kaohe Makai 24.4	Caprock	Dunkirk 19NNE
Kapaluaw Maui AP462	Clovios	Judith Gap 13E
Kilauea 1134	EL Morronatl Mon	Hot Springs
Kualoa RCH HQ 886.9	Hobbs	Libby Dam Base
Kaneohe 838.1	Rio Rancho #2	Olney
Makaha Ctry Club 800.3	Roswell Climat	Polebridge
Poamoho EXP FM 855.2	State Univ	Round Butte 1 NNW

Dakota (SD) in the United States from 2010/09 to 2010/11 has been collected for our experiments. It had temperature recordings from 8 stations in New Mexico, 8 stations in Hawaii, 8 stations in Montana, and 4 stations in South Dakota and Wyoming located between the Montana and New Mexico in latitude. We clustered these 28 time series using the proposed algorithm. In our experiment we considered the thresholds, $\tau = 0.3$, $c = 3$, and $\gamma = 5$. As a result, the stable time series and switching time series are obtained. The stable time series is composed of three clusters, which correspond exactly to the real case. The temperature data recorded by the stations located in the same state are grouped successfully in the same cluster; see Table 1 (the entries in the column are the name of the station where the daily temperature data is recorded). In our result, the four switching time series are obtained. From Figure 2, it is easy to see the dynamic behavior of switching time series over time. The Colony station is located in the south of Wyoming state and this state is between Montana state and New Mexico state. The temperature recorded at Colony station is similar to the average temperature in New Mexico state since they are all inland areas and have some common character in the geography. The meteorological information shows that the temperature in Colony region suddenly decreased in the middle of September and November because of the arrival of the cold air from the north. This result is reflected accurately in our figure. This phenomenon can also be observed in the other three stations located in South Dakota. The latter indicates the arrival of the winter in the north of America. Different from the other three stations, to be the same as the Ft Meade station, the Bison station is located in the south of the South Dakota state. They belong to the low basin topography. Meanwhile, this region is usually cloudy and the ground antiradiation is intensive. These reasons suggest that the climate in this region is mutable. Thus its temperature is bounced among three clusters.

Example B (test on a dataset). The Beef dataset created by Keogh et al. [15] is usually employed to test the result of classification or clustering for time series. There are 30 time series in this dataset. They are classified into 5 groups, and the length of each of series is 470. Here, we choose its testing set to show the validity of the proposed algorithm. There are 7

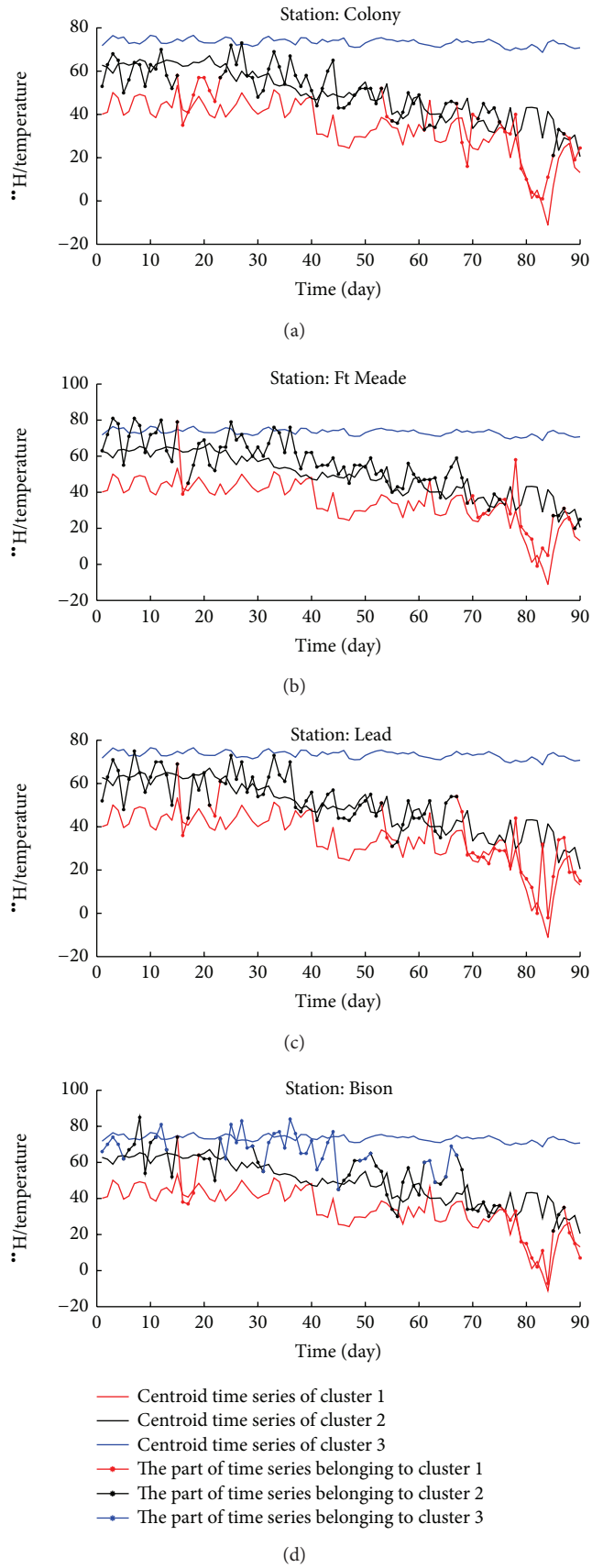


Figure 2: The dynamic behavior of the four switching time series over time.

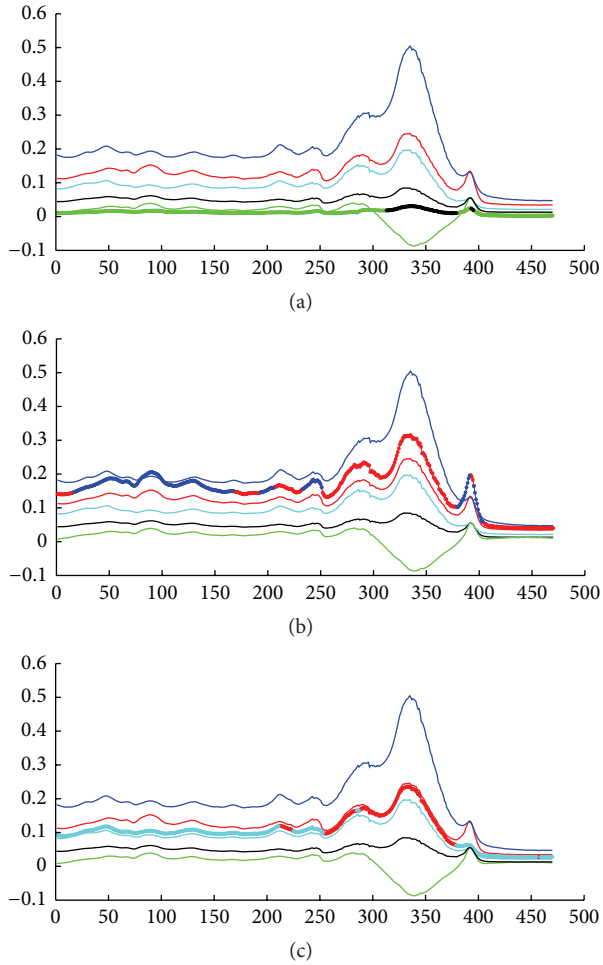


FIGURE 3: The dynamic behavior of the 6th, 7th and 18th switching time series from top to bottom orderly. The five slender lines represent five centroid series of five clusters, respectively.

switching time series in this dataset, that is, 4th, 6th, 7th, 10th, 14th, 18th, and 28th. Their membership matrix is as follows:

$$\begin{pmatrix} 0.0994 & 0.4310 & 0.0337 & 0.1655 & 0.2704 \\ 0.0570 & 0.3011 & 0.0239 & 0.0952 & 0.5227 \\ 0.4400 & 0.0854 & 0.2532 & 0.1716 & 0.0498 \\ 0.2605 & 0.1965 & 0.0417 & 0.4328 & 0.0685 \\ 0.0925 & 0.4876 & 0.0237 & 0.3223 & 0.0739 \\ 0.4011 & 0.0698 & 0.0322 & 0.4687 & 0.0282 \\ 0.1325 & 0.3062 & 0.0267 & 0.4686 & 0.0660 \end{pmatrix}. \quad (9)$$

The clustering accuracy of our proposal is 74.5% if we label the series to the cluster in terms of the largest membership. In Figure 3, we can find the changes cluster of the 6th, 7th, and 18th series. The 6th series primarily belongs to the 5th cluster. In segment from about 320 to 380, this series is obviously closed to another cluster. The 18th series switches between two clusters and this result corresponded to its membership. These results present propitious analysis that help in predicting some properties of time series. Compared with existing classification or clustering methods (http://www.cs.ucr.edu/~eamonn/time_series_data/), our result is acceptable.

5. Discussion and Conclusion

At present, most cluster methods for time series directly adopt the methods that deal with the problem of grouping the static dataset. They usually consider the time series as the points in an n-dimensional space. By doing so, the property of dynamic behavior of time series over time is neglected. However, the dynamic behavior or evolutionary nature of time series is a very important property when clustering them. For instance, a switching series belongs to different clusters in different time segments. For some time series set, it is possible that at the beginning, the set can be grouped into m clusters and then n $(n \neq m)$ clusters after a certain time points. These problems suggest the need to develop new methods to cluster time series which do not directly employ the cluster methods for static data to implement cluster for time series. In this sense, this paper can be considered as an attempt along this way.

No matter what we employ that the existing crisp or fuzzy clustering method, it is impossible to find the switching property of switching time series over time. Furthermore, this evolutionary property is a basic nature of time series and should be reflected when studying it. Thus, we propose a dynamic fuzzy cluster algorithm to reveal the evolution property for time series by finding key points and improving FCM algorithm. Different from the existing fuzzy cluster methods, the proposed algorithm can only allows each time series to belong to different clusters with various membership degrees but also reveals the changes procedure of clustering switching series over time. This is helpful in predicting and analyzing the evolutionary properties for time series.

References

[1] P. D'Urso and E. A. Maharaj, "Autocorrelation-based fuzzy clustering of time series," *Fuzzy Sets and Systems*, vol. 160, no. 24, pp. 3565–3589, 2009.

[2] P. D'Urso and E. A. Maharaj, "Wavelets-based clustering of multivariate time series," *Fuzzy Sets and Systems*, vol. 193, pp. 33–61, 2012.

[3] T. W. Liao, "Clustering of time series data—a survey," *Pattern Recognition*, vol. 38, no. 11, pp. 1857–1874, 2005.

[4] D. Chakrabarti, R. Kumar, and A. Tomkins, "Evolutionary Clustering," in *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '06)*, pp. 554–560, Philadelphia, Pa, USA, 2006.

[5] Y. Chi, X. D. Song, D. Y. Zhou, K. Hino, and B. L. Tseng, "On evolutionary spectral clustering," *ACM Transactions on Knowledge Discovery from Data*, vol. 3, no. 4, article 17, 2009.

[6] M. Corduas and D. Piccolo, "Time series clustering and classification by the autoregressive metric," *Computational Statistics & Data Analysis*, vol. 52, no. 4, pp. 1860–1872, 2008.

[7] Y. M. Xiong and D. Y. Yeung, "Time series clustering with ARMA mixtures," *Pattern Recognition*, vol. 37, no. 8, pp. 1675–1689, 2004.

[8] J. A. Vilar, A. M. Alonso, and J. M. Vilar, "Non-linear time series clustering based on non-parametric forecast densities," *Computational Statistics & Data Analysis*, vol. 54, no. 11, pp. 2850–2865, 2010.

[9] J. G. Brida, D. M. Gómez, and W. A. Risso, "Symbolic hierarchical analysis in currency markets: an application to contagion in currency crises," *Expert Systems with Applications*, vol. 36, no. 4, pp. 7721–7728, 2009.

[10] X. H. Zhang, J. Q. Liu, Y. Du, and T. J. Lv, "A novel clustering method on time series data," *Expert Systems with Applications*, vol. 38, no. 9, pp. 11891–11900, 2011.

[11] M. C. Chiang, C. W. Tsai, and C. S. Yang, "A time-efficient pattern reduction algorithm for k-means clustering," *Information Sciences*, vol. 181, no. 4, pp. 716–731, 2011.

[12] E. Keogh and S. Kasetty, "On the need for time series data mining benchmarks: a survey and empirical demonstration," *Data Mining and Knowledge Discovery*, vol. 7, no. 4, pp. 349–371, 2003.

[13] E. Keogh, J. Lin, and W. Truppel, "Clustering of time series subsequences is meaningless: implications for previous and future research," in *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM '03)*, pp. 115–122, November 2003.

[14] T. Rakthanmanon, E. Keogh, S. Lonardi, and S. Evans, "Time series epenthesis: clustering time series streams requires ignoring some data," in *Proceedings of the IEEE 11th International Conference on Data Mining (ICDM '11)*, 2011, http://www.cs.ucr.edu/~stelo/papers/ICDM11_TSE.pdf.

[15] E. Keogh, X. Xi, L. Wei, and C. A. Ratanamahatana, The UCR time series classication/clustering homepage, http://www.cs.ucr.edu/~eamonn/time_series_data/.

[16] T. C. Fu, "A review on time series data mining," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 1, pp. 164–181, 2011.

[17] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651–666, 2010.

[18] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," *Journal of Cybernetics*, vol. 3, no. 3, pp. 32–57, 1973.

[19] F. Höppner and F. Klawonn, "Compensation of translational displacement in time series clustering using cross correlation," in *Advances in Intelligent Data Analysis VIII*, N. M. Adams, C. Robardet, A. Siebes, and J. F. Boulicaut, Eds., pp. 71–82, Springer, Berlin, Germany, 2009.

[20] F. Klawonn, "Fuzzy clustering: insights and a new approach," *Mathware & Soft Computing*, vol. 11, no. 2-3, pp. 125–142, 2004.

[21] R. Killick, I. A. Eckley, K. Ewans, and P. Jonathan, "Detection of changes in variance of oceanographic time-series using changepoint analysis," *Ocean Engineering*, vol. 37, no. 13, pp. 1120–1126, 2010.

[22] S. Eschrich, J. W. Ke, L. O. Hall, and D. B. Goldgof, "Fast accurate fuzzy clustering through data reduction," *IEEE Transactions on Fuzzy Systems*, vol. 11, no. 2, pp. 262–270, 2003.

[23] C. S. Möller-Levet, F. Klawonn, K. H. Cho, and O. Wolkenhauer, "Fuzzy clustering of short time-series and unevenly distributed sampling points," in *Advances in Intelligent Data Analysis V*, vol. 2810 of *Lecture Notes in Computer Science*, pp. 330–340, 2003.

[24] E. N. Nasibov and S. Peker, "Time series labeling algorithms based on the K-nearest neighbors' frequencies," *Expert Systems with Applications*, vol. 38, no. 5, pp. 5028–5035, 2011.

[25] S. R. Kannan, S. Ramathilagam, and P. C. Chung, "Effective fuzzy c-means clustering algorithms for data clustering problems," *Expert Systems With Applications*, vol. 39, pp. 6292–6300, 2012.

[26] J. Mennis and D. S. Guo, "Spatial data mining and geographic knowledge discovery—an introduction," *Computers, Environment and Urban Systems*, vol. 33, no. 6, pp. 403–408, 2009.

[27] M. F. Macchiato, L. la Rotonda, V. Lapenna, and M. Ragosta, "Time modelling and spatial clustering of daily ambient temperature: an application in southern Italy," *Environmetrics*, vol. 6, no. 1, pp. 31–53, 1995.

[28] P. S. P. Cowpertwait and T. F. Cox, "Clustering population means under heterogeneity of variance with an application to a rainfall time series problem," *The Statistician*, vol. 41, no. 1, pp. 113–121, 1992.

[29] I. Horenko, "On clustering of non-stationary meteorological time series," *Dynamics of Atmospheres and Oceans*, vol. 49, no. 2-3, pp. 164–187, 2010.

[30] N. Y. Wang and S. M. Chen, "Temperature prediction and TAIFEX forecasting based on automatic clustering techniques and two-factors high-order fuzzy time series," *Expert Systems with Applications*, vol. 36, no. 2, pp. 2143–2154, 2009.

[31] A. M. Alonso, J. R. Berrendero, A. Hernández, and A. Justel, "Time series clustering based on forecast densities," *Computational Statistics & Data Analysis*, vol. 51, no. 2, pp. 762–776, 2006.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 196256, 6 pages
http://dx.doi.org/10.1155/2013/196256

Research Article

A Cost-Sensitive Ensemble Method for Class-Imbalanced Datasets

Yong Zhang and Dapeng Wang

School of Computer and Information Technology, Liaoning Normal University, No. 1, Liushu South Street, Ganjingzi, Dalian, Liaoning 116081, China

Correspondence should be addressed to Yong Zhang; zhyong@lnnu.edu.cn

Received 28 December 2012; Accepted 25 March 2013

Academic Editor: Jianhong (Cecilia) Xia

In imbalanced learning methods, resampling methods modify an imbalanced dataset to form a balanced dataset. Balanced data sets perform better than imbalanced datasets for many base classifiers. This paper proposes a cost-sensitive ensemble method based on cost-sensitive support vector machine (SVM), and query-by-committee (QBC) to solve imbalanced data classification. The proposed method first divides the majority-class dataset into several subdatasets according to the proportion of imbalanced samples and trains subclassifiers using AdaBoost method. Then, the proposed method generates candidate training samples by QBC active learning method and uses cost-sensitive SVM to learn the training samples. By using 5 class-imbalanced datasets, experimental results show that the proposed method has higher area under ROC curve (AUC), F-measure, and G-mean than many existing class-imbalanced learning methods.

1. Introduction

In the classification problem field, the scenario of imbalanced data sets appears when the number of samples that represent the different classes is very different among them [1]. Class-imbalanced problems widely exist in the fields of medical diagnosis, fraud detection, network intrusion detection, science and engineering problems, and so on. We consider the binary-class-imbalanced data sets, where there is only one positive (minority) class and one negative (majority) class. Most of data are in the majority class, and little data are in the minority class. Many traditional classification methods tend to be overwhelmed by the majority class and ignore the minority class. The classification performance for the positive class becomes unsatisfactory.

It is important to select the suitable training data for classification in the class-imbalanced classification problem. Resampling is one of the effective techniques for adjusting the size of training sets. Many resampling methods are used to reduce or eliminate the extent of data set imbalance, such as oversampling the minority class, undersampling the majority class, and the combination of both methods. Resampling techniques can be used with many base classifiers, such as support vector machine (SVM), C4.5, Naïve Bayes classifier, and AdaBoost, to address the class-imbalanced problem. So, it provides a convenient and effective way to deal with imbalanced learning problems using standard classifiers [2]. Additionally, modified learning algorithmic solutions are the effective approaches to the imbalanced data classification problem. These solutions are obtained by modifying existing learning algorithms so that they can deal with imbalanced problems effectively. Integrated approach, cost-sensitive learning, feature selection, and single-class learning belong to the solutions. Cost-sensitive learning deals with class imbalance by incurring different costs for the two classes and is considered an important type of methods to handle class imbalance. The difficulty with cost-sensitive classification is that costs of misclassification are often unknown [3].

Although the existing imbalance-learning methods applied for normal SVMs can solve the problem of class imbalance, they can ignore potential useful information in major samples, and probably lead to overfitting problem.

This paper presents a cost-sensitive ensemble method. The proposed method uses AdaBoost method to train subclassifiers according to the ratio of imbalanced samples, integrates these sub-classifiers into a classifier, and uses cost-sensitive SVM to train the candidate data selected by a query-by-committee (QBC) algorithm.

The rest of the paper is organized as follows. Following the introduction, Section 2 presents a comprehensive study on the class-imbalanced problem and discusses the existing class-imbalanced solutions. Section 3 simply introduces cost-sensitive SVM. Section 4 proposes a cost-sensitive ensemble method for class-imbalanced data sets. In Section 5, we apply a statistical test to compare the performance of the proposed method with the existing methods. Finally, Section 6 concludes this paper.

2. Related Work

Many techniques are proposed to solve classification problems based on imbalanced data sets. There are two major categories of techniques developed to address the class-imbalance issue. One is resampling and the other is modified learning algorithmic solutions [4].

Resampling is one of the effective techniques for adjusting the size of a training dataset. In general, it can be further divided into undersampling approach and over-sampling approach. Undersampling uses only some samples of the majority class to reduce the data size and removes samples of the majority class to balance a data set. So the risk is that the reduced sample set may not represent the full characteristics of the majority class. There are many studies which discuss under-sampling methods. For example, Kim [5] proposes an under-sampling method based on a self-organizing map (SOM) neural network to obtain sampling data which retains the original data characteristics. Yen and Lee [6] present a cluster-based under-sampling approach for selecting the representative data as training data. The proposed method improves the classification accuracy for the minority class. Aiming at the deficiency of under-sampling where many majority-class samples are ignored, Liu et al. [7] propose two effective informed under-sampling methods, EasyEnsemble and BalanceCascade. EasyEnsemble method samples several subsets from the majority-class, trains a learner using each of them, and combines the outputs of those learners. BalanceCascade method trains the learners sequentially. In each step of BalanceCascade, the majority class samples which are correctly classified by the current trained learners are removed from further consideration.

The over-sampling approach is to add more new data instances to the minority class to balance a data set. These new data instances can either be generated by replicating the data instances of the minority class or by applying synthetic methods. However, over-sampling often involves making exact copies of samples which may lead to overfitting [8]. synthetic minority oversampling technique (SMOTE) [1] is an intelligent over-sampling method using synthetic samples. SMOTE method adds new synthetic samples to the minority class by randomly interpolating pairs of the closest neighbors in the minority class. SMOTEBoost algorithm [9] combines SMOTE technique and the standard boosting procedure. It utilizes SMOTE for improving the accuracy over the minority class and utilizes boosting not to sacrifice accuracy over the entire data set. Wang et al. [10] propose an adaptive over-sampling technique based on data density (ASMOBD), which can adaptively synthesize different number of new samples around each minority sample according to its level of learning difficulty. Gao et al. [11] propose probability density function estimation based on over-sampling approach for two class-imbalanced classification problems.

At the algorithmic level, the solutions mainly include cost-sensitive learning, integrated approach, and modified algorithms. Many cost-sensitive learning methods have been proposed [12, 13]. A common strategy of these methods is to intentionally increase the weights of samples with higher misclassification cost in the boosting process. However, misclassification costs are often unknown, and a cost-sensitive classifier may result in over-fitting training. Sun et al. [14] investigate cost-sensitive boosting algorithms for advancing the classification of imbalanced data and propose three cost-sensitive boosting algorithms by introducing cost items into the learning framework of AdaBoost. Guo and Viktor [15] propose a modified boosting procedure, DataBoost, to solve the imbalanced problem. DataBoost combines the boosting and ensemble-based learning algorithms. In terms of modified algorithms, several specific attempts using SVMs have been made at improving their class prediction accuracy in the case of class imbalances [16, 17]. The results obtained with such methods show that SVMs have the particular advantage of being able to solve the problem of skewed vector spaces, without introducing noise. Wang and Japkowicz [13] combine modifying the data distribution approach and modifying the classifier approach in class-imbalanced problem and use support vector machines with soft margins as the base classifier to solve the skewed vector spaces problem.

In addition, Wang et al. [18] develop two models to yield the feature extractors and propose a method for extracting minimum positive and maximum negative features for imbalanced binary classification. Based on the divide-and-conquer principle, the scalable instance selection approach OligoIS is proposed in [19] for class-imbalanced data sets. OligoIS can deal with the class-imbalanced problem that is scalable to data sets with many millions of instances and hundreds of features.

3. Cost-Sensitive SVM

SVM has been widely used in many application areas of machine learning. The goal of the SVM-learning algorithm is to find a separating hyperplane that separates these data points into two classes. In order to find a better separation of classes, the data are first transformed into a higher-dimensional feature space. However, regular SVM is invalid to the imbalanced data sets. For imbalanced data sets, the learned boundary is too close to the minority samples, so SVM should be biased in a way that will push the boundary away from the positive samples [16]. Using different error

costs for the positive and negative classes, SVM can be extended to the cost-sensitive setting by introducing an additional parameter that penalizes the errors asymmetrically.

Consider that we have a binary classification problem, which is represented by a data set $\{(x_1, y_1), (x_2, y_2), \ldots, (x_l, y_l)\}$, where $x_i \subset \mathfrak{R}^k$ represents a k-dimensional data point and $y_i \in \{+1, -1\}$ represents the class of that data point, for $i = 1, \ldots, l$. Let $I_+ = \{i : y_i = +1\}$ and $I_- = \{i : y_i = -1\}$. The support vector technique requires the solution of the quadratic programming problem as follows [20]:

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C^+ \sum_{i \in I_+} \xi_i + C^- \sum_{i \in I_-} \xi_i \tag{1}$$

subject to

$$\begin{aligned} &y_i \left(w \cdot \phi\left(x_i\right) + b\right) \geq 1 - \xi_i, \\ &\xi_i \geq 0, \quad i = 1, \ldots, l, \end{aligned} \tag{2}$$

where the training vectors x_i are mapped into a higher-dimensional space by the function ϕ. Parameter C^+ represents the cost of misclassifying the positive sample, and C^- represents the cost of misclassifying the negative sample. The optimal result can be obtained when C^-/C^+ equals the minority-to-majority class ratio. The slack variables $\xi_i > 0$ hold for misclassified samples, and therefore, $\sum_{i=1}^{l} \xi_i$ can be thought of as a measure of the amount of misclassifications. This quadratic-optimization problem can be solved by constructing a Lagrangian representation and transforming it into the following dual problem:

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j K\left(x_i, x_j\right) \tag{3}$$

subject to

$$\begin{aligned} &0 \leq \alpha_i \leq C^+ \quad \text{for } i \in I_+, \\ &0 \leq \alpha_i \leq C^- \quad \text{for } i \in I_-, \\ &\sum_{i=1}^{l} \alpha_i y_i = 0, \end{aligned} \tag{4}$$

where α_i is the Lagrangian parameter. Note that the kernel trick $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$ is used in (3).

4. An Ensemble Method Based on Cost-Sensitive SVM and QBC

This paper presents an ensemble method based on cost-sensitive SVM and QBC, called CQEnsemble, specifically designed for imbalanced data classification. The proposed method applies division and boost techniques to a simple QBC strategy [21, 22] and improves classification precision on the basis of maximizing data balance. In order to overcome the shortages of over-sampling and under-sampling, the CQEnsemble method trains sub-classifiers using AdaBoost algorithm [23] according to the ratio of imbalanced samples and integrates these sub-classifiers into a classifier. AdaBoost can be used in conjunction with many otherlearning algorithms to improve their performance. In this way, the proposed method not only fully uses the minority class information but also feedbacks the different aspects of information of the majority class.

Suppose that an imbalanced dataset contains n samples from the majority class and m samples from the minority class where $n \gg m$. First, the CQEnsemble method divides training data set into m equivalent subsets, where m is greater than or equal to 3. Then, we randomly select two subsets and generate two sub-classifiers as QBCs committees to vote for the other $m - 2$ equivalent subsets. We add samples, in which the vote results are different in two QBC's committees, to candidate data set. It is difficult to decide the category of these samples. So, these samples probably include abundant information. Last, we integrate candidate data set and two selected subsets into new training datasets, train, and get a classifier using cost-sensitive SVM method. Experiments of this paper show that the CQEnsemble method can get comprehensive classification information when the value of m is 5.

Based on the description above, the proposed CQEnsemble method is described as follows.

Algorithm 1 (the CQEnsemble method).

Input. Imbalanced data set D.

Output. An ensemble classifier H.

Step 1. Suppose that the training set is A and the total number of samples is n. Divide A into m ($m \geq 3$) equivalent subsets randomly, labeled as N_i ($i = 1, 2, \ldots, m$).

Step 2. Select two subsets randomly and label them as N_i ($i = 1, 2$) conveniently. For each subset N_i do

Step 2.1. Compute the ratio of the number of majority-class samples to the number of minority-class samples r_i ($i = 1, 2$).

Step 2.2. Divide the majority-class samples into r_i subsets.

Step 2.3. Merge the minority-class samples and each subset to the training set, and get r_i training sets.

Step 2.4. Classify each training set in Step 2.3 using AdaBoost algorithm, and get r_i weak classifiers H_{ij}, where $j = 1, 2, \ldots, r_i$.

Step 2.5. Regard these weak classifiers H_{ij} as features, and integrate into classifier H_i.

End for

Step 3. Use classifiers H_i ($i = 1, 2$) to respectively train samples in the rest $m - 2$ subsets, and add samples in which the results are different in two classifiers H_i ($i = 1, 2$) to new candidate set D_c.

Step 4. Merge two selected subsets N_i ($i = 1, 2$) to the candidate set D_c, and get a new training set F.

Step 5. Classify data set F using cost-sensitive SVM method, and get a classifier H.

5. Experiment and Analysis

In this section, we first give several evaluation measures for class-imbalanced problem, and then present and discuss, in detail, the results obtained by the experiments carried out in this research.

5.1. Evaluation Measures. Accuracy is an important evaluation metric for assessing the classification performance and guiding the classifier modeling. However, accuracy is not a useful measure for imbalanced data, particularly when the number of instances of the minority class is very small compared with the majority class [24]. For example, if we have a ratio of 1 : 100, a classifier that assigns all instances to the majority class will have 99% accuracy. But this measurement is meaningless to some applications where the learning concern is the identification of the rare cases.

Several measures have been developed to deal with the classification problem with the class imbalance, including *F-measure*, *G-mean*, and *AUC* [25]. Given the number of true positives (TPs), false positives (FPs), true negatives (TNs), and false negatives (FNs), we can obtain the confusion matrix presented in Table 1 after a classification process. We can also define several common measures. The TP rate TPR, recall R, or sensitivity S_n is defined as

$$\text{TPR} = R = S_n = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{5}$$

The TN rate TNR or specificity S_p is defined as

$$\text{TNR} = S_p = \frac{\text{TN}}{\text{TN} + \text{FP}}. \tag{6}$$

Precision P is defined as the fraction of relevant instances that are retrieved as follows:

$$P = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{7}$$

Based on these measures, other measures have been presented, such as *F-measure* and *G-mean*. *F-measure* is often used in the fields of information retrieval and machine learning for measuring search, document classification, and query classification performance. *F-measure* considers both the precision P and the recall R to compute the score [26]. It can be interpreted as a weighted average of the precision and recall as follows:

$$F\text{-}measure = \frac{2 \times P \times R}{P + R}. \tag{8}$$

G-mean is defined by two parameters called sensitivity S_n and specificity S_p. Sensitivity shows the performance of the positive class, and specificity shows the performance of the negative class. *G-mean* measures the balanced performance of a learning algorithm between these two classes. *G-mean* is defined as

$$G\text{-}mean = \sqrt{S_n \times S_p}. \tag{9}$$

Table 1: Confusion matrix.

	Predicted positive class	Predict negative class
Actual positive class	TP (true positive)	FN (false negative)
Actual negative class	FP (false positive)	TN (true negative)

A receiver operating characteristic (ROC) curve is a graphical plot which depicts the performance of a binary classifier as its discrimination threshold is varied. In an ROC curve, the true positive rate (sensitivity) is plotted in function of the false positive rate (specificity) for different cut-off points. Each point on the ROC curve represents a (sensitivity, specificity) pair corresponding to a particular decision threshold. The ideal point on the ROC curve would be (0, 1); that is, all positive samples are classified correctly, and no negative samples are misclassified as positive. An ROC curve depicts relative trade-offs between benefits (true positives) and costs (false positives) across a range of thresholds of a classification model. However, it is difficult to decide which one is the best method when comparing several classification models. *AUC* is the area under an ROC curve. It has been proved to be a reliable performance measure for imbalanced and cost-sensitive problems [25]. *AUC* provides a single measure of a classifier's performance for evaluating which model is better on average.

5.2. Experimental Results and Analysis. In our experiments, we used 5 data sets to test the performance of the proposed method. These data sets are from the UCI Machine Learning Repository [27]. Information about these data sets is summarized in Table 2. These data sets vary extensively in their sizes and class proportions. We take the minority class as the target class and all the other categories as majority class. When more than two classes exist in the data set, the target class is considered to be positive and all the other classes are considered to be negative. We compared the performance of 5 methods, including AdaBoost, SMOTE [1], SMOTEBoost [9], EasyEnsemble [7], and our proposed CQEnsemble method.

In our experiments, *F-measure*, *G-mean,* and *AUC* are used as metrics. For each data set, we perform a 5-fold cross validation. In each fold four out of five samples are selected to be training set, and the left one out of five samples is testing set. This process repeats 5 times so that all samples are selected in both training set and testing set.

Figure 1 shows the average *F-measure* values of the compared methods. The results show that CQEnsemble has higher *F-measure* than other compared methods on *haberman*, *pima,* and *letter* data sets. EasyEnsemble achieves the highest *F-measure* on *transfusion* data set among these methods, and AdaBoost achieves the highest *F-measure* on *phoneme* data set. The results indicate that CQEnsemble can further improve the *F-measure* metric of imbalanced learning.

Table 2: Summary of data sets.

Data set	Total samples	no of attributes	no of positive	no of negative	Ratio (majority/minority)
Haberman	306	3	81	225	2.8
Transfusion	926	4	178	748	4.2
Pima	768	8	268	500	1.9
Phoneme	5404	5	1586	3818	2.4
Letter	20000	16	789	19211	24.3

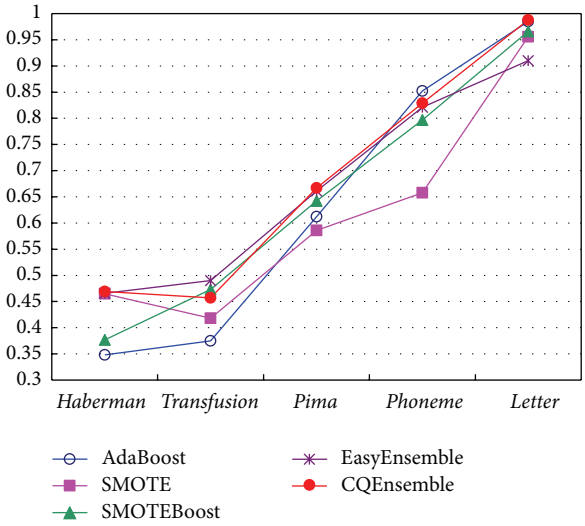


Figure 1: *F-measure* of the compared methods.

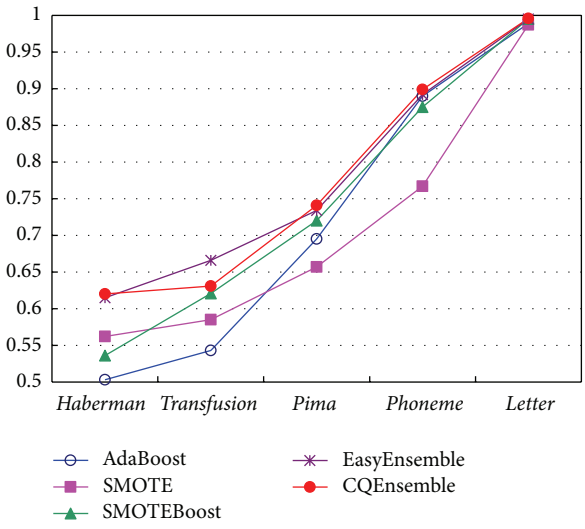


Figure 2: *G-mean* of the compared methods.

The average *G-mean* values of the compared methods are summarized in Figure 2. The results show that CQEnsemble has higher *G-mean* than other compared methods on most of datasets, while EasyEnsemble is slightly higher *G-mean* than CQEnsemble on *transfusion* dataset. From Figures 1 and 2, EasyEnsemble has the highest *F-measure* and *G-mean* on *transfusion* dataset among these methods.

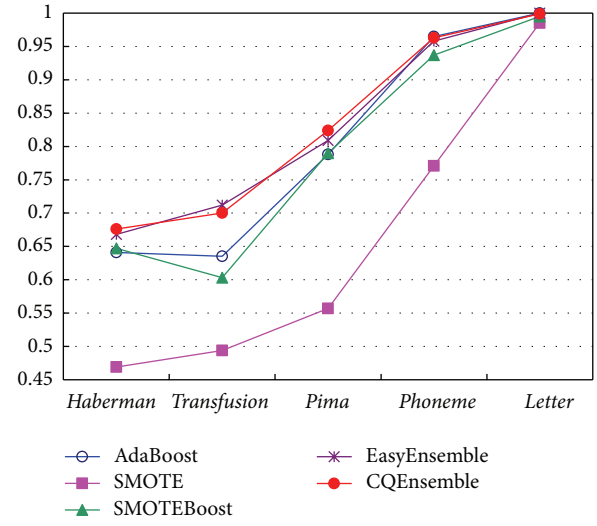


Figure 3: *AUC* of the compared methods.

Figure 3 shows the *AUC* metric of each method for *haberman*, *transfusion*, *pima*, *phonem,e* and *letter* data sets. The results show that the proposed CQEnsemble method obtains the highest average *AUC* among these compared methods. These methods are equivalent for *letter* data set. After all, SMOTE method is the weakest in 5 methods; EasyEnsemble method is slightly better than AdaBoost, SMOTE, and SMOTEBoost, while CQEnsemble method is better than EasyEnsemble method. The results show that the CQEnsemble method effectively avoids the shortages of resampling methods.

CQEnsemble attains higher average *F-measure*, *G-mean*, and *AUC* than almost all the other methods, except that CQEnsemble is slightly worse comparable to EasyEnsemble with *F-measure*, *G-mean*, and *AUC* on *transfusion* data set. The experimental results imply that the proposed CQEnsemble method is better than AdaBoost, SMOTE, SMOTEBoost, and EasyEnsemble methods on most of data sets. These experiments also indicate that the combination of division-boost method and cost-sensitive learning can further improve the performance of imbalanced learning.

6. Conclusions

In this paper, we propose CQEnsemble method based on cost-sensitive SVM and QBC to solve imbalanced data classification. CQEnsemble method divides the majority class

into several subsets according to the proportion of imbalance samples. CQEnsemble method selects the effective training samples to join the last training set based on QBC active learning algorithm, so it avoids the shortages of the over-sampling and under-sampling. Experiment results show that the proposed method has higher *F-measure*, *G-mean,* and *AUC* than many existing class-imbalance learning methods.

Acknowledgments

This work is supported by China Postdoctoral Science Foundation (no. 20110491530) and Science Research Plan of Liaoning Education Bureau (no. L2011186).

References

[1] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[2] M. Gao, X. Hong, S. Chen, and C. J. Harris, "A combined SMOTE and PSO based RBF classifier for two-class imbalanced problems," *Neurocomputing*, vol. 74, pp. 3456–3466, 2011.

[3] G. Weiss, "Mining with rarity: a unifying framework," *SIGKDD Explorations*, vol. 6, no. 1, pp. 7–19, 2004.

[4] C. Seiffert, T. M. Khoshgoftaar, J. van Hulse, and A. Napolitano, "RUSBoost: a hybrid approach to alleviating class imbalance," *IEEE Transactions on Systems, Man, and Cybernetics A*, vol. 40, no. 1, pp. 185–197, 2010.

[5] M. S. Kim, "An effective under-sampling method for class imbalance data problem," in *Proceedings of the 8th Symposium on Advanced Intelligent Systems*, pp. 825–829, 2007.

[6] S. J. Yen and Y. S. Lee, "Cluster-based under-sampling approaches for imbalanced data distributions," *Expert Systems with Applications*, vol. 36, no. 3, pp. 5718–5727, 2009.

[7] X. Y. Liu, J. X. Wu, and Z. H. Zhou, "Exploratory undersampling for class-imbalance learning," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 39, no. 2, pp. 539–550, 2009.

[8] C. Drummond and R. C. Holte, "C4.5 decision tree, class imbalance, and cost sensitivity: why under-sampling beats over-sampling," in *Proceedings of the Workshop on Learning from Imbalanced Data Sets II, International Conference on Machine Learning*, 2003.

[9] N. V. Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer, "SMOTEBoost: improving prediction of the minority class in boosting," in *Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD '03)*, pp. 107–119, September 2003.

[10] S. Wang, Z. Li, W. Chao, and Q. Cao, "Applying adaptive over-sampling technique based on data density and cost-sensitive SVM to imbalanced learning," in *The International Joint Conference on Neural Networks (IJCNN '12)*, 2012.

[11] M. Gao, X. Hong, S. Chen, and C. J. Harris, "Probability density function estimation based over-sampling for imbalanced two-class problems," in *The International Joint Conference on Neural Networks (IJCNN '12)*, 2012.

[12] C. Elkan, "The foundations of cost-sensitive learning," in *Proceedings of the 17th International Joint Conference on Artificial Intelligence*, pp. 973–978, 2001.

[13] B. X. Wang and N. Japkowicz, "Boosting support vector machines for imbalanced data sets," *Knowledge and Information Systems*, vol. 25, no. 1, pp. 1–20, 2010.

[14] Y. Sun, M. S. Kamel, A. K. C. Wong, and Y. Wang, "Cost-sensitive boosting for classification of imbalanced data," *Pattern Recognition*, vol. 40, no. 12, pp. 3358–3378, 2007.

[15] H. Guo and H. L. Viktor, "Learning from imbalanced data sets with boosting and data generation: the DataBoost-IM approach," *SIGKDD Explorations*, vol. 6, no. 1, pp. 30–39, 2004.

[16] R. Akbani, S. Kwek, and N. Japkowicz, "Applying support vector machines to imbalanced datasets," in *Proceedings of the 15th European Conference on Machine Learning (ECML '04)*, pp. 39–50, Pisa, Italy, September 2004.

[17] Y. Tang, Y. Q. Zhang, and N. V. Chawla, "SVMs modeling for highly imbalanced classification," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 39, no. 1, pp. 281–288, 2009.

[18] J. Wang, J. You, Q. Li, and Y. Xu, "Extract minimum positive and maximum negative features for imbalanced binary classification," *Pattern Recognition*, vol. 45, pp. 1136–1145, 2012.

[19] N. García-Pedrajas, J. Pérez-Rodríguez, and A. de Haro-García, "OligoIS: scalable instance selection for class-imbalanced data sets," *IEEE Transactions on Systems, Man, and Cybernetics B*, 2012.

[20] K. Veropoulos, C. Campbell, and N. Cristianini, "Controlling the sensitivity of support vector machines," in *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 55–60, 1999.

[21] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, pp. 287–294, July 1992.

[22] Y. Freund, H. S. Seung, E. Shamir, and N. Tishby, "Selective sampling using the query by committee algorithm," *Machine Learning*, vol. 28, no. 2-3, pp. 133–168, 1997.

[23] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *Proceedings of the 2nd European Conference on Computational Learning Theory*, pp. 23–37, 1995.

[24] M. V. Joshi, V. Kumar, and R. C. Agarwal, "Evaluating boosting algorithms to classify rare classes: comparison and improvements," in *Proceedings of the 1st IEEE International Conference on Data Mining (ICDM '01)*, pp. 257–264, December 2001.

[25] T. Fawcett, "ROC graphs: notes and practical considerations for researchers," Tech. Rep. HPL-2003-4, HP Labs, Palo Alto, Calif, USA, 2003.

[26] D. Lewis and W. Gale, "Training text classifiers by uncertainty sampling," in *Proceedings of the 7th Annual International ACM SIGIR Conference on Research and Development in Information*, pp. 73–79, New York, NY, USA, 1998.

[27] A. Frank and A. Asuncion, *UCI Machine Learning Repository*, University of California, School of Information and Computer Science, Irvine, Calif, USA, 2010, http://archive.ics.uci.edu/ml/.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 693194, 6 pages
http://dx.doi.org/10.1155/2013/693194

Research Article

A Study on Coastline Extraction and Its Trend Based on Remote Sensing Image Data Mining

Yun Zhang,[1,2,3] Xueming Li,[1] Jianli Zhang,[2,3] and Derui Song[2,3]

[1] *Liaoning Normal University, Dalian 116029, China*
[2] *National Marine Environmental Monitoring Center, Dalian 116023, China*
[3] *Key Laboratory of Sea Areas Management Technology, SOA, Dalian 116023, China*

Correspondence should be addressed to Yun Zhang; cloud208@163.com

Received 31 January 2013; Accepted 21 March 2013

Academic Editor: Jianhong (Cecilia) Xia

In this paper, data mining theory is applied to carry out the field of the pretreatment of remote sensing images. These results show that it is an effective method for carrying out the pretreatment of low-precision remote sensing images by multisource image matching algorithm with SIFT operator, geometric correction on satellite images at scarce control points, and other techniques; the result of the coastline extracted by the edge detection method based on a chromatic aberration Canny operator has a height coincident with the actual measured result; we found that the coastline length of China is predicted to increase in the future by using the grey prediction method, with the total length reaching up to 19,471,983 m by 2015.

1. Introduction

The coastline is an adjacent transition zone between land and ocean that is the result of dynamic change processes involving natural changes and human activities. At present, 60% of the world's population is gathered near a coastal zone rich in metallic mineral resources, oil and gas resources, tidal and wave energy sources, and other renewable energy sources. The coastline is an important place for human economic and social activities. A quick and accurate determination of the coastline length variation trend is not only a necessary technical activity for the study of land-ocean interactions, coastal reclamation, port development, and urban expansion, but also an important subject for marine economics and marine multidisciplinary research.

There are two coastline extraction methods. One is the field detection method, which is the more common method, that is, the photogrammetric technology and GPS technology. The other is the remote sensing image coastline automatic extraction technology and image interpretation. Such software and technologies for coastline extraction have gradually become more mature, as discussed by Li and Jigang, Lee et al., Manavalan et al., Holman et al., Rudin et al., Chan et al., Donoho, Xiaofeng et al., and Chaoyang [1–9]. The automatic remote sensing image classification have been well researched, such as the proposal by Jiang et al. for an automatic scheme for the classification of land use based on change detection and a semisupervised classifier [10]. Stavrakoudis et al. [11] built a boosted genetic fuzzy classifier for land cover classification of remote sensing imagery. Although the coastline can be extracted by these two methods, they fail to include a systematic and comprehensive analysis of coastline length and type variation trend, which means that the automatic sensing image classifications do not meet the shoreline classification. The spatial data mining theory presented in this study is aimed at the image processing of remote sensing image data and is focused on the discovery of potential, hidden, and useful models and rules between the image targets from remote sensing images as discussed by Xiaocheng and Xiaoqin [12]. It combines the spatial data mining method as discussed by Dengke, Deren et al., Zhanquan, Kaichang et al., and Longshu et al. [13–17] with classic remote sensing image processing technology as discussed by Aihua et al., Deren and Juliang, Guobao et al., and Tao and Bin [18–21]. A new coastline extraction and analysis model was put forward to analyze

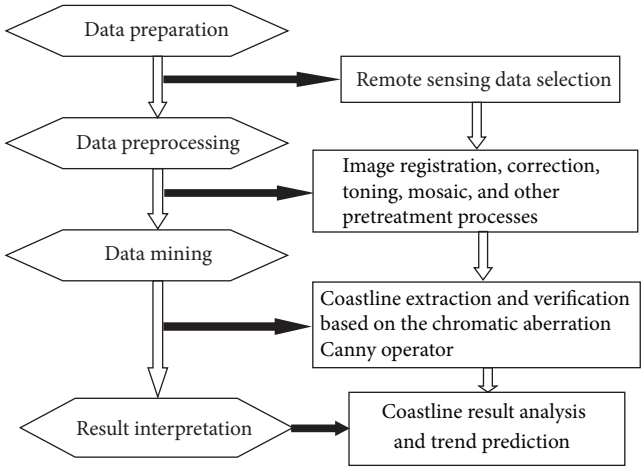


Figure 1: Coastline extraction flow chart based on data mining.

the future variation trend of coastline length by means of grey prediction algorithm to achieve the resulting conclusions and to propose feasible recommendations for the exploitation of the future coastal zone.

2. Coastline Extraction Process

Data mining refers to the process as discussed by Kantardzic et al. [22], in which implicit, unknown, potential, and useful information and knowledge can be extracted from a large number of incomplete, noisy, fuzzy, and random databases. According to the knowledge structure of data mining, the coastline extraction processes can be divided into four stages as follows in Figure 1: image selection; image registration, correction, toning, mosaic, and other pretreatment processes; coastline extraction and mining; and the coastline result analysis and trend prediction.

3. Study on the Remote Sensing Image Coastline Extraction Technology

3.1. Data Preparation. The remote sensing image data presented and selected in this paper were collected at five stages from 2007 to 2011. The space range covered the coastal zones of the Chinese Mainland (including Hong Kong and Macao) and Hainan Island encompassing 290 scenic spots. Details of the images are as in Table 1.

Table 1: Image details in 2007–2011.

Period	Satellite designation	Spatial resolution (m)	Quantity (scenic spots)
2011	HJ-1A	30	65
2010	HJ-1A	30	42
2009	HJ-1A	30	34
2008	CBERS-01	20	86
2007	CBERS-01	20	63

HJ-1A satellite is the abbreviation of environmental and disaster monitoring and forecasting; CBERS-01 satellite was jointly invested in and developed by China and Brazil on October 14, 1999. Because HJ-1A images and CBERS-01 images are low-resolution areas, the results of shoreline extracted have small differences, but this does not affect this study.

3.2. Data Preprocessing

3.2.1. Automatic Registration of the Coastal-Zone Image. The multisource image matching algorithm of a SIFT operator was presented in this paper, which provided different spectral band setting, imaging model, reflectivity, dimension, and other aspects of the multisource data. RANSAC and the identical point triangulation network construction method were used for eliminating the mismatching points in order to achieve the automatic matching of multisource remote sensing images and control point image databases, after which the automatic image registration was completed.

3.2.2. Geometric Correction of the Coastal-Zone Image. It was difficult to find the control points due to the large water area on the remote sensing image at the coastal zone. Meanwhile, a small number of control points were used to achieve the geometric correction of the low-precision remote sensing images in order to reduce the workload of the field measurements. Combined with satellite orbit parameters and the geometric correction model with correction precision in different satellite data sources and terrain conditions, the geometric correction of multiscene images was jointly carried out to achieve the geometric correction of satellite images at several control points in the use of the correlation between subscene overlapping regions.

3.2.3. Automatic Toning and Mosaic of Coastal-Zone Image. Image data toning processing was carried out by using the color-transferring algorithm to maintain the consistency of typical surface features. In the fuzzy classification method, the overlapping region calculation method and image multiscale segmentation method were used for the study on the color transferring algorithm to maintain the consistency of typical surface features. The coastal-zone remote sensing image was spliced by using the automatic matching-line generation image mosaic technology.

3.3. Data Mining

3.3.1. Coastline Extraction Based on Edge Detection. Edge detection methods were widely used for the automatic extraction of the water sideline so as to significantly reduce the quantity of data, remove the irrelevant information, and retain the important structural properties of the images. The methods consisted of Roberts Cross operator, Prewitt operator, Sobel operator, Canny operator, Kirsch operator, and Compass operator, in which the Canny operator can be regarded as the most common edge detection method. In order to compensate for the insufficient remote sensing image detection of complex surface features, edge features, or broken linear characteristics by the conventional Canny

operator, the chromatic aberration Canny operator was used instead to extract the coastline.

(1) Calculation of the Chromatic Aberration Amplitude by Means of Chromatic Aberration Canny Operator. In order to improve on deficiencies of the traditional Canny operator, an improved Canny operator was used based on the LAB color model as the main technical route in this paper, that is, the chromatic aberration Canny operator color difference canny (CDC) algorithm. This algorithm was able to detect the edge through the color gradient (chromatic aberration) between pixels and determine the pixel amplitude as per the calculation of the first-order partial derivative finite difference at the x-direction, y-direction, 135° direction, and 45° direction within the 8 adjacent zones of pixels. Its main advantages include high edge-positioning accuracy and effective noise suppression. It was able to accurately determine the color differences in accordance with the vision of human eye, and good results were achieved during practical application. The mathematical expression of the chromatic aberration Canny operator (CDC) is as follows.

Chromatic aberration in the x-direction:

$$D_x[i,j] = \mathrm{CD}\left(I[i+1,j], i-1, j\right), \tag{1}$$

chromatic aberration in the y-direction:

$$D_y[i,j] = \mathrm{CD}\left(I[i,j+1], i, j-1\right), \tag{2}$$

chromatic aberration in the 135° direction:

$$D_{135^\circ}[i,j] = \mathrm{CD}\left(I[i+1,j+1], I[i-1,j-1]\right), \tag{3}$$

chromatic aberration in the 45° direction:

$$D_{45^\circ}[i,j] = \mathrm{CD}\left(I[i-1,j+1], I[i+1,j-1]\right),$$

$$\mathrm{CD}(A,B) = \left[\left(L_A - L_B\right)^2 + \left(A_A - A_B\right)^2 + \left(B_A - B_B\right)^2\right]^{1/2}, \tag{4}$$

where $\mathrm{CD}(A,B)$ represents the chromatic aberration between the pixel point A and the pixel point B.

The pixel chromatic aberration amplitude and direction were calculated as per the coordinate conversion equation from the rectangular coordinates to the polar coordinates. The chromatic aberration amplitude was calculated as per the second-order norm equation as follows:

$$\mathrm{CDC}[i,j] = \sqrt{D_x[i,j]^2 + D_y[i,j]^2 + D_{135^\circ}[i,j]^2 + D_{45^\circ}[i,j]^2}. \tag{5}$$

Chromatic aberration direction:

$$\theta[i,j] = \arctan\left(\frac{D_y[i,j]}{D_x[i,j]}\right). \tag{6}$$

(2) Adaptive Calculated Dynamic Threshold. A whole image can be divided into several subimages. There might be a certain overlapping region between subimages so as to achieve the continuous profile and calculate the parameters for the proportion between the overlapping region and the subimages. The high and low threshold values of subimages were adaptively set according to the nonmaximum suppression results. The calculation equation should be as follows:

$$\tau_{\mathrm{high}} = (1-\beta)\tau_H + \beta\tau_h,$$
$$\tau_{\mathrm{Low}} = (1-\beta)\tau_L + \beta\tau_l, \tag{7}$$

where τ_H and τ_L represent the global high and low threshold values of the whole image, respectively; τ_h and τ_l represent the local high and low threshold values of subimage zone, respectively. $0 < \beta < 1$ represents the threshold adjustment rate. If $\beta = 0$, the whole image should not be adjusted. If $\beta = 1$, the whole image should be divided fully in accordance with the local features of the subimages.

(3) Boundary Tracking Generated Coastline. When the point with the chromatic aberration amplitude of a certain pixel in the whole image is greater than the high threshold value which was used as the starting point for tracking, the neighborhood pixel (other chromatic aberration amplitudes within the 8 pixel neighborhoods were greater than the high threshold value) should be set as the edge and used as the starting point for tracking. If there were no pixels (chromatic aberration amplitude was greater than the high threshold value) around the pixel point, the pixel (chromatic aberration amplitude was greater than the low threshold value) should be discovered within the 8 pixel neighborhoods and used as the starting point for tracking, up until the edge and the starting point cannot be found, and thus determined as the contour endpoint. The template was checked to determine whether the edge could be connected. In the event of any noise at the isolated point, it should be removed. Finally, the boundary was extracted for fine processing.

(4) Coastline Classification and Analysis Based on the Spectrum Feature Database. Based on the sample collection and field spectrum measurement of the remote sensing image of the sea area, the systematic research on the types and usage methods for development and exploitation of coastal zones, as well as the spectral features and image features in the common star-source data, the remote sensing classification criteria were presented for development and exploitation of coastal zones in this paper. Coastlines can be divided into natural coastlines, artificial coastlines, and estuary coastlines. The natural coastlines can be divided into bedrock coastline, silt-muddy coastline, sandy coastline, and biological coastline (including mangroves and coral reefs). The artificial coastlines can be divided into cofferdam coastline, marine reclamation land coastline, transportation engineering coastline, and protected coastline.

The object-oriented image feature analysis for the sea area was used in this study. Based on the main types and components of coastline development and exploitation, all kinds of target surface features were detected and extracted, including the spectrum, shape, texture, shadow, space, location, and related layout. The standard curve spectrum was obtained corresponding to the component. The spectral characteristics

Table 2: The analytical coastline length of China based on remote sensing from 2007 to 2011.

Year	Coastline length (m)	Natural coastline length (m)	Artificial coastline length (m)	Estuary coastline length (m)
2007	18,501,296	9,383,095	8,980,905	137,297
2008	18,515,830	9,730,865	8,651,800	133,165
2009	18,599,902	10,167,232	8,301,830	130,839
2010	18,644,768	10,378,360	8,136,654	129,754
2011	18,946,699	10,733,161	8,085,561	127,977

of each component were analyzed, the coastline category information extraction was achieved by using a fuzzy classification algorithm, and the length of Chinese coastline was obtained (as shown in Table 2) after the analysis of the spectrum feature database.

3.3.2. Validation of Coastline Extraction Results. After the completion of the coastline extraction algorithm model, it was necessary to test and validate the model extraction results. The validation methods of the model extraction result should include the point-by-point comparison method and the overall comparison method. Point-by-point inspection refers to the superposition of extracted results and actual conditions with point-by-point comparison as to its accuracy. Overall comparison should refer to the evaluation of extracted results by means of the indexes for the overall spatial pattern as discussed by Wu [23]. The Kappa coefficient was selected to quantitatively reflect the model operation for extracting the accuracy of coastline. This was mainly based on the different-stage coastline extraction results and the corresponding remote sensing image data to carry out the Kappa coefficient calculation of the adjacent grid map, to determine the average coefficient value of results at each stage, and to obtain the Kappa coefficient for the accuracy of coastline extraction results over the timeframe. The calculation equation should be as follows:

$$K = \frac{(P_o - P_C)}{(1 - P_C)}, \tag{8}$$

where P_o represents the percentage of consistent types of parts on the comparative grid map, that is, the observation consistency ratio; P_C represents the expectation consistency ratio, in which $P_o = s/n$, $P_C = (a1 * b1 + a0 * b0)/(n * n)$ (the total pixel number of grid: n; the pixel number of grid (1): $a1$; the pixel number of grid (0): $a0$; the pixel number of extracted grid (1): $b1$; the pixel number of extracted grid (0): $b0$; the pixel number of two grids with the equal corresponding pixel value: s). The different Kappa coefficients showed consistency to varying extents.

Shoreline information from the National Marine Data & Information Service was used to test the accuracy of the extracted shoreline. The calculation equation was as follows:

$$I = \frac{(L - L')}{L}, \tag{9}$$

where I is the accuracy coefficient, L is the actual length of shoreline, and L' is the extracted length of shoreline.

Table 3: The accuracy test of coastline extraction in China from 2007 to 2011.

Year	2007	2008	2009	2010	2011
Kappa coefficient	0.69	0.72	0.78	0.81	0.86
Accuracy coefficient	0.018	0.013	0.023	0.015	0.012

It can be seen from Table 3 that all the Kappa coefficients obtained in the previous five years were greater than 0.6 and that the accuracy coefficients obtained in the previous five years were less than 0.023. Previous studies about Kappa by Blackman and Koval [24] and Landis and Koch [25] have shown that the coastline extraction results achieved using the chromatic aberration Canny operator edge detection method should be consistent with the actual coastline heights, and the coastline extraction results in 2010 and 2011 should be almost completely consistent with the actual coastline heights.

3.3.3. Validation of Coastline Classification Results. In order to ensure the accuracy of the coastline classification results, the supervised classification method was used to complete the automatic classification of the coastline remote sensing images and to carry out sampling secondary interpretation and confirmation by means of the artificial visual interpretation method, which ensured the accuracy of coastline properties and the distribution boundary of the classification results. From the coastline extraction results over the timeframe, 2,000 samples were extracted for analysis. As per the probability calculation, the ratio of conforming samples was up to 95% or above, which met the required standards.

4. Analysis of Coastline Length Changes and Trends

4.1. Variation of Coastline Length over the Years. Variations in coastline length reflect the overall impact of coastline resource utilization, natural erosion and siltation, oceanic dynamics, and other factors. Changes and trends in the coastline length depend on the development speed and trends between natural coastline, artificial coastline, and estuary coastline.

It can be seen from Table 2 that over the past five years, except for when the coastline length decreased slightly in 2009, the total length of China's coastline increased. From 2007-2008 and 2009–2011, the rate of increase of China's

artificial coastline was greater than the rate of decrease of natural coastline and estuary coastline, so the overall coastline length increased. In 2009, the artificial coastline increased by 436,368 m, less than the rate of increase in 2008. The natural coastline and the estuary coastline decreased by 349,970 m, more than the rate of decrease in 2008. Analysis showed that the main reason for this was the formal implementation of China's Sea Reclamation Scheme and Management System in 2009. The area of sea reclamation (mostly including the cutoff type of reclamation) was increased by 6887.38 hectares over 2008 levels, while the coastline length slightly decreased in 2009. After 2010, when the coastal marine administrative departments at all levels began making considerable efforts on sea reclamation management and paid more attention to sea reclamation methods, which resulted in larger proportions of offshore type (island type) and convex barrier type reclamations, and combined with the enhancement of environmental protection and ecological awareness, the rate of decrease of the natural coastline and estuary coastline slowed, while the coastline length increased.

4.2. Coastline Changes and Trends in the Future

4.2.1. Prediction Methods. There was a big difference between the method of endpoint rate and the average rate as calculating the coastline change rates. It was known from the coastline change rate calculation method and the impact factor analysis made by Jingfu et al. [26] that a change in coastline length should be regarded as a periodic and oscillating change and that the rate calculation method should not be used for prediction of future variation trends. Having only five stages of data, the total change in length of the coastline should be predicted using the grey sequence prediction method.

4.2.2. Future Variation Trend. Using the grey prediction method, the coastline length values extracted from 2007–2011 were included into the grey sequence prediction model. The development factor ($a = -0.01$) and the grey action (b =18,357.74) should be calculated, and the grey prediction model GM (1, 1) of coastline should be obtained:

$$X(1)^{(t+1)} = \left[X(0)^{(1)} - 2637356.24\right] e^{(0.01t)} - 2637356.24, \tag{10}$$

where $X(1)^{(t+1)}$ represents the predicted accumulated value of $t + 1$ time; $X(0)^{(1)}$ represents the original value of starting time. The accumulative length of coastline (in 2008–2015) should be calculated as per (10). The former term should be subtracted by the later term, and the coastline prediction results were shown in Table 4. The variance ratio was 0.391 and the small error probability was 1, which was bigger than 0.95. It was shown that the accuracy of predicted results was consistent with the requirements. Therefore, the predicted value of the mainland coastline length obtained by the grey prediction method was in accordance with the actual conditions. The predicted results showed that the future mainland coastline length would increase and China's mainland coastline length would be up to about 19,471,983 m by 2015.

Table 4: China's coastline length prediction table.

Year	Coastline length (m)	Prediction length (m)	Residual (m)	Relative error
2007	18,501,296	18,501,296		
2008	18,515,830	18,549,380	−33550.33	0.00
2009	18,599,902	18,678,455	−78552.97	0.00
2010	18,644,768	18,808,427	−163659.73	0.01
2011	18,946,699	18,939,304	7394.30	0.00
2012		18,501,296		
2013		113,250,456		
2014		19,337,425		
2015		19,471,983		

5. Conclusions and Discussions

Through the data mining algorithm, the preprocessing of China's low-precision remote sensing images taken in the previous five years, the extraction of coastline lengths and types from 2007–2011, and the analysis of coastline changes and trends, the following conclusions can be made.

(1) The multisource imaging matching algorithm of a SIFT operator, the geometric correction of a few control point satellite images, the color-transferring algorithm for maintaining the consistency of typical surface features, toning processing technology, and the automatic matching-line generation image mosaic technology were used to preprocess the low-precision environmental disaster reduction satellite and CBERS-01 satellite images. Combined, these are all effective methods.

(2) The coastline can be extracted using the chromatic aberration Canny operator edge detection method. The Kappa coefficient should be greater than 0.6 throughout the calculation, which shows that the extraction results are consistent with the actual measured results. By analysing the coastline type using the spectrum feature database, trends identify a decreased natural coastline and estuary coastline, but an increased artificial coastline were found. These trends are consistent with the current utilization of the coastline area in China.

(3) The total length of China's coastline increased overall. This is despite the coastline decreasing slightly in 2009 due to the formal implementation of China's Sea Reclamation Scheme. By using the grey prediction method, China's coastline length is predicted to increase in the future, with the total length reaching 19,471,983 m by 2015.

With the continuous increase of artificial coastline length and related economic benefits, the balance of the marine

ecosystem will be disturbed, affecting the harmony between humans and nature. In order to ensure the sustainable exploitation and utilization of coastal resources and the reasonable adjustment of coastal development methods, the impact of coastline change trends on the marine ecological environment and the changes of spatial position should be studied further.

Acknowledgments

This work was financially supported by the Public Science and Technology Research Funds Projects of Ocean (no. 201005011) and Foundation of Key Laboratory of Sea Areas Management Technology SOA (201107).

References

[1] Z. Li and W. Jigang, "Coastline GPS real-time dynamic measurement technology and error impact," *Surveying and Mapping Sciences*, vol. 3, pp. 9–12, 2008.

[2] J. S. Lee, I. Jurkevich, P. Dewaele, P. Wambacq, and A. Oosterlinck, "Speckle filtering of synthetic aperture radar images: a review," *Remote Sensing Reviews*, vol. 8, no. 4, pp. 313–340, 1994.

[3] P. Manavalan, P. Sathyanath, and G. L. Rajegowda, "Digital image analysis techniques to estimate waterspread for capacity evaluations of reservoirs," *Photogrammetric Engineering and Remote Sensing*, vol. 59, no. 9, pp. 1389–1395, 1993.

[4] R. Holman, J. Stanley, and T. Özkan-Haller, "Applying video sensor networks to nearshore environment monitoring," *IEEE Pervasive Computing*, vol. 2, no. 4, pp. 14–21, 2003.

[5] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, no. 1–4, pp. 259–268, 1992.

[6] T. F. Chan, S. Osher, and J. Shen, "The digital TV filter and nonlinear denoising," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 231–241, 2001.

[7] D. L. Donoho, "Orthonormal ridgelets and linear singularities," *SIAM Journal on Mathematical Analysis*, vol. 31, no. 5, pp. 1062–1099, 2000.

[8] M. Xiaofeng, Z. Dongzhi, Z. Fengshou et al., "Study on the coastline satellite remote sensing extraction method," *Remote Sensing Technology and Application*, vol. 4, pp. 575–580, 2007.

[9] Z. Chaoyang, *Study on the Remote Sensing Image Coastline Extraction and Its Change Detection Technology*, Surveying and Mapping Institute, PLA Information Engineering University, Zhengzhou, China, 2006.

[10] D. Jiang, Y. Huang, D. Zhuang, Y. Zhu, X. Xu et al., "A simple semi-automatic approach for land cover classification from multispectral remote sensing imagery," *PLoS ONE*, vol. 7, no. 9, Article ID e45889, 2012.

[11] D. G. Stavrakoudis, J. B. Theocharis, and G. C. Zalidis, "A boosted genetic fuzzy classifier for land cover classification of remote sensing imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 66, no. 4, pp. 529–544, 2011.

[12] Z. Xiaocheng and W. Xiaoqin, "Study on the remote sensing image data mining," *Remote Sensing Information*, vol. 3, pp. 58–62, 2005.

[13] G. Dengke, *Study on the GIS-Based Spatial Data Method*, Xi'an University of Electronic Science and Technology, 2010.

[14] L. Deren, W. Shuliang, and L. Deyi, *Spatial Data Mining Theory and Application*, Science and Technology Press, Beijing, China, 2006.

[15] W. Zhanquan, *Study on the Key Spatial Data Mining Technologies Based on the Geographic Information System*, Zhejiang University, 2005.

[16] D. Kaichang, L. Deren, and L. Deyi, "Cloud theory and its application in the spatial data mining and knowledge discovery," *Chinese Journal of Image and Graphics*, vol. 4, no. 11, pp. 924–929, 1999.

[17] L. Longshu, N. Zhiwei, and L. Cheng, "Research and practice on the spatial attribute data mining based on the rough set," *Journal of System Simulation*, vol. 14, no. 12, pp. 1702–1705, 2002.

[18] L. Aihua, Y. Jianwei, M. Tushu et al., "Research and system implementation of coastline change trend prediction method," *Surveying and Mapping Sciences*, vol. 34, no. 4, pp. 109–110, 2009.

[19] L. Deren and S. Juliang, "The wavelet and its application in image edge detection," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 22, no. 2, pp. 4–111120, 1993.

[20] Z. Guobao, Y. Hua, and C. Weinan, "Multi-scale edge extraction based on the biorthogonal wavelet transform," *Chinese Journal of Image and Graphics*, vol. 3, no. 8, pp. 651–654, 1998.

[21] D. Tao and Z. Bin, "Study on the coastline positions determined by remote sensing image based on the analysis of wavelet technology," *Marine Science*, vol. 4, pp. 19–21, 1999.

[22] M. Kantardzic, S. Siqing, C. Yin et al., *Data Mining: Concept, Model, Method and Algorithm*, Tsinghua University Press, 2003.

[23] F. Wu, "Calibration of stochastic cellular automata: the application to rural-urban land conversions," *International Journal of Geographical Information Science*, vol. 16, no. 8, pp. 795–818, 2002.

[24] N. Blackman and J. Koval, "Interval estimation for Cohen's Kappa as a measure of agreement," *Statistics in Medicine*, vol. 19, no. 5, pp. 723–741, 2000.

[25] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 4, pp. 671–679, 1977.

[26] L. Jingfu, Y. Ping, R. Jingco et al., "Selection of methods for calculating shoreline change rate and analysis of affecting factor," *Advances in Marine Science*, vol. 21, no. 1, pp. 52–53.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 926519, 6 pages
http://dx.doi.org/10.1155/2013/926519

Research Article

Analysis of Similarity/Dissimilarity of DNA Sequences Based on Chaos Game Representation

Wei Deng[1,2] and Yihui Luan[1]

[1] *School of Mathematics, Shandong University, Jinan 250100, China*
[2] *School of Science, Shandong Jianzhu University, Jinan 250101, China*

Correspondence should be addressed to Yihui Luan; yhluan@sdu.edu.cn

Received 20 November 2012; Revised 27 January 2013; Accepted 8 February 2013

Academic Editor: Yong Zhang

The Chaos Game is an algorithm that can allow one to produce pictures of fractal structures. Considering that the four bases A, G, C, and T of DNA sequences can be divided into three classes according to their chemical structure, we propose different kinds of CGR-walk sequences. Based on CGR coordinates of random sequences, we introduce some invariants for the DNA primary sequences. As an application, we can make the examination of similarity/dissimilarity among the first exon of β-globin gene of different species. The results indicate that our method is efficient and can get more biological information.

1. Introduction

A DNA sequence is comprised of four different nucleotides: adenine (A), cytosine (C), guanine (G), and thymine (T). Since the DNA molecule contains plentiful biological, physical, and chemical information, it has become very important to analyze DNA sequences statistically. Now the nucleotides stored in GenBank have exceeded hundreds of millions of bases and the increasing rate is considerably rapid. Therefore, biologists, physicists, mathematicians, and computer specialists have adopted different techniques to research DNA sequences in recent years, including the statistical methods and some mapping rules of the bases.

A great number of studies have proposed to convert the DNA sequences into digital sequences before downstream analysis. There are many statistical methods such as random walk, lévy-walk, entropy near method, root-mean-square fluctuation, wavelet transform and Fourier transform, and so forth, [1–12], which can be used as effective tools to process the DNA sequences. One-dimensional DNA walk was first proposed by Peng et al. [1]. Bai et al. [13] later discussed the representation of DNA primary sequences by the same walk. Meanwhile, some investigators proposed several kinds of graphical representation of DNA sequences from different perspectives. For example, G-curve and H-curve were first proposed by Hamori and Ruskin in 1983 [14]. R. Zhang and C. T. Zhang [15] considered a DNA primary sequence termed as Z-curve. Several researchers in their recent studies have outlined different kinds of graphical representation of DNA sequences based on 2D [16–21], 3D [22–25], 4D [26], 5D [27], and 6D [28] spaces. We here need to stress Chaos Game Representation (CGR) which was proposed as a scale-independent representation for genomic sequences by Jeffrey [3] in 1990. Gao and Xu [29] pointed out that the CGR-walk model can easily generate a model sequence and can be fitted with a long-memory ARFIMA (p, d, q) model reasonably. However, they treated the four bases equally and ignored the hidden chemical classification of nucleotides.

Motivated by the above work, we consider in this paper different classifications of the four bases according to their chemical structure and the strength of the hydrogen bond, that is, purine R = {A, G} and pyrimidine Y = {C, T}; amino group M = {A, C} and keto group K = {G, T}; weak H-bonds W = {A, T} and strong H-bonds S = {G, C}. Then we give three kinds of mapping from the four bases A, C, G, and T to the continuous space and reconstruct CGR-walk sequences based on CGR coordinates. So we can convert a DNA sequence into a random numeric sequence, then select some numerical characterizations of the random sequence as new invariants for the DNA sequence. As an application,

we make a comparison of the similarity and dissimilarity of the first exon of β-globin gene sequences derived from nine species.

2. CGR-Walk Based on Three kinds of Classification and Primary Sequences

2.1. The CGR *Space Proposed by Jeffrey.* During the past several years, a new field of physics has developed, known as "nonlinear dynamics," "chaotic dynamical systems," or simply "chaos." In fact, the technique of CGR, formally an iterative mapping, can be traced further back to the foundation of statistical mechanics, in particular, to chaos theory [2]. Based on the technique from chaotic dynamics, CGR produces a picture of gene sequence which displays both local and global patterns. The Chaos Game is an algorithm which allows one to produce pictures of fractal structures. Mathematically, it is described by an iterated function system (IFS).

The CGR space can be viewed as a continuous reference system, where all possible sequences of any length occupy a unique position. And the position is produced by the four possible nucleotides, which are treated as vertices of a binary square. So it is planar. Since a genetic sequence can be treated formally as a string composed of the four letters "A," "C," "G," and "T" (or "U"), the binary CGR vertices are assigned to the four nucleotides as A = (0, 0), G = (1, 1), C = (0, 1), T = (1, 0). The CGR coordinates are calculated iteratively by moving a pointer to half the distance between the previous position and the current binary representation. For example, if a "G," is the next base, then a point is plotted half way between the previous point and the "G" corner. The iterated function can be given by

$$\mathrm{CGR}_i = \mathrm{CGR}_{i-1} - 0.5\left(\mathrm{CGR}_{i-1} - g_i\right), \tag{1}$$

where

$$i = 1, \ldots, n_{\mathrm{G}}; \quad \mathrm{CGR}_0 = (0.5, 0.5); \quad g_i \in \{A, G, C, T\}. \tag{2}$$

We take the first 6 bases of the sequence of human β-globin in Table 1 as an example and present the above procedure in Figure 1.

2.2. The Newly Proposed CGR *Space.* The aforementioned work treats the four nucleic acid bases equally. In this paper, however, we take the chemical structures of the four nucleic acid bases into consideration and make adjustments to the classification based on the elements of the minor diagonal. In the CGR space proposed by Jeffrey, the elements of the minor diagonal are purine R = {A, G} and the leading diagonal elements are pyrimidine Y = {C, T}. Considering amino group M = {A, C} and keto group K = {G, T}, we get the second CGR space as shown in Figure 2. In the same way, according to the strength of the hydrogen bond, the bases can also be classified into weak H-bonds W = {A, T} and strong H-bonds S = {G, C}, so the third kind of CGR space is obtained in Figure 3.

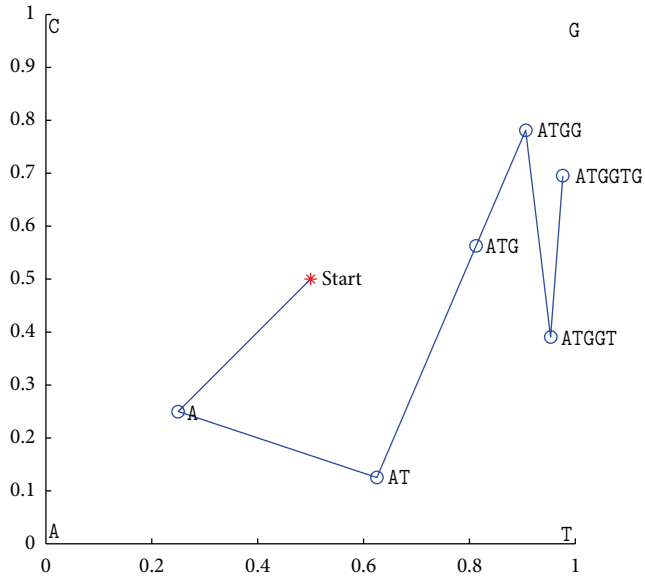


Figure 1: CGR-RY of the first 6 bases of exon-1 of human β-globin: ATGGTG.

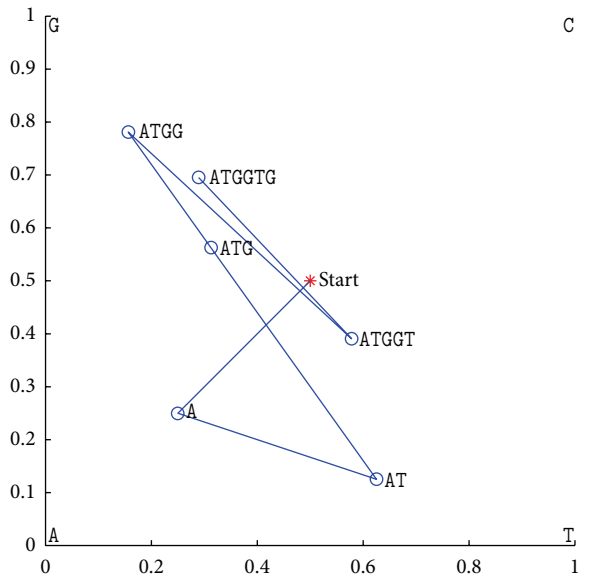


Figure 2: CGR-MK of the first 6 bases of exon-1 of human β-globin: ATGGTG.

2.3. CGR-*Walk Digital Sequence.* Now we can obtain map relationships between DNA sequences and the CGR coordinates in a right-angled plane. For a DNA sequence, we define an equation as follows:

$$z_i = x_i + y_i, \tag{3}$$

where x_i and y_i are the x-coordinate and y-coordinate of CGR, respectively. Then we can get a data sequence $\{z_i : i = 1, 2, \ldots, N\}$. In this way, we convert a DNA sequence into a random walk sequence under three different patterns. Consistent with the above three figures, we call them CGR-RY-, CGR-MK-, and CGR-WS-walk sequences, respectively.

Table 1: The coding sequences of the first exon of β-globin gene of different species.

Species	Coding sequence
Human	ATGGTGCACCTGACTCCTGAGGAGAAGTCTGCCGT TACTGCCCTGTGGGGCAAGGTGAACGTGGATTAAG TTGGTGGTGAGGCCCTGGGCAG
Goat	ATGCTGACTGCTGAGGAGAAGGCTGCCGTCACCGG CTTCTGGGGCAAGGTGAAAGTGGATGAAGTTGGTG CTGAGGCCCTGGGCAG
Opossum	ATGGTGCACTTGACTTCTGAGGAGAAGAACTGCA TCACTACCATCTGGTCTAAGGTGCAGGTTGACCA GACTGGTGGTGAGGCCCTTGGCAG
Gallus	ATGGTGCACTGGACTGCTGAGGAGAAGCAGCTCAT CACCGGCCTCTGGGGGAAGGTCAATGTGGCCGAAT GTGGGGCCGAAGCCCTGGCCAG
Lemur	ATGACTTTGCTGAGTGCTGAGGAGAATGCTCATGT CACCTCTCTGTGGGGCAAGGTGGATGTAGAGAAAG TTGGTGGCGAGGCCTTGGGCAG
Mouse	ATGGTTGCACCTGACTGATGCTGAGAAGTCTGCTG TCTCTTGCCTGTGGGCAAAGGTGAACCCCGATGAA GTTGGTGGTGAGGCCCTGGGCAGG
Rabbit	ATGGTGCATCTGTCCAGTGAGGAGAAGTCTGCGGT CACTGCCCTGTGGGGCAAGGTGAATGTGGAAGAAG TTGGTGGTGAGGCCCTGGGC
Rat	ATGGTGCACCTAACTGATGCTGAGAAGGCTACTGT TAGTGGCCTGTGGGGAAAGGTGAACCCTGATAATG TTGGCGCTGAGGCCCTGGGCAG
Gorilla	ATGGTGCACCTGACTCCTGAGGAGAAGTCTGCCGT TACTGCCCTGTGGGGCAAGGTGAACGTGGATGAAG TTGGTGGTGAGGCCCTGGGCAGG

Table 2: Hurst exponent of the CGR-walk sequence $\{X_n\}$ of the nine species in Table 1.

	Human	Goat	Opossum	Gallus	Lemur	Mouse	Rabbit	Rat	Gorilla
$H(X_n^{\mathrm{RY}})$	0.445	0.5024	0.6536	0.5075	0.5016	0.538	0.429	0.5791	0.4698
$H(X_n^{\mathrm{MK}})$	0.7452	0.7853	0.6547	0.7212	0.7487	0.7094	0.8099	0.5237	0.7467
$H(X_n^{\mathrm{WS}})$	0.641	0.6894	0.6292	0.5756	0.6753	0.8118	0.615	0.7255	0.6302

3. Numerical Characterization of DNA Sequences

Researchers from computer science and mathematics have been attracted to study the comparison of DNA sequences. As pointed out in references [13, 16–28], some related work has made progress.

Now, we may represent a DNA sequence by a random numerical sequence based on CGR-walk technique. Gao and Xu [29] also substantially corroborated the results that long-range correlations are uncovered remarkably in the data. In this paper, we explore the tendency of a series of data by calculating the hurst exponent [30]. And some work has been done to study the relation between long-range correlation and hurst exponent [31]. In order to numerically characterize a DNA sequence given by the CGR, we treat the hurst exponent as the efficient invariant that is sensitive to this kind of graphical representation.

Because a DNA sequence can be regarded as an ordered set of alphabet $\mathcal{N} = (\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T})$, we represent a DNA sequence as a finite set with N elements, denoted as $[i] := \{1, 2, \ldots, N\}$. For any time series $\{u_i\}_{i=1}^N$, one can define several quantities as follows [30]:

(i) the partial mean

$$\langle u \rangle_n = \frac{1}{n} \sum_{i=1}^{n} u_i, \quad 2 \le n \le N, \tag{4}$$

Table 3: Mean square deviations of the CGR-walk sequence $\{X_n\}$ of the nine species of in Table 1.

	Human	Goat	Opossum	Gallus	Lemur	Mouse	Rabbit	Rat	Gorilla
$D(X_n^{\mathrm{RY}})$	0.3979	0.3927	0.3998	0.4192	0.4054	0.3866	0.4060	0.4266	0.3921
$D(X_n^{\mathrm{MK}})$	0.3858	0.3949	0.3500	0.3940	0.3636	0.3871	0.3866	0.3908	0.3838
$D(X_n^{\mathrm{WS}})$	0.3590	0.3724	0.3907	0.3411	0.4010	0.3912	0.3742	0.3713	0.3574

Table 4: Similarity/dissimilarity table for the nine DNA sequences in Table 1 based on Euclidean distance between the 3-component vectors in Table 2.

Species	Human	Goat	Opossum	Gallus	Lemur	Mouse	Rabbit	Rat	Gorilla
Human	0	0.0851	0.2277	0.0936	0.0663	0.1978	0.0715	0.2724	0.0271
Goat		0	0.2087	0.1307	0.0392	0.1484	0.1074	0.2750	0.0778
Opossum			0	0.1692	0.1846	0.2229	0.2734	0.1788	0.2056
Gallus				0	0.1036	0.2385	0.1248	0.2581	0.0711
Lemur					0	0.1467	0.1125	0.2432	0.0552
Mouse						0	0.2464	0.2089	0.1976
Rabbit							0	0.3416	0.0767
Rat								0	0.2660
Gorilla									0

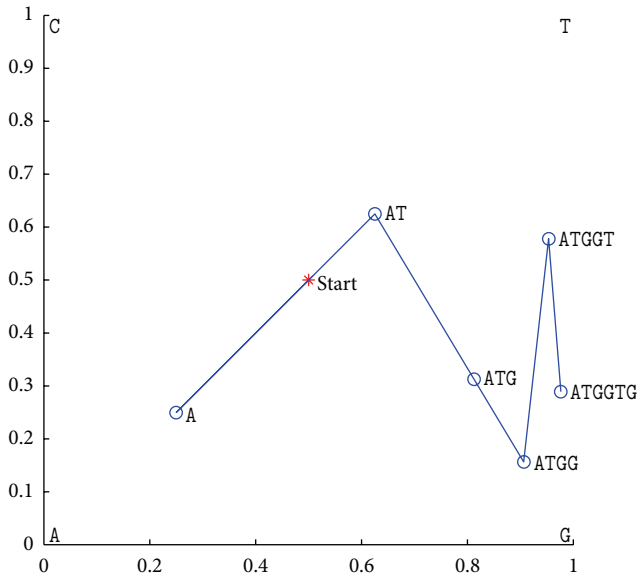


Figure 3: CGR-WS of the first 6 bases of exon-1 of human β-globin: ATGGTG.

(ii) the partial difference

$$u(i,n) = \sum_{i=1}^{n} \left(u_i - \langle u \rangle_n\right), \quad 2 \le n \le N, \tag{5}$$

(iii) the difference

$$R(n) = \max_{1\le i\le n} \{u(i,n)\} - \min_{1\le i\le n} \{u(i,n)\}, \quad 2 \le n \le N, \tag{6}$$

(iv) and the standard deviation

$$S(n) = \left[\frac{1}{n}\sum_{i=1}^{n}\left(u_i - \langle u \rangle_n\right)^2\right]^{1/2}, \quad 2 \le n \le N. \tag{7}$$

Hurst exponent is found to obey the relation:

$$\frac{R(n)}{S(n)} \sim \left(\frac{n}{2}\right)^H, \tag{8}$$

where H is called the hurst exponent.

So we can compute the hurst exponent of RY-, MK- and WS-CGR-walk sequences and characterize the coding sequences of the first exon of β-globin gene of the nine species in Table 1. The results are listed in Table 2.

Besides, there are other numerical characterizations of random sequences, such as the mean, variance, mean square deviation, and so on. Here we choose the mean square deviation of CGR-walk sequence as follows:

$$D\left(X_i^k\right) = \left[\frac{1}{N}\sum_{i=1}^{N}\left(X_i^k - \mu_{X_i^k}\right)^2\right]^{1/2}. \tag{9}$$

In (9) k means the classification of RY-, MK-, and WS-sequences, and $\mu_{X_i^k}$ is the mean [13]. We then present the mean square deviations of three kinds of the CGR-walk sequences $\{X_i\}$ in Table 3.

4. Similarity and Dissimilarity among the Coding Sequences of the First Exon of β-Globin Gene of Different Nine Species

Here we construct the three-component vectors in this way, whose components, respectively, are values of hurst exponent and mean square deviation. The analysis of similarity/dissimilarity among DNA sequences represented by the three-component vectors is based on the assumption that two DNA sequences are similar if the corresponding vectors point to one direction in the 3D space. Alternatively we can investigate the similarity among the vectors by calculating

Table 5: Similarity/dissimilarity table for the nine DNA sequences in Table 1 based on Euclidean distance between the 3-component vectors in Table 3.

Species	Human	Goat	Opossum	Gallus	Lemur	Mouse	Rabbit	Rat	Gorilla
Human	0	0.0171	0.0479	0.0290	0.0481	0.0342	0.0173	0.0317	0.0063
Goat		0	0.0490	0.0410	0.0442	0.0212	0.0157	0.0342	0.0187
Opossum			0	0.0691	0.0180	0.0394	0.0407	0.0526	0.0481
Gallus				0	0.0686	0.0602	0.0364	0.0313	0.0333
Lemur					0	0.0317	0.0353	0.0455	0.0499
Mouse						0	0.0258	0.0449	0.0344
Rabbit							0	0.0213	0.0220
Rat								0	0.0380
Gorilla									0

the Euclidean distance between their end points. Apparently, the smaller the Euclidean distance is, the more similar the two corresponding DNA sequences are.

In Tables 4 and 5, we list the values of Euclidean distances between the 3-component vectors separately including hurst exponent and mean square deviation. We observe that the smallest entry is always the human-gorilla pair. Furthermore, the largest entries are associated with these rows belonging to opossum (the most remote species from the remaining mammals) and gallus (the only nonmammalian representative). We believe that these results are not accidental, and they coincide with other results in [13, 16–28].

5. Conclusion

DNA sequences play an important role in modern biological research because all the information of the hereditary and species evolution is contained in these macromolecules. How to gain more information from these DNA sequences is still a very challenging question. Description, comparison, and similarity analysis of DNA sequences still occupy important positions.

In this paper, we first construct three kinds of CGR spaces according to the elements of the minor diagonal because the four bases can be classified into R-Y, M-K, and W-S according to their chemical structures. Then we describe a DNA sequence by CGR-walk and convert it to a digital sequence. And we outline some efficient invariants of DNA sequences. As an application, we compare the similarity/dissimilarity of exon-1 of β-globin genes for nine species. From the above tables, we can conclude that the results we got are consistent with known evolutionary facts. Therefore, the method proposed in the paper is visual and efficient.

On one hand, our work can be treated as an effective application of CGR. On the other hand, our method is a valid supplement to graphical representation of DNA sequences. In comparison with other graphical representations of biological sequences, our approach has the following advantages.

(1) Our graphical representation based on CGR considers the chemical structure classification of the nucleotides and thus may provide more biological information.

(2) It provides a more simple way of viewing, sorting, and comparing various gene structures, even for longer DNA sequences.

(3) Our graph is more sensitive, so it can numerically characterize the DNA sequences in a more exact way.

Acknowledgments

The authors thank all the anonymous reviewers for their valuable suggestions and support. This research is supported by the National Science Foundation of China under Grants 11071146 and 10921101.

References

[1] C. K. Peng, S. V. Buldyrev, A. L. Goldberger et al., "Long-range correlations in nucleotide sequences," *Nature*, vol. 356, no. 6365, pp. 168–170, 1992.

[2] J. S. Almeida, J. A. Carriço, A. Maretzek, P. A. Noble, and M. Fletcher, "Analysis of genomic sequences by Chaos Game Representation," *Bioinformatics*, vol. 17, no. 5, pp. 429–437, 2001.

[3] H. J. Jeffrey, "Chaos game representation of gene structure," *Nucleic Acids Research*, vol. 18, no. 8, pp. 2163–2170, 1990.

[4] S. V. Buldyrev, N. V. Dokholyan, A. L. Goldberger et al., "Analysis of DNA sequences using methods of statistical physics," *Physica A*, vol. 249, no. 1–4, pp. 430–438, 1998.

[5] S. V. Buldyrev, A. L. Goldberger, S. Havlin, C. K. Peng, H. E. Stanley, and G. M. Visvanathan, *Fractals in Biology and Medicine: from DNA To the Heartbeat*, Springer, Berlin, Germany, 1994.

[6] S. V. Buldyrev, A. L. Goldberger, S. Havlin, C.K. Peng, M. Simons, and H. E. Stanley, "Generalized Lévy-walk model for DNA nucleotide sequences," *Physical Eeview E*, vol. 47, no. 6, pp. 4514–4523, 1993.

[7] S. V. Buldyrev, A. L. Goldberger, S. Havlin et al., "Long-range correlation properties of coding and noncoding DNA sequences: GenBank analysis," *Physical Review E*, vol. 51, no. 5, pp. 5084–5091, 1995.

[8] G. Dodin, P. Vandergheynst, P. Levoir, C. Cordier, and L. Marcourt, "Fourier and wavelet transform analysis, a tool for visualizing regular patterns in DNA sequences," *Journal of Theoretical Biology*, vol. 206, no. 3, pp. 323–326, 2000.

[9] A. A. Tsonis, P. Kumar, and J. B. Elsneretal, "Navelet analysis of DNA sequences," *Physical Review E*, vol. 53, pp. 1828–1834, 1996.

[10] L. F. Luo, L. Tsai, and Y. M. Zhou, "Informational parameters of nucleic acid and molecular evolution," *Journal of Theoretical Biology*, vol. 130, no. 3, pp. 351–361, 1988.

[11] L. F. Luo and L. Tsai, "Fractal dimension of nucleic acid and its relation to evolutionary level," *Chemical Physics Letters*, vol. 5, pp. 421–424, 1988.

[12] A. Arneodo, Y. D'Aubenton-Carafa, E. Bacry, P. V. Graves, J. F. Muzy, and C. Thermes, "Wavelet based fractal analysis of DNA sequences," *Physica D*, vol. 96, no. 1–4, pp. 291–320, 1996.

[13] F.-L. Bai, Y.-Z. Liu, and T.-M. Wang, "A representation of DNA primary sequences by random walk," *Mathematical Biosciences*, vol. 209, no. 1, pp. 282–291, 2007.

[14] E. Hamori and J. Ruskin, "H curves, a novel method of representation of nucleotide series especially suited for long DNA sequences," *The Journal of Biological Chemistry*, vol. 258, no. 2, pp. 1318–1327, 1983.

[15] R. Zhang and C. T. Zhang, "Z-curve, an intuitive tool for visualizing and analyzing the DNA sequences," *Journal of Biomolecular Structure & Dynamics*, vol. 11, pp. 767–782, 1994.

[16] X. F. Guo, M. Randic, and S. C. Basak, "A novel 2-D graphical representation of DNA sequences of low degeneracy," *Chemical Physics Letters*, vol. 350, no. 1-2, pp. 106–112, 2001.

[17] M. Randic, "Graphical representations of DNA as 2-D map," *Chemical Physics Letters*, vol. 386, pp. 468–471, 2004.

[18] G. H. Huang, B. Liao, Y. F. Liu, and Z. B. Liu, "HCL curve: a novel 2D graphical representation for DNA sequences," *Chemical Physics Letters*, vol. 462, pp. 129–132, 2008.

[19] A. Nandy and P. Nandy, "On the uniqueness of quantitative DNA difference descriptions in 2D graphical representation models," *Chemical Physics Letters*, vol. 368, no. 1-2, pp. 102–107, 2003.

[20] M. Randic, M. Vracko, N. Lers, and D. Plavsic, "Analysis of similarity/dissimilarity of DNA sequences based on novel 2-D graphical representation," *Chemical Physics Letters*, vol. 371, pp. 202–207, 2003.

[21] Y. Yao and T. Wang, "A class of new 2-D graphical represent ation of DNA sequences and their application," *Chemical Physics Letters*, vol. 398, pp. 318–323, 2004.

[22] B. Liao and K. Ding, "A 3D graphical representation of DNA sequences and its application," *Theoretical Computer Science*, vol. 358, no. 1, pp. 56–64, 2006.

[23] Z. Cao, B. Liao, and R. Li, "A group of 3D graphical representation of DNA sequences based on dual nucleotides," *International Journal of Quantum Chemistry*, vol. 108, no. 9, pp. 1485–1490, 2008.

[24] Y. Huang and T. Wang, "New graphical representation of a DNA sequence based on the ordered dinucleotides and its application to sequence analysis," *International Journal of Quantum Chemistry*, vol. 112, pp. 1746–1757, 2012.

[25] B. Liao, Y. Zhang, K. Ding, and T. M. Wang, "Analysis of similarity/dissimilarity of DNA sequences based on a condensed curve representation," *Journal of Molecular Structure*, vol. 717, no. 1–3, pp. 199–203, 2005.

[26] R. Chi and K. Ding, "Novel 4D numerical representation of DNA sequences," *Chemical Physics Letters*, vol. 407, no. 1-3, pp. 63–67, 2005.

[27] B. Liao, R. Li, W. Zhu, and X. Xiang, "On the similarity of DNA primary sequences based on 5-D representation," *Journal of Mathematical Chemistry*, vol. 42, no. 1, pp. 47–57, 2007.

[28] B. Liao and T. M. Wang, "Analysis of similarity/dissimilarity of DNA sequences based on nonoverlapping triplets of nucleotide bases," *Journal of Chemical Information and Computer Sciences*, vol. 44, no. 5, pp. 1666–1670, 2004.

[29] J. Gao and Z. Y. Xu, "Chaos game representation (CGR)-walk model for DNA sequences," *Chinese Physics B*, vol. 18, no. 1, pp. 370–376, 2009.

[30] Z. G. Yu and V. Anh, "Time series model based on global structure of complete genome," *Chaos, Solitons and Fractals*, vol. 12, no. 10, pp. 1827–1834, 2001.

[31] L. L. Jiang, Z. Y. Xu, and J. Gao, "Multifractal hurst analysis of DNA sequence," *China Journal of Bioinformatics*, vol. 7, no. 4, pp. 264–267, 2009.

Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 528678, 7 pages
http://dx.doi.org/10.1155/2013/528678

Research Article

Crude Oil Price Prediction Based on a Dynamic Correcting Support Vector Regression Machine

Li Shu-rong and Ge Yu-lei

College of Information and Control Engineering, China University of Petroleum, Qingdao, Shandong 266580, China

Correspondence should be addressed to Li Shu-rong; lishuron@upc.edu.cn

Received 10 December 2012; Accepted 28 January 2013

Academic Editor: Fuding Xie

A new accurate method on predicting crude oil price is presented, which is based on ε-support vector regression (ε-SVR) machine with dynamic correction factor correcting forecasting errors. We also propose the hybrid RNA genetic algorithm (HRGA) with the position displacement idea of bare bones particle swarm optimization (PSO) changing the mutation operator. The validity of the algorithm is tested by using three benchmark functions. From the comparison of the results obtained by using HRGA and standard RNA genetic algorithm (RGA), respectively, the accuracy of HRGA is much better than that of RGA. In the end, to make the forecasting result more accurate, the HRGA is applied to the optimize parameters of ε-SVR. The predicting result is very good. The method proposed in this paper can be easily used to predict crude oil price in our life.

1. Introduction

In recent years, crude oil prices have experienced four jumps and two slumps. The fluctuation of crude oil price adds more changes to the development of world economy. Grasping the change of oil price can provide guidance for economic development [1]. Therefore, it is very important to predict the crude oil price accurately.

The predicting methods can be divided into two aspects. One is from the qualitative angle [2]; the other is from quantitative angle, such as econometric model and statistical model [3, 4]. And the latter method is adopted by most scholars. But it is a difficult job to predict crude oil price, since the price is nonlinear and nonstationary time series [5]. The traditional predicting methods such as $AR(p)$ model, $MA(q)$ model, and $ARMA(p,q)$ model, base on linear model. They are only suitable for linear prediction and cannot be applied to model and predict nonlinear time series [6]. Wang got the predicting model by using time series and artificial neural network in 2005 [7], Xie proposed a new method for crude oil price forecasting based on support vector machine (SVM) in 2006 [8], Mohammad proposed a hybrid artificial intelligence model for crude oil price forecasting by means of feed-forward neural networks and genetic algorithm in 2007 [9], and Guo proposed a hybrid time series model on the base of GMTD model in 2010 [10]. The experimental results tell us that the prediction accuracy of these methods is better than traditional models. But the results is still existing biggish errors especially when the crude oil price is fluctuating violently.

Neural network technique provides a favorable tool for nonlinear time series forecasting. But the predictive ability of conventional neural network is low, because of the problems such as the local minimum, over learning, and the lacking of theoretical direction for selecting the hidden layer nodes. The SVM was proposed in the 1990s [11]; it can get the optimal results on the basis of the current information. The basic idea of SVM is that it fits the sample capacity of functions on the basis of regulating the upper bound of the minimum VC dimension, which also means the numbers of support vector. Compared with neural network [12, 13], SVM has strong generalization ability of learning small samples and with the inferior dependence on quantity. But the prediction performance of SVM is very sensible to parameter selection. On the other hand, the research on parameter optimization of SVM is very few at the moment. The parameters are usually determined on experience or trial method. In this way, if the parameters are not suitably chosen, the SVM will lead to poor

prediction performance. So, it is important to find one good method to get the optimal parameters of SVM.

In this paper, an ε-support vector regression machine with dynamic correction factor is proposed. And a novel hybrid RNA genetic algorithm (HRGA) is proposed to obtain the optimal parameters for a SVM. The HRGA is from the development of biological science and technology; the structure and information of RNA molecular are known profoundly. To improve the optimal performance of genetic algorithm, one genetic algorithm which bases on coding and biological molecular operation has been widely concerned [14]. This method improves the search efficiency and optimization performance through coding the individuals into biological molecules by use of bases [15, 16]. The appropriate mutation operator can improve the population diversity and prevent premature. While the mutation operator of classical RNA genetic algorithm (RGA) is fixed, so we need to find a suitable method to determine the mutation operator. In 2003, Kennedy did some improvement on particle swarm optimization (PSO) and proposed the bare bones particle swarm algorithm [17].

In the proposed HRGA, the position displacement idea of bare bones PSO is applied to change the mutation operator. The nucleotide base encoding, RNA recoding operation, and protein folding operation are reserved in the new algorithm. Thus, the strong global search capability is kept. At the same time, to make sure of the directivity of local searching, the optimal experience of the whole population and the historical experience of individuals are used. The convergence speed and solution precision are improved. Furthermore, to test the validity of HRGA, three benchmark functions are adopted. The mean value of optimum of HRGA is smaller than that of traditional RNA genetic algorithm.

Once the support vector regression machine is designed optimally, it can be used to predict crude oil price. Dynamic correction factor is brought in to improve the predictive effect and can strengthen the robustness of systems. In order to test the performance of the proposed predicting method, we provided the predicting results by using a back propagation neural network and a traditional support vector regression machine which are also improved with dynamic correction factor [7, 8]. The results show that our predicting method obtains greater accuracy than that of the other two in this paper.

The paper is organized as follows. Section 2 discusses the support vector regression machine with dynamic correction factor. Section 3 presents HRGA based on bare bones PSO, and some testing examples are applied to verify the effectiveness of the algorithm. Section 4 applies the dynamic correcting ε -SVR to predict the crude oil price. Section 5 concludes the paper.

2. Support Vector Regression Machine with Dynamic Correction Factor

Consider the training sample set (x_i, y_i), $i = 1, 2, \ldots, n$, $x_i \in R^n$, as the input variable and $y_i \in R$ as the output variable.

The basic idea of SVM is to find a nonlinear mapping ϕ from input space to output space [18–20]. Data x is mapped to a high-dimensional characteristic space F on the basis of the nonlinear mapping. The estimating function of linear regression in characteristic space F is as follows:

$$f(x) = [\omega \times \varphi(x)] + b, \quad \phi : R^n \longrightarrow F, \omega \in F, \tag{1}$$

where b denotes threshold value.

Function approximation problem is equal to the following function:

$$R_{\text{reg}}(f) = R_{\text{emp}}(f) + \lambda\|\omega\|^2 = \sum_{i=1}^{s} C(e_i) + \lambda\|\omega\|^2, \tag{2}$$

where $R_{\text{reg}}(f)$ denotes the objective function, $R_{\text{emp}}(f)$ denotes the empirical risk function, s denotes the sample quantity, λ denotes adjusting constant, and C denotes the error penalty factor. $\|\omega\|^2$ reflects the complexity of f in the high-dimensional characteristic space.

Since linear ε insensitive loss function has better sparsity, we can get the following loss function:

$$|y - f(x)|_\varepsilon = \max\{0, |y - f(x) - \varepsilon|\}. \tag{3}$$

The empirical risk function is as follows:

$$R^{\varepsilon}_{\text{emp}}(f) = \frac{1}{n}\sum_{i=1}^{n} |y - f(x)|_\varepsilon. \tag{4}$$

According to the statistical theory, we bring in two groups of nonnegative slack variable $\{\xi_i\}_{i=1}^n$ and $\{\xi_i^*\}_{i=1}^n$. Then, the question can be converted to the following nonlinear ε-support vector regression machine (ε -SVR) problem:

$$\min_{(\xi_i, \xi_i^*)} \left\{ \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{n} (\xi_i + \xi_i^*) \right\},$$
$$y_i - \omega \cdot \phi(x) - b \le \varepsilon + \xi_i^*, \tag{5}$$
$$\omega \cdot \phi(x) + b - y_i \le \varepsilon + \xi_i,$$
$$\xi_i, \xi_i^* \ge 0,$$

where ε denotes the insensitive loss function. C is used to balance the complex item and the training error of the model.

We bring into Lagrange multipliers α_i and α_i^*, then the convex quadratic programming problem above can be changed into the below dual problem:

$$\max_{(\alpha_i, \alpha_i^*)} \left[-\frac{1}{2}\sum_{i,j=1}^{n} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) K(X_i, X_j) + \sum_{i=1}^{n} \alpha_i^* (y_i - \varepsilon) - \sum_{i=1}^{n} \alpha_i (y_i - \varepsilon) \right], \tag{6}$$
$$\sum_{i=1}^{n} (\alpha_i - \alpha_i^*) = 0,$$
$$0 \le \alpha_i, \alpha_i^* \le C, \quad i = 1, 2, \ldots, n,$$

where $K(X_i, X_j)$ denotes the inner product kernel satisfying Mercer theorem.

We can get the ε-*SVR* function through solving the above dual problem:

$$f(x) = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) K(X_i, X) + b. \quad (7)$$

When ε-*SVR* is used on prediction, it may have a certain error since the data fluctuates violently such as the crude oil price. To reduce the error in some certain as possible as we can, we bring in the dynamic correction factor ε^*. The main idea of the dynamic correction factor is that we use the error of back step with multiplying ε^* to revise the current predicting results. Thus, we can reduce the current predicting error. The dynamic correcting SVR can be defined as follows:

$$Y_d(i+1) = Y_1(i+1) + \varepsilon^* [Y(i) - Y_d(i)], \quad (8)$$

where Y denotes the real results, Y_d denotes the final prediction results, Y_1 denotes the initial predicting results, ε^* denotes the dynamic correction factor, and i denotes the prediction steps.

In order to make the predicting results more accurate, the optimal value of ε^* and the parameters of ε-*SVR* involving C, δ (the variable in gauss kernel function) should be designed (in (8)). To this end, an HRGA is studied below to optimize the following problem:

$$\min_{(\varepsilon^*, C, \sigma)} \sum_{i}^{n} [Y_d(i) - Y(i)]^2. \quad (9)$$

3. HRGA Based on Bare Bones PSO

Assuming that population size is N, the dimension of particle is m. The position of particle i on generation t is $X_i(t) = (x_{i1}(t), \ldots, x_{ij}(t), \ldots, x_{im}(t)), i = 1, 2, \ldots, N$. The speed of particle i on generation t is $V_i(t) = (v_{i1}(t), \ldots, v_{ij}(t), \ldots, v_{im}(t))$. The historic optimal value of individuals is $PBest_i(t) = (pbest_{i1}(t), \ldots, pbest_{ij}(t), \ldots, pbest_{im}(t))$.

Let the global optimal value be $GBest(t) = (gbest_1(t), \ldots, gbest_j(t), \ldots, gbest_m(t))$.

As to standard particle swarm, the position and speed are updated as

$$\begin{aligned} v_{ij}(t+1) &= w v_{ij}(t) + c_1 \cdot r_{1j} \cdot \left(pbest_{ij}(t) - x_{ij}(t)\right) \\ &\quad + c_2 \cdot r_{2j} \cdot \left(gbest_j - x_{ij}(t)\right), \\ x_{ij}(t+1) &= x_{ij}(t) + v_{ij}(t+1), \end{aligned} \quad (10)$$

where ω denotes the inertia weight [21], c_1 and c_2 denote the accelerating operators, and r_{1j} and r_{2j} are uniform distributed random numbers in [0, 1].

In the bare bones particle swarm optimization (PSO), (10) is replaced by (11) as the evolution equation of particle swarm algorithm:

$$\begin{aligned} x_{ij}(t+1) = N\Big(&0.5\left(pbest_{ij}(t) + gbest_j(t)\right) \\ &\times \left|pbest_{ij}(t) - gbest_j(t)\right|\Big). \end{aligned} \quad (11)$$

The position of particle is some random numbers which are gotten from the Gauss distribution. The distribution has the mean value of $(pbest_{ij}(t) + gbest_j(t))/2$ and the standard deviation of $|pbest_{ij}(t) - gbest_j(t)|$.

RNA genetic algorithm is on the basis of base coding and biological molecules operation. Since in the biological molecule, every three bases compose one amino acid. In other words, the bases' length of individuals must be divided exactly by 3. When RNA recoding and protein folding [22], to reduce calculation and to control population size, we assume that the protein folding operation only occurs on the individuals without RNA recoding. Then the most important work is to change the mutation probability [23, 24].

Angeline told us that the essence of particle swarm's position updating was one mutation operation in 1998 [25]. Traditional RNA genetic algorithm mutates as the fixed mutation probability, and the mutation is random with one direction. However HRGA can reflect the historic information of individuals and the sharing information of the population. HRGA can make every individual do directional mutation and improve search efficiency. Moreover, HRGA ensures the strong global search capability, since it does not change the selection and crossover operator.

The procedure of HRGA based on bare bones particle swarm algorithm to optimize the ε-*SVR* parameters and the dynamic correction factor is as follows.

Step 1. Get one group of ε-*SVR* parameters, and the dynamic correction factor randomly, code every parameter, and get the initial RNA population with N individuals, crossover probability P_c, and mutation probability P_m. Assign values for every $PBest_i$ (individual's historic optimal solution) and *GBest* (population's global optimal solution).

Step 2. Compute its error function and get the fitness function. Comparing it with corresponding fitness value of $PBest_i$ and *GBest*, then update $PBest_i$ and *GBest*.

Step 3. Execute the selection operation. Get current generation through coping N individuals from the initial or the last generation.

Step 4. Decide whether the value meets the RNA recoding condition or not. If Y, recode RNA, then go to Step 6. If N, go to Step 5.

Step 5. Decide meet the protein mutual folding condition or not. If Y, execute the protein mutual folding operation. If N, execute the protein own folding operation.

Step 6. Execute the mutation operation as (11) for all the crossover individuals, on the basis of the $PBest_i$ and *GBest*, which have been gotten from Step 2.

Step 7. Repeat Step 2 to Step 6 until the training target meets the condition. At last, we get the optimal parameters of ε-*SVR* and the dynamic correction factor.

Table 1: Benchmark functions.

Function	Formula	Global minimum
Sphere	$f_1(x) = \sum_{i=1}^{n} x_i^2,\ x_i \in [-100, 100].$	$f_1^*(x) = 0,\ x^* = (0, 0, \ldots, 0)$
Rosenbrock	$f_2 = \sum_{i=1}^{n=1} \left(100 \times \left(x_{i+1} - x_i^2\right)^2\right) + \left(x_i - 1\right)^2,\ x_i \in [-30, 30].$	$f_2^*(x) = 0,\ x^* = (1, 1, \ldots, 1)$
Griewank	$f_3 = \frac{1}{4000}\sum_{i=1}^{n} x_i{}^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1,\ x_i \in [-600, 600].$	$f_3^*(x) = 0,\ x^* = (0, 0, \ldots, 0)$

Table 2: Testing results of HRGA and standard RGA.

Function	Hunting zone	Dimension	Iterative times	RGA		HRGA	
				Parameter selection	Mean value of optimum	Parameter selection	Mean value of optimum
Sphere	(−100, 100)	2	250	$P_c = 0.9$, $P_m = 0.1$	8.3421e − 6	$P_c = 0.85$, $P_m = 0.3$	1.9824e − 125
	(−100, 100)	10	1000	$P_c = 0.9$, $P_m = 0.1$	0.0437	$P_c = 0.85$, $P_m = 0.3$	1.4597e − 67
	(−100, 100)	20	1500	$P_c = 0.9$, $P_m = 0.1$	0.1012	$P_c = 0.85$, $P_m = 0.3$	1.5775e − 31
Rosenbrock	(−30, 30)	2	250	$P_c = 0.85$, $P_m = 0.1$	0.5311	$P_c = 0.6$, $P_m = 0.3$	0.0195
	(−30, 30)	10	1000	$P_c = 0.85$, $P_m = 0.1$	15.0766	$P_c = 0.6$, $P_m = 0.3$	3.0922
	(−30, 30)	20	1500	$P_c = 0.85$, $P_m = 0.1$	124.2468	$P_c = 0.6$, $P_m = 0.3$	8.0576
Griewank	(−600, 600)	2	250	$P_c = 0.8$, $P_m = 0.15$	0.0062	$P_c = 0.5$, $P_m = 0.4$	0.0044
	(−600, 600)	10	1000	$P_c = 0.8$, $P_m = 0.15$	0.1468	$P_c = 0.5$, $P_m = 0.4$	0.0132
	(−600, 600)	20	1500	$P_c = 0.8$, $P_m = 0.15$	0.0191	$P_c = 0.5$, $P_m = 0.4$	2.0109e − 3

The flowchart of HRGA to optimize the *ε-SVR* parameters and the dynamic correction factor is shown in Figure 1.

3.1. HRGA Testing. Three classical benchmark functions shown in Table 1 are used to test the property of HRGA.

In addition, among the three functions, Sphere is unimodal function, and the other two are multimodal function.

With the population size $N = 50$, and other parameters determined by multiple test for each function. Each function is tested by HRGA and standard RGA in different dimensions. Each experience is carried on 100 times. Record the mean value of target function's optimum (shown in (12)). The result is displayed in Table 2:

$$\mathrm{MVO} = \frac{1}{N}\sum_{i=1}^{N} f_i'(x). \tag{12}$$

In this equation, MVO denotes the mean value of target function's optimum; $f_i'(x)$ denotes the optimum of benchmark functions in every experiment.

As to the experimental results, with different dimensions having the same iterative times, the mean value of optimum of HRGA is smaller than that of RGA for the three benchmark functions. The average performance of HRGA is closer to the optimum. We can increase the mutation probability appropriately and enhance the convergence speed, since the mutation operator of HRGA has directional local search.

4. Crude Oil Price Prediction Based on a Dynamic Correcting ε-SVR

In this paper, we get the crude oil price from the US Energy Information Administration Web [26]. Since the oil price fluctuates violently, in order to facilitate the processing and decrease the error, we adopt the Cushing, OK WTI Spot Price FOB (dollars per barrel)monthly from January 1986 to now. We take the one hundred data from January 1986 to April 1994 as the test sample. And give the next 20-month dynamic predicting data from May 1994 to December 1995. The relative error of forecasting is shown in Table 2. The prediction effect figure of HRGA and *ε-SVR* with dynamic correction factor is shown in Figure 2. We use Gauss function as the kernel function of *ε-SVR*, which is given as follows:

$$k(x, y) = \exp\left(-\frac{\|x - y\|^2}{2 \cdot \sigma^2}\right). \tag{13}$$

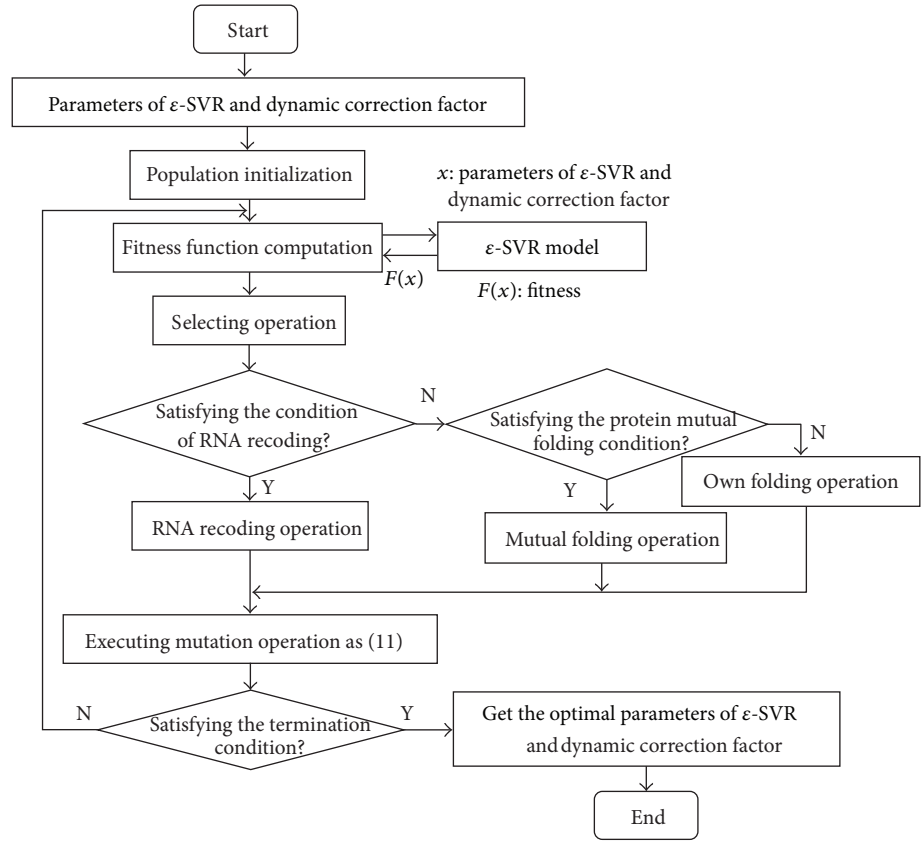


Figure 1: The flowchart of HRGA.

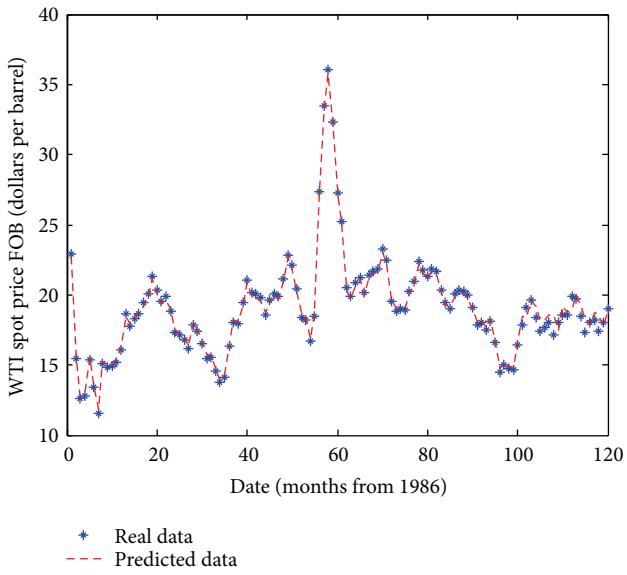


Figure 2: The prediction curve of HRGA-ε-*SVR*.

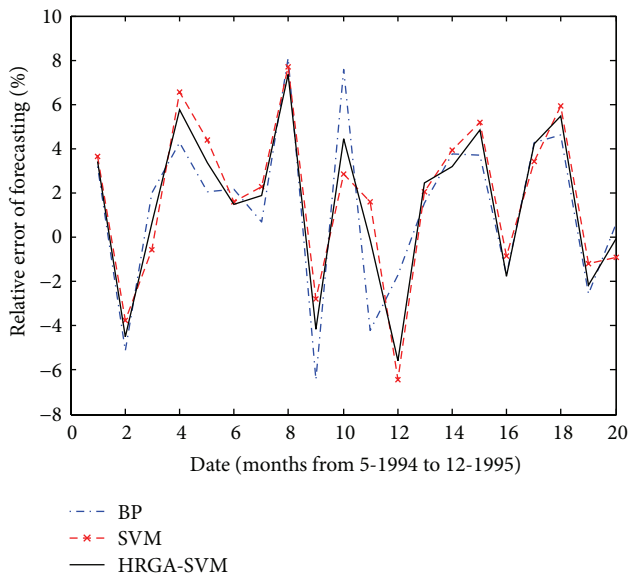


Figure 3: Errors analysis with different models.

Parameter setting of HRGA-ε-*SVR* is with population size being 100, maximum evolution generation being 150, coding length of C being 9, coding length of ε being 8, coding length of σ being 13, coding length of ε^* being 8, P_c being 0.8, and P_m being 0.1.

The optimization interval is set to be

$$1 \le C \le 10000, \quad 0.0001 \le \varepsilon \le 0.1,$$
$$0.01 \le \sigma \le 500, \quad 0.5 \le \varepsilon^* \le 2. \tag{14}$$

Table 3: Analysis results of forecasting error.

Date	BP/%	ε-SVR%	HRGA-ε-SVR /%
5-1994	3.22	3.64	3.42
6-1994	−5.1	−3.78	−4.48
7-1994	1.97	−0.57	0.61
8-1994	4.27	6.55	5.77
9-1994	2.05	4.41	3.38
10-1994	2.19	1.59	1.51
11-1994	0.69	2.3	1.87
12-1994	8.08	7.68	7.35
1-1995	−6.46	−2.79	−4.16
2-1995	7.58	2.87	4.44
3-1995	−4.23	1.61	−0.13
4-1995	−1.69	−6.46	−5.57
5-1995	1.56	2.08	2.48
6-1995	3.75	3.94	3.2
7-1995	3.7	5.19	4.83
8-1995	−1.69	−0.85	−1.75
9-1995	4.27	3.4	4.19
10-1995	4.64	5.94	5.45
11-1995	−2.58	−1.17	−2.14
12-1995	0.62	−0.92	−0.03
δ	4.09	3.96	3.87

When analyzing the results, we define the evaluation index:

$$E_r = \frac{x_i - y_i}{x_i} \times 100\%, \qquad \sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{x_i - y_i}{x_i}\right)^2}. \quad (15)$$

The forecasting error analysis results are shown in Figure 3. In this figure, SVM refers to ε-*SVR*. The BP neural network and ε-*SVR* are with dynamic correction factor which differs them to the traditional method. From Figure 2, we can know that the prediction result is very close to the real value. The HRGA-ε-SVR can be used to predict the crude oil price. Table 3 tells us the WTI crude oil price predicting relative errors of twenty months. Among the three methods in twenty months, the biggest absolute value of relative error of HRGA-ε-*SVR* is the smallest, which is 7.35%, and the smallest root-mean-square of relative error is 3.87%. As to Figure 3, the fluctuation range of HRGA-ε-*SVR* is smaller than those of the other two methods obviously. This means that HRGA-ε-SVR is the best one among the three methods.

5. Conclusions

In this paper, we have presented a novel method on predicting crude oil price. This method bases on an ε-support vector regression machine with dynamic correction factor correcting predicting errors. We also proposed the HRGA, with the position displacement idea of bare bones PSO changing the mutation operator, to optimize the parameters in an ε-SVR. The predicting result of crude oil price shows the validity of the proposed method. Thus, the ε-SVR model can also be applied to predict tendency in other practical areas.

Acknowledgments

The research was partially supported by Grant no. 60974039 from the National Science Foundation of China and by Grant no. ZR2011FM002 from the Natural Science Foundation of Shandong Province.

References

[1] B. Hunt, P. Isard, and D. Laxton, "The macroeconomic effects of higher oil prices," IMF Working Paper No.wp/01/14, 2001.

[2] Y. Fan, K. Wang, Y. J. Zhang et al., "International crude oil market analysis and price forecast in 2009," *Bulletin of Chinese Academy of Sciences*, vol. 4, no. 1, pp. 42–45, 2009.

[3] C. Morana, "A semiparametric approach to short-term oil price forecasting," *Energy Economics*, vol. 23, no. 3, pp. 325–338, 2001.

[4] S. Mirmirani and H. Cheng Li, "A comparison of VAR and neural networks with genetic algorithm in forecasting price of oil," *Advances in Econometrics*, vol. 19, pp. 203–223, 2004.

[5] Z. J. Ding, Q. Min, and Y. Lin, "Application of ARIMA model in forecasting prude oil price," *Logistics Technology*, vol. 27, no. 10, pp. 156–159, 2008.

[6] J. P. Liu, S. Lin, T. Guo, and H. Y. Chen, "Nonlinear time series forecasting model and its application for oil price forecasting," *Journal of Management Science*, vol. 24, no. 6, pp. 104–112, 2011.

[7] S. Y. Wang, L. Yu, and K. K. Lai, "Crude oil price forecasting with TEI@ I methodology," *Journal of Systems Sciences and Complexity*, vol. 18, no. 2, pp. 145–166, 2005.

[8] W. Xie, L. Yu, S. Xu, and S. Wang, "A new method for crude oil price forecasting based on support vector machines," *Lecture Notes in Computer Science*, vol. 3994, pp. 444–451, 2006.

[9] R. A. N. Mohammad and A. G. Ehsan, "A hybrid artificial intelligence approach to monthly forecasting of crude oil price time series," in *The Proceedings of the 10th International Conference on Engineering Applications of Neural Networks*, pp. 160–167, 2007.

[10] S. Guo and P. Lai, "The time series mixed model and its application in price prediction of international crude oil," *Journal of Nanjing University of Information Science & Technology*, vol. 2, no. 3, pp. 280–283, 2010.

[11] Y. B. Hou, J. Y. Du, and M. Wang, *Neural Networks*, Xidian University Press, Xi'an, China, 2007.

[12] H. Zhu, L. Qu, and H. Zhang, "Face detection based on wavelet transform and support vector machine," *Journal of Xi'an Jiaotong University*, vol. 36, no. 9, pp. 947–950, 2002.

[13] R. Feng, C. L. Song, Y. Z. Zhang, and H. H. Shao, "Comparative study of soft sensor models based on support vector machines and RBF neural networks," *Journal of Shanghai Jiaotong University*, vol. 37, pp. 122–125, 2003.

[14] J. Tao and N. Wang, "DNA computing based RNA genetic algorithm with applications in parameter estimation of chemical engineering processes," *Computers & Chemical Engineering*, vol. 31, no. 12, pp. 1602–1618, 2007.

[15] K. Wang and N. Wang, "A protein inspired RNA genetic algorithm for parameter estimation in hydrocracking of heavy oil," *Chemical Engineering Journal*, vol. 167, no. 1, pp. 228–239, 2011.

[16] K. Wang and N. Wang, "A novel RNA genetic algorithm for parameter estimation of dynamic systems," *Chemical Engineering Research & Design*, vol. 88, no. 11, pp. 1485–1493, 2010.

[17] D. Bratton and J. Kennedy, "Defining a standard for particle swarm optimization," in *Proceedings of the IEEE Swarm Intelligence Symposium (SIS '07)*, pp. 120–127, April 2007.

[18] N. Y. Deng and Y. J. Tian, *A New Method of Data Mining and Germany: Support Vector Machines*, Science Press, Beijing, China, 2004.

[19] U. Thissen, R. Van Brakel, A. P. De Weijer, W. J. Melssen, and L. M. C. Buydens, "Using support vector machines for time series prediction," *Chemometrics and Intelligent Laboratory Systems*, vol. 69, no. 1-2, pp. 35–49, 2003.

[20] K. J. Kim, "Financial time series forecasting using support vector machines," *Neurocomputing*, vol. 55, no. 1-2, pp. 307–319, 2003.

[21] Y. Shi and R. Eberhart, "Modified particle swarm optimizer," in *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 519–523, 1998.

[22] D. P. Clark, *Molecular Biology: Understanding the Genetic Revolution*, Academic Press, New York, NY, USA, 2005.

[23] J. Lis, "Genetic algorithm with the dynamic probability of mutation in the classification problem," *Pattern Recognition Letters*, vol. 16, no. 12, pp. 1311–1320, 1995.

[24] M. Serpell and J. E. Smith, "Self-adaptation of mutation operator and probability for permutation representations in genetic algorithms," *Evolutionary Computation*, vol. 18, no. 3, pp. 491–514, 2010.

[25] P. J. Angeline, "Evolutionary optimization versus PSO: philosophy and performance differences," *Evolutionary Programming*, vol. 7, pp. 601–610, 1998.

[26] http://www.eia.gov/dnav/pet/hist/LeafHandler.ashx?n=PET&s=RWTC&f=M.