Advances in Multimedia

# Recent Machine Learning Progress in Image Analysis and Understanding

Lead Guest Editor: Shengping Zhang
Guest Editors: Huiyu Zhou and Lei Zhang

# Recent Machine Learning Progress in Image Analysis and Understanding

Advances in Multimedia

# Recent Machine Learning Progress in Image Analysis and Understanding

Lead Guest Editor: Shengping Zhang
Guest Editors: Huiyu Zhou and Lei Zhang

# Editorial Board

# Contents

Hindawi
Advances in Multimedia
Volume 2018, Article ID 1685890, 2 pages
https://doi.org/10.1155/2018/1685890

*Editorial*

# Recent Machine Learning Progress in Image Analysis and Understanding

**Shengping Zhang**,[1] **Huiyu Zhou**,[2] **and Lei Zhang**[3]

[1]*Harbin Institute of Technology, Weihai 264209, China*
[2]*Queen's University Belfast, Belfast, UK*
[3]*University of Pittsburgh, Pittsburgh, USA*

Correspondence should be addressed to Shengping Zhang; s.zhang@hit.edu.cn

Received 28 November 2018; Accepted 28 November 2018; Published 10 December 2018



Recently, artificial intelligence and machine learning have attracted increasing attention and achieved great success in both research community and industry especially in the field of multimedia. With the recent progress in machine learning especially in deep learning, many tasks in image analysis and understanding have been applied to solve real problems. For example, since the deep learning based classifier was successfully used in image classification in 2012, deep learning has also been widely used in other computer vision tasks such as video classification and image super-resolution. Learning an effective feature representation from a large number of data is capable of extracting the underlying structure features of the data, which produce better representation than hand-crafted features since the learned features adapt well to the tasks at hand. However, most of the existing deep learning based methods need to learn a huge number of parameters especially with the increasingly complicated network, which restricts their applications in image analysis and understanding in real-time environments.

The primary purpose of this special issue is to organize a collection of recently developed machine learning methods as well as their applications in image analysis and understanding. The special issue is intended to be an international forum for researchers to report the recent developments in this field in an original research paper style. Review articles which describe the current state of the art are also welcomed. From 16 submissions, 7 papers are published in this special issue. Each paper was reviewed by at least two reviewers and revised according to review comments. These papers involve image classification, image segmentation, visual tracking, person reidentification, and so on.

In F. Yang et al.'s paper, the authors implement the MLP neural network on volume data to denoise while preserving the boundary. This method can considerably improve quality of volume data acquired by devices. Then we improve the LH method by combining the regional depth information to achieve the transfer function semiautomatic generation. This method can avoid the influence of noise and make the voxels more centralized. In the LH histogram the voxel distribution at the diagonal line is more concentrated, and the boundary of important objects is effectively emphasized. The features of interest in the data can thus be found exactly by mapping scalar value of boundary voxels which correspond to the points in LH histogram to appropriate opacity and color.

In W. Liu and Y. Zhao's paper, the authors make deep study about the using of word structure of light on the object surface reconstruction. After an image for denoising, they can minimize the impact of other lights to the photographic picture, increasing the compatibility of reading photos; to get each of the height line of the sum and interpolation operations, they then get a smooth three-dimensional reconstruction of the surface of the object. The latter part of the research process will focus on three-dimensional high-precision, high-speed, and real-time reconstruction for further study.

In K. Zhang et al.'s paper, the authors concentrate on identifying tomato leaf disease using deep convolutional neural networks by transfer learning. The utilized networks are based on the pre-trained deep learning models of AlexNet, GoogLeNet, and ResNet. First we compared the relative performance of these networks by using SGD and Adam optimization method, revealing that the ResNet with SGD

optimization method obtains the highest result with the best accuracy 96.51%. Then, the performance evaluation of batch size and number of iterations affecting the transfer learning of the ResNet was conducted. A small batch size 16 combining a moderate number of iterations 4992 is the optimal choice in this work. Our findings suggest that, for a particular task, neither large batch size nor large number of iterations may not improve the accuracy of the target model. The setting of batch size and number of iterations depends on your data set and the utilized network. Next, the best combined model was used to fine-tune the structure. Fine-tuning ResNet layers from 37 to "fc" obtained the highest accuracy 97.28% in identifying tomato leaf disease. Based on the amount of available data, layer-wise fine-tuning may provide a practical way to achieve the best performance of the application at hand. We believe that the results obtained in this work will bring some inspiration to other similar visual recognition problems, and the practical study of this work can be easily extended to other plant leaf disease identification problem.

In M. Sun et al.'s paper, after experimenting with two different data sources using the proposed method, the following conclusions can be drawn: through a large number of experiments, a set of optimal combination parameters suitable for random forest was obtained. That is, we set the number of decision trees at 300 and the number of nodes at 4. When the random forest parameters were optimal, samples of 5% and 10% of the total were selected for the experiment, and the samples of 5% and 10% were added each time. The weighted entropy algorithm was used to select samples with the largest entropy values to train the new training set proposed in this paper. The classifier used the remaining data as the test data to evaluate the performance of the classifier and to test the universality of the classifier. Compared with the traditional classifier based on supervised classification and SVM, we proved via a large number of experiments that the proposed weighted entropy semisupervised ensemble classifier based on random forest showed better classification performance and better universality.

In R. J. Hemalatha et al.'s paper, a method was presented to evaluate the active contour segmentation algorithms to segment the synovial region from arthritis affected finger ultrasound image. Performance analysis metrics like Dice coefficient and Hausdroff distance and statistical analysis metrics like standard error and F-test show the significant difference between the two segmentation method (Caselles and Lankton) for synovial region. Further classification is performed for the derived features such as performance metrics and statistical values. Higher accuracy is described for Lankton as the result of classification process. Hence the output of the research work shows that Lankton method is the best method for synovial region segmentation from ultrasound images.

In W. Liu and H. Wang's paper, the authors propose to use compressed features to model the tracked target's appearance and then use SVM to perform tracking. The experimental results indicate that the proposed method outperforms several state-of-the-art methods. The advantages of the proposed method are twofold: (1) It is good at handling scale changes of the target over time because the used features are obtained by multiscale wavelet transformation. (2) The speed of the proposed method can achieve real-time because the dimensionality of the used features was reduced by compressed sensing techniques.

M. A. Syed et al.'s paper presents a metric learning approach that exploits both multimodal transforms and cross views impostors to improve the capability of metric to discriminate among different persons as well as enhance rejection capability to decline large number of real world diverse impostors. In real world mostly pedestrian images are multimodal, and in public spaces several persons share similar clothing; therefore, our IRM3 is learned to tackle such issues of reidentification and person tracking in public spaces. Extensive experiments on three challenging datasets (VIPeR, CUHK01, and CUHK03) demonstrate the effectiveness of our IRM3 metric which has outperformed many previous state of the art metrics. In addition, we further intend to extend our approach for testing in real world scenario and intend to solve various other issues for real time implementation.

The accepted papers present recent machine learning progress in image analysis and understanding. We hope that this special issue would attract a major attention of the peers.

## Conflicts of Interest

The editors declare that they have no conflicts of interest regarding the publication of this special issue.

## Acknowledgments

We would like to express our appreciation to all the authors, reviewers, and the editor-in-chief for great support to make this special issue possible.

*Shengping Zhang*
*Huiyu Zhou*
*Lei Zhang*

Hindawi
Advances in Multimedia
Volume 2018, Article ID 4189125, 9 pages
https://doi.org/10.1155/2018/4189125

*Research Article*

# Height Estimation of Target Objects Based on Structured Light

**Wei Liu and Yongsheng Zhao**

*Department of Modern Education Technology, Ludong University, Yantai, China*

Correspondence should be addressed to Wei Liu; ldulw@sina.com

Received 20 June 2018; Accepted 25 September 2018; Published 1 November 2018

Guest Editor: Shengping Zhang



The height estimation of the target object is an important research direction in the field of computer vision. The three-dimensional reconstruction of structured light has the characteristics of high precision, noncontact, and simple structure and is widely used in military simulation and cultural heritage protection. In this paper, the height of the target object is estimated by using the word structure light. According to the height dictionary, the height under the offset is estimated by the movement of the structured light to the object. In addition, by effectively preprocessing the captured structured light images, such as expansion, seeking skeleton, and other operations, the flexibility of estimating the height of different objects by structured light is increased, and the height of the target object can be estimated more accurately.

## 1. Introduction

In recent years, with the development of science and technology, three-dimensional reconstruction technology as an important part of machine vision has attracted more and more attention, especially in industrial product design and cultural heritage protection. However, based on the three-dimensional reconstruction of the surface of the structured light, it is possible to reconstruct the surface of the object by laser scanning without touching the object, which can greatly protect the original culture from damage in the cultural heritage. This can make a great contribution to the excavation of ancient excellent culture and the spread of Chinese civilization. Therefore, the three-dimensional reconstruction based on structured light has important practical significance for the protection of cultural heritage and the design of industrial products [1–3]. At present, in the three-dimensional reconstruction of the main use of word-structured optical scanning method and three-dimensional reconstruction of structured light technology as a noncontact active measurement technology, with low cost, high precision, vision, real-time, anti-interference ability, and so on, these characteristics will inevitably make the next few years of this reconstruction will have a better development prospects [4, 5].

3D surface reconstruction is to rebuild the actual shape of the real life of the object, which has become an important topic in computer vision. And researchers from all over the world have made considerable achievements in this regard. The structure of three-dimensional reconstruction system for structured light mainly includes cameras and lasers; you can use the ordinary camera to complete the task of detection, but because of the different structure of the light, the experimental results will be affected. According to the laser projection of different ways can be divided into point, line, and multiline structure of light. Point structure light for the laser projector projection of a beam of light, measured the surface of the measured object a point; the camera can only get this photo of the three-dimensional coordinates of the information; the amount of information is too small; a word line structure light projector projects a light plane; the intersection of the light plane and the measurement object can draw a cross section information; the algorithm is easy to use; multirow structured light projects multiple light planes; the surface of the object forms multiple laser lines; pictures can give us multiple cross-section information, which is large amount of information; however, it is necessary to increase the matching of light bars, which greatly improves the difficulty and complexity of the algorithm and is still in the stage of experimental research [6–8].

At home and abroad for the 3D surface reconstruction conducted in-depth study, Horn [9] proposed the concept of SFS, which is a widely concerned three-dimensional shape

reconstruction of the important ideas. The main content of this idea is to reconstruct the three-dimensional shape of the surface of the object by identifying and analyzing the shape information of the direction of the light, the brightness, the surface shape of the object, and the grayscale variation of the reflection model. Ikeuchi and Horn [10] are used to solving the three-dimensional reconstruction by using the illuminance equation and the smoothing criterion as the constraint of reconstruction. So, the problem of 3D reconstruction is transformed into the minimization problem of solving function. In this case, Horn proposed another smoothing standard. The main content is a smooth surface, where the surface obeys the integrable constraint, because the algorithm is seeking to directly recover complex surface unit normal vectors, so that reconstruction cannot get the absolute height of the surface. Fuqiang Zhou [11] used to use the same way to achieve the cross-laser plane calibration. In the experiment, four edge feature points of the space disk are obtained and the radius of the disk is calculated by fitting the feature points. The absolute error of the radius is 0.0_59mm. Harbin Institute of Technology Dongbin Zhao [12] and other scholars put forward a new monocular image restoration object surface height and gradient algorithm is an iterative calculation of the composite image, obtaining accurate surface height, and they also validate the feasibility of the algorithm for actual solder joint images. Ruiling Liu [13], for high light and shadow, put forward a four-light source vector selection algorithm; she compares the normal vector of different pixels recovery with the mirror reflection direction and chooses the nearest normal vector to restore the shape of the required vector, which avoids the error caused by the threshold elimination of high light and shadow in the traditional algorithm, and remove the high light and shadow constraints on the algorithm to expand the scope of application of the algorithm.

## 2. Based on the Gradient of Moving Objects Detection

In the process of 3D reconstruction of structured light, in order to reconstruct the 3D structure of the 2D image taken by the camera, the camera parameter must be calibrated and the geometric model of camera imaging should be built; that is, the camera's internal and external parameters should be measured. Then, the correspondence between the image and the spatial point is constructed; that is, the laser plane equation is calibrated. This paper mainly used Zhenyou Zhang camera calibration method [14].

*2.1. Camera Parameter Calibration.* The camera model is very similar to the model used by Heikkila and Silven of the University of Oulu in Finland. We especially recommend their CVPR'97 paper: the function of the four-step camera calibration program with an implicit image correction [15].

In the camera model, the parameters are as follows:

Focal length: stored in pixels in 2∗1vector $f_c$.

Main points: the coordinates of the primary point are stored in the 2∗1 vector $cc$.

Skew factor: define the skew factor for the angle between the $x$ and $y$ axes in the scalar $\alpha_c$.

Distortion: the image distortion factor (radial and tangential distortion) is stored in the 5∗1 vector $k_c$.

Let $p$ be the spatial point of the coordinate vector $XX_c = [X_c; Y_c; Z_c]$ in the reference frame of the camera. And then the projection is performed on the image plane based on the intrinsic parameter $(f_c, cc, \alpha_c, k_c)$.

Let $x_n$ be normalized (pinhole) image projection:

$$x_n = \begin{bmatrix} \frac{Xc}{Zc} \\ \frac{Yc}{Zc} \end{bmatrix} \tag{1}$$

Let $r^2 = x^2 + y^2$; after the lens is distorted, the new normalized point coordinates $x_d$ are defined:

$$\begin{aligned} x_d &= \begin{bmatrix} x_d(1) \\ x_d(2) \end{bmatrix} \\ &= \left(1 + k_c(1)\, x^2 + k_c(2)\, r^4 + k_c(5)\, r^6\right) x_n + d_x \end{aligned} \tag{2}$$

where $d_x$ is the tangential distortion vector:

$$d_x = \begin{bmatrix} 2k_c(3)\, xy + k_c(4)\left(r^2 + 2x^2\right) \\ k_c(3)\left(r^2 + 2x^2\right) + 2k_c(4)\, xy \end{bmatrix} \tag{3}$$

Therefore, $k_c(5)$ contains the radial, tangential distortion coefficient [16]. It is worth noting that this distortion model was first introduced by Brown in 1966, called the "Plumb Bob" model (radial polynomial + "thin prism"). The tangential distortion is due to the incorrect alignment of the "eccentric" in the composite lens or other manufacturing defects of the lens assembly.

Once the distortion is applied, the final pixel coordinates of the P on the projection plane are $x_{pixel} = [x_p; y_p]$:

$$\begin{aligned} x_p &= f_c(1)\left(x_d(1) + \alpha_c * x_d(2)\right) + cc(1) \\ y_p &= f_c(2)\, x_d(2) + cc(2) \end{aligned} \tag{4}$$

Thus, the pixel coordinate vector $x_{pixel}$ and the normalized (distorted) coordinate vector $x_d$ are related to each other by a linear equation [17]:

$$\begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = KK \begin{bmatrix} x_d(1) \\ x_d(2) \\ 1 \end{bmatrix} \tag{5}$$

where $KK$ is called the camera matrix and is defined as follows:

$$KK = \begin{bmatrix} f_c(1) & alpha\_fc(1) & cc(1) \\ 0 & f_c(2) & cc(2) \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

*2.2. External Condition Variable Setting.* In reconstructing the surface height of the object, we are using triangles similar to the surface reconstruction of the object. In the case of similar judgments, the degree of use of the triangles is the same. But when the triangles are similar, we use the same degree of the angle of the two similar triangles [18]. Thus, in the world coordinate system with the intersection of the center of view and the center of the object, there exists a proportional relationship between the tangent values of the corners in the two right angles, so that the degree of the angle $\alpha$ and the distance $L$ between the optical center and the object need to be known. These two variables can change the position of the camera structure by artificial changing. Therefore, when setting up the camera and the structure of the light we need to measure the angle $\alpha$ and length $L$. And these two are invariants; that is to say in the whole process of shooting we must ensure that the two variables remain unchanged or reconstructed objects will be distorted. So, in the process of taking pictures we must ensure that the external variables remain unchanged, so as to better reconstruct the surface of the object.

## 3. Basic Principle of Three-Dimensional Reconstruction of Structured Light

In order to obtain the three-dimensional information of the object in the structured light measurement, the basic idea is to use the geometric information in the structured light image to help provide the geometric information in the scene [19]. According to the geometric relationship inside the camera, we can determine the structure of light and the geometric relationship between objects, thus rebuilding the surface of the object.

*3.1. The Correspondence between Pixels and a World Coordinate Point.* As shown in Figure 1, the angle between the structural smooth and the optical axis of the camera is $\alpha$, and the origin $O_w$ of the world coordinate system $O_w - X_wY_wZ_w$ is located at the intersection of the camera's optical axis and the structured light plane. The $X_w$-axis and the $Y_w$-axis are parallel to the camera coordinate systems $X_c$ and $Y_c$, respectively, and $Z_w$ and $Z_c$ coincide but are opposite [20]. The distance between $O_w$ and $O_c$ is $l$. Thus, the world coordinate system and the camera coordinate system have the following relationship:

$$\begin{aligned} X_c &= X_w \\ Y_c &= -Y_w \\ Z_c &= l - Z_w \end{aligned} \tag{7}$$

A$'$ is the image of A in the world coordinate system; the line of sight OA$'$ is

$$\frac{X_w}{X} = -\left(\frac{Y_w}{y}\right) = \frac{(l - Z_w)}{f} \tag{8}$$

In the world coordinate system, the plane equation of structured light is

$$X_w = Z_w \tan\alpha \tag{9}$$

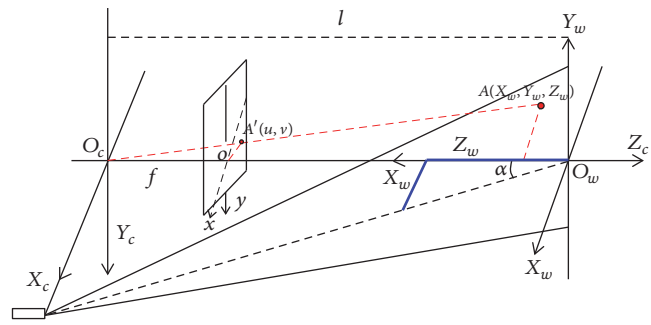



Figure 1: Dimensional reconstruction of structured light.

where $\alpha$ is the angle between the camera and the laser pen. The solutions of (9) are

$$\begin{aligned} X_w &= \frac{(x_l \tan\alpha)}{(x + f\tan\alpha)} \\ Y_w &= \frac{(-y_l \tan\alpha)}{(x + f\tan\alpha)} \\ Z_w &= \frac{x_l}{(x + f\tan\alpha)} \end{aligned} \tag{10}$$

Because $O_p - uv$ is the Cartesian coordinate system defined on the digital image [21], $(u, v)$ is the coordinates of the pixels, and u and v represent the number of rows and rows of pixels in the image array, respectively. Establish the coordinate system $O_i - xy$ expressing in physical units parallel to the $u$-axis and the $v$-axis, the origin is the camera optical axis and image. The plane is usually located in the center of the image, but in reality there will be a small offset; $O_p - xy$'s coordinates are recorded as $(u_0, v_0)$. The physical dimensions of each pixel in the $x$-axis and $y$-axis directions are $S_x$ and $S_y$; the coordinates of any one of the two coordinate systems are represented by a uniform coordinate and a matrix, with the following relationship:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{S_x} & 0 & u_0 \\ 0 & \frac{1}{S_y} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{11}$$

The inverse relationship is

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & 0 & -u_0S_x \\ 0 & S_y & -v_0S_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{12}$$

Thus, it can be learned that the correspondence between pixel points and world coordinate points is

$$X_w = \frac{(f_x(u - u_0)l\tan\alpha)}{(f_x(u - u_0) + \tan\alpha)}$$

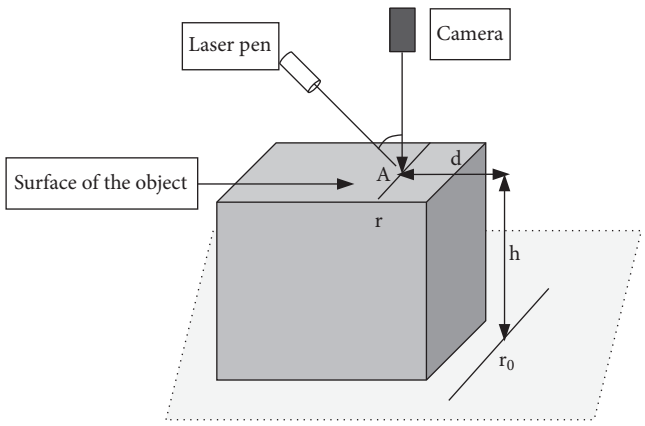



Figure 2: Schematic diagram of experimental laser photography.

$$Y_w = \frac{\left(-f_y\left(v - v_0\right) l \tan\alpha\right)}{\left(f_x\left(u - u_0\right) + \tan\alpha\right)}$$
$$Z_w = \frac{\left(f_x\left(u - u_0\right) l\right)}{\left(f_x\left(u - u_0\right) + \tan\alpha\right))} \tag{13}$$

*3.2. Surface Height Calculation Principle.* As shown in Figure 2, the corresponding relationship between the pixel and the point in the world coordinate system is shown in (13). In the experiment, we simplified the shooting method. The laser angle and the vertical direction remained unchanged at 30°, so it was easy to calculate [22, 23]. When the shooting platform has no objects, the laser light directly to the platform will not be offset, but when the object is placed on the platform, the laser light to the surface of the object will occur after a certain shift. As shown in Figure 1, $r_0$ is the reference laser line, and $r$ is the laser line that is offset after adding the object. Since the angle is 30°, $\tan\alpha = h/d = \sqrt{3}/3$, so the relationship between the horizontal offset $d$ of the laser and the height $h_0$ of the point $A$ of the object is known [24]. It can also be seen from Figure 1 that if the distance $L$ between the light and the object and $\alpha$ changes, the reconstruction will change.

## 4. Denoising after Loading the Mask

Due to shooting methods and other reasons, there is a certain amount of noise in the loaded laser mask. Here to solve the two main noises, other light source interference and laser line breakage, the main method is to filter the connected domain to remove other light source interference, through the expansion of the skeleton to avoid laser line breakage.

*4.1. Filter the Connected Domain to Remove Other Light Sources.* Filtering the connected domain is to keep the connected domain in the image and remove those nonconnected pixels. Here is the use of bwareaopen function; this function is also called delete the minimum area function; you can set the minimum size of the connected domain, which has the default value of 8. In the experiment, this value is set to 2

Figure 3: With interference laser mask.

Figure 4: Remove the interference after the laser mask.

in this paper. In the design of the function, after loading the laser mask, the image will be converted to a black and white image; the image matrix is shown as 0-1 matrix. However, due to other light sources, there are some interference points in the image (as shown in Figure 3).

These interference points do not exist in the form of communication, but in the form of pixels scattered in the image, so you can filter the connection domain and remove these interference points, and this operation will have a sharp effect on the laser itself. That is, around the laser line "burr" will be deleted, which will make the reconstruction results more smooth. The effect after screening is shown in Figure 4.

*4.2. Expand the Skeleton to Obtain the Laser Line.* Laser mask image screening connection domain processing will be loaded; the laser line itself will be interference. The biggest problem is that the laser line is broken. In view of this situation, first of all, we have carried out the expansion operation and first broken the laser line through the expansion of the connection; the effect is shown in Figure 5.

After the laser line is inflated; the laser light becomes thicker. Obviously, we cannot use this inflated image directly to a high degree of reconstruction. All we have to do is to get a

Figure 5: The effect after expansion.

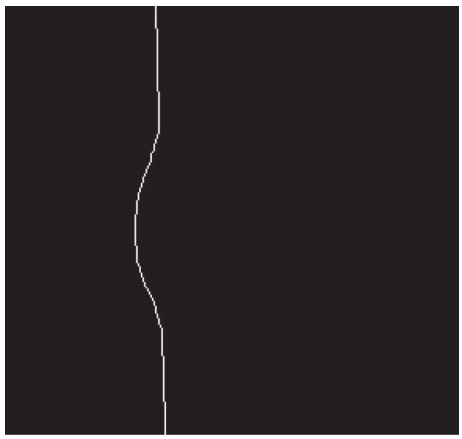


Figure 6: The effect after seeking skeleton operations.

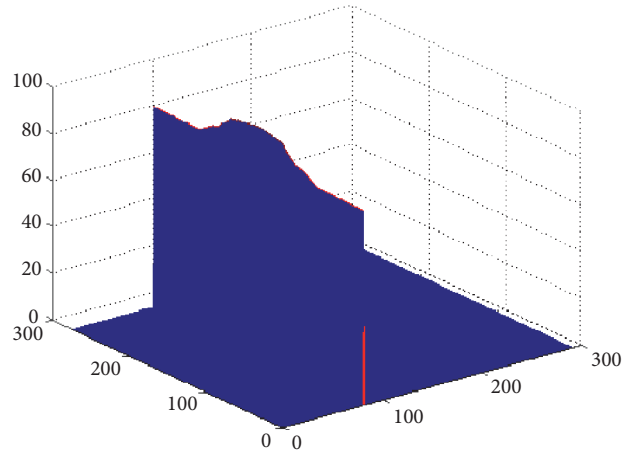



Figure 7: Single laser height reconstruction.

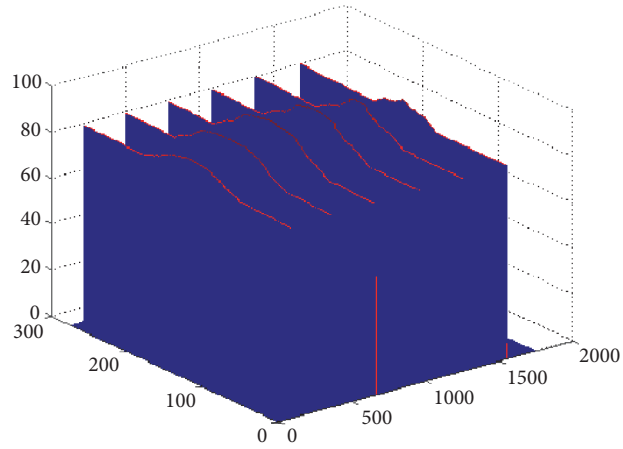



Figure 8: Multiple poststack height reconstruction.

thin continuous laser line, and we cannot change the shape of the original laser line. So we took the skeleton operation. This operation will be the same as the original laser line shape of the laser line, and this laser line is a single row of pixels of the laser line, which is in line with our requirements. The effect is shown in Figure 6.

## 5. Perform a High Degree of Summation and Interpolation

The main content of this part is taking the main process after rebuilding a single laser height: superposition and interpolation. The superposition is mainly a comprehensive display of each reconstructed laser height. Interpolation is the linear interpolation of the resulting discrete data, making it appear continuously and smoothly.

*5.1. Height Superimposed and Evenly Displayed.* This paper is designed to reconstruct the height of the surface of the object by using a single word structure, but a laser can only reproduce the height of the laser line (Figure 7).

Therefore, if the we use word structure of light on the surface of the three-dimensional reconstruction, there are two ways. One is to take the image into video and then a frame of a video of the laser line in a high degree of reconstruction, which will get a relatively smooth surface of the object, but this method is more difficult to shoot, the data being many. The second is that the isometric image is highly reconstructed and then interpolated. This method is relatively simple. No matter what method is used, the final reconstruction is a section of the height matrix. Therefore, to sum up the height of each reconstruction, each height matrix in a world coordinate system is displayed (shown in Figure 8).

Because we need to ensure the laser line and the location of the mandrel and their angle in the shooting of the image, we can only be moving objects when we shoot an object. Only in this way can we ensure the same angle between the laser line plane and the camera object, in order to accurately rebuild the height of the object, that is, to ensure that the angle between XcOw and OcOw. Therefore, when shooting a number of laser lights to rebuild the height we can only move the object to shoot, but the image will be in the same position if we shoot laser line. Then, the reconstruction of the laser height will be superimposed. Therefore, it is necessary for man-made reconstruction of the laser height according to the distance when the object is moving evenly distributed

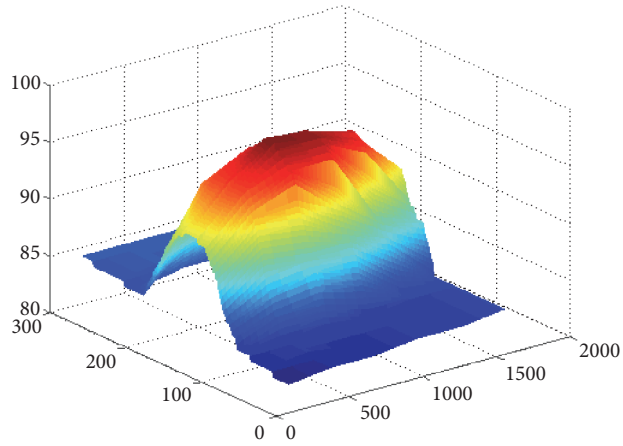



Figure 9: The results of the reconstruction after interpolation.

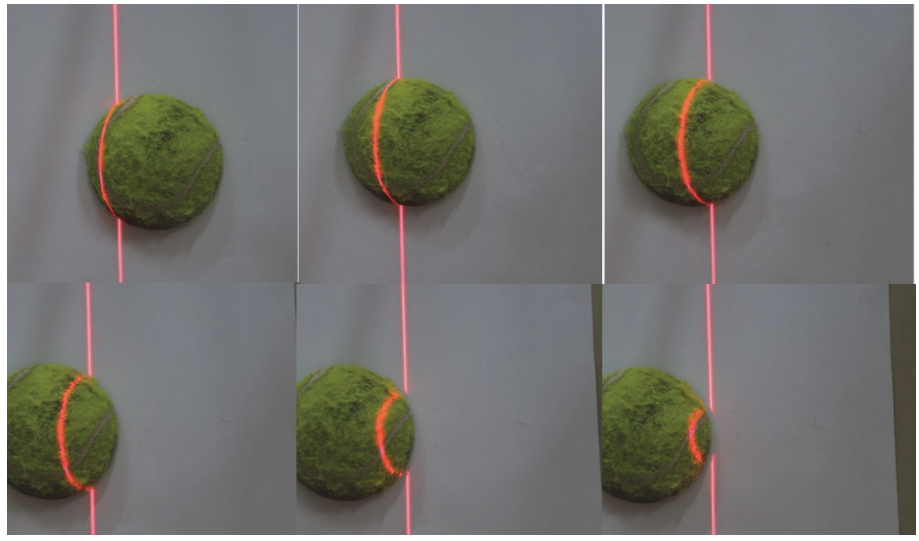


Figure 10: Hemispherical picture taken in kind.

such that each height line is shown to be scattered (shown in Figure 8).

*5.2. Interpolate to Reconstruct a Smooth Surface.* As the design uses a word structure of light on the surface of the three-dimensional reconstruction of the object, the word structure of light can only rebuild a laser line under a height. After the above superposition, we will get a lot of high degrees of reconstruction, but these are not continuous but a height line. In order to rebuild these lines into the surface, there are two ideas: one is to take a lot of height lines for superposition; the other is to take a limited height line for superposition and then interpolation. These two methods can get a smooth surface reconstruction of the object, but the former method of workload is too large; here we use the second method, the height of the superposition of the interpolation operation, and the superposition of the use of the griddata function and of the discrete height of the linear interpolation to get a smooth surface of the object and the effect is shown in Figure 9.

## 6. Experimental Results and Analysis

This chapter is mainly to reconstruct the experimental results according to the physical comparison and analyze the advantages and disadvantages of the reconstruction of the experimental results.

*6.1. Comparison of Physical and Reconstruction Results.* In order to better test the continuity of reconstruction of a high degree, this paper is selected as a hemisphere, because the hemisphere in the rise or fall is continuous, so this can better reflect the effect of reconstruction. And in order to reduce the reflection of interference that the laser irradiation on the surface of the object caused, we then select the rough diffuse reflector to take pictures. As can be seen from Figure 10, the hemisphere's tennis is exactly in line with our basic requirements, and the rough surface of the tennis is just a diffuse material.

The three-dimensional reconstruction of structured light is based on the degree of deviation of the laser line and then

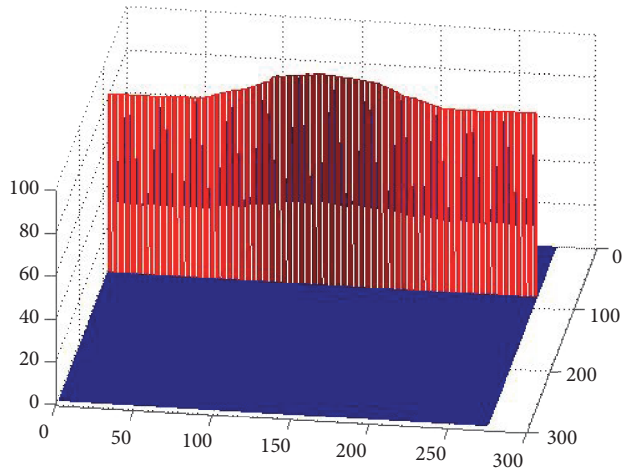



Figure 11: Highest height of reconstruction.

Figure 12: Height of the actual object.

multiplied by the height of the offset to reconstruct the height of the object. But in the process of readding because the reconstruction of the height is too small, basically we do not see the surface of the reconstruction of the object.

Therefore, this article will rebuild the height in accordance with a certain proportion of the amplification. But for the comparison of the actual object and the reconstruction height, it can be seen that the height of the reconstruction is higher than the actual height. The results are shown in Figures 11 and 12.

*6.2. Analysis of Other Groups of Results.* From the comparison to the hemisphere reconstruction results and the actual object, the reconstruction results have been reconstructed out of the hemisphere, but for the reconstruction of the hemisphere there is a certain error. For example, the hemisphere is not very standard, and there is an error in the reconstructed hemisphere surface. First of all, from Figure 8 we can see that before the interpolation of their height of the degree of bending the laser line hit the object on the degree of bending more consistently. After the interpolation, we can see that his image has better reconstructed the surface of the object, and the interpolation is relatively smooth.

The design of this paper, from the beginning has been the use of the hemisphere for debugging and a series of operations; when the program is completed introducing a number of other objects, the compatibility of the program was tested. The first is to introduce a rectangular model (shown in Figure 13)

The laser data taken in the program according to the experimental data taken in Figure 13 is shown in Figure 14; as a result of shooting, the experimental data is the deviation of the laser line in the reconstruction of the experimental results that are skewed. The actual height is shown in Figure 15.

Figures 13, 14, and 15 show the rejoined results of a rectangle introduced into the program.

*6.3. Height Comparison.* According to the actual height of the object measurement results and reconstruction results to do a comparison, as shown in Table 1. From the table we can see, in the reconstruction of the height of the object, the accuracy is very high.

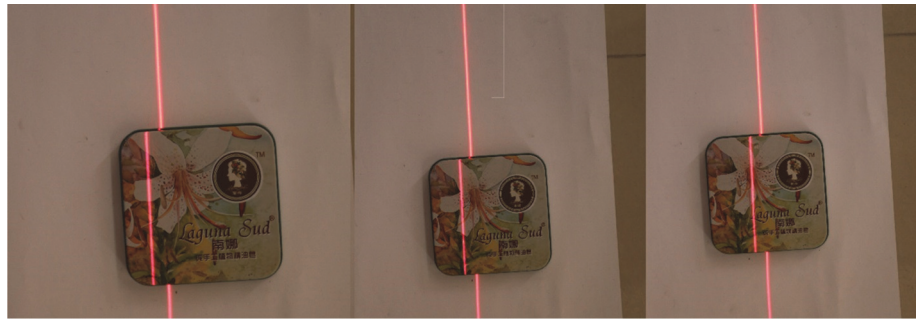


Figure 13: Rectangular physical photograph.

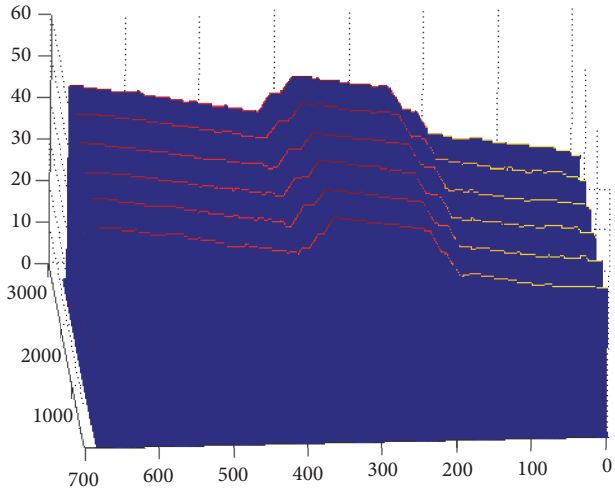



Figure 14: Laser height reconstruction.

Figure 15: Actual cuboid height.

## 7. Summary

In this paper, we deeply study the using of word structure of light on the object surface reconstruction. Given an image for denoising, we can minimize the impact of other lights to the photographic picture by increasing the compatibility of the given photos. To get each of the height lines of the sum and interpolation operations, we then get a smooth three-dimensional reconstruction of the surface of the object. The latter part of the research process will focus on three-dimensional high-precision, high-speed, and real-time reconstruction for further study.

Table 1: Comparison of actual height and reconstruction height.

| | Hemisphere | rectangular | Half column |
|---|---|---|---|
| Actual height | 3.4 | 1.8 | 2.4 |
| Reconstruction height | 3.45 | 1.89 | 2.37 |

## Data Availability

The datasets used in the experiment are from previously reported studies and datasets, which have been cited.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

The study was supported by a Project of Shandong Province Higher Educational Science and Technology Program (no. J14LN64).

## References

[1] X. Hao, Z. Sun, and W. Li, "3D road reconstruction research based on structured light," *Computer Engineering and Design*, vol. 36, no. 8, pp. 2303–2307, 2015.

[2] X. Luo and Y. Fan, "Three-dimensional reconstruction based on multi-view synchronization imagining," *Computer and Digital Engineering*, vol. 44, no. 2, pp. 317–330, 2016.

[3] Z. Yang and H. Song, "3D reconstruction of ancient cultural relics based on SFS method," *Geotechnical Investigation and Surveying*, vol. 1, pp. 67–70, 2018.

[4] S. Yi, Z. He, and P. Wang, "Research on 3d reconstruction based on structured light," *Electronic Technology*, vol. 8, pp. 15–18, 2017.

[5] S. Wang, Z. Zeng, and C. Li, "A survey of 3d reconstruction based on structured light scanning," *Journal of Beijing Institute of Graphic Communication*, vol. 24, no. 2, pp. 66–74, 2016.

[6] S. Pathak, A. Moro, H. Fujii, A. Yamashita, and H. Asama, "3D reconstruction of structures using spherical cameras with

small motion," in *Proceedings of the 2016 16th International Conference on Control, Automation and Systems (ICCAS)*, pp. 117–122, Gyeongju, South Korea, October 2016.

[7] G. Yan and J. Yan, "On calibration method in a three-dimensional reconstruction system based on structured light vision," *Journal of Liming Vocational University*, vol. 88, no. 3, pp. 83–88, 2015.

[8] L. Yang and J. Yuan, "The 3D surface measurement and simulation for turbine blade surface based on color encoding structural light," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 8, no. 3, pp. 273–280, 2015.

[9] B. K. P. Horn and M. J. Brooks, "The variational approach to shape from shading," *Computer Vision Graphics and Image Processing*, vol. 33, no. 2, pp. 174–208, 1986.

[10] K. Ikeuchi and B. K. P. Horn, "Numerical shape from shading and occluding boundaries," *Artificial Intelligence*, vol. 17, no. 1-3, pp. 141–184, 1981.

[11] F.-Q. Zhou and G.-J. Zhang, "New method for calibrating cross structured-light sensor," *Opto-Electronic Engineering*, vol. 33, no. 11, pp. 52–56, 2006.

[12] D. Zhao, S. Chen, and L. Wu, "Analysis and realization of the calculus of height from a single image," *Computer Science*, vol. 23, no. 2, pp. 147–152, 2000.

[13] R. Liu and J. Han, "lgorithm of Shape Recovery Without Highlight and Shadow Constraints," *Journal of Xi'an Jiaotong University*, vol. 40, no. 8, pp. 892–896, 2006.

[14] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[15] J. Heikkila and O. Silven, "A four-step camera calibration procedure with implicit image correction," in *Proceedings of the 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1106–1112, IEEE, San Juan, Puerto Rico, USA, 1997.

[16] R. J. Woodham, "Photometric method for determining surface orientation from multiple images," *Optical Engineering*, vol. 19, no. 1, pp. 139–144, 1980.

[17] T. Wei and R. Klette, "Height from gradient with surface curvature and area constraints," in *Proceedings of the 3rd Indian Conference on Computer Vision Graphics and Image (ICVGIP 2002)*, pp. 52–60, Allied Publishers Private Limited, Ahmadabad, India, 2002.

[18] Y. Liu, L. Zhang, and F. Zhu, "Development of simulation software for laser synchronous scanning triangulation system," *Machinery*, vol. 52, no. 602, pp. 68–72, 2014.

[19] F. Cao and Y. Zhu, "3D Reconstruction Based on SFS Method and Accuracy Analysis," *Computer Science*, vol. 44, no. S1, pp. 244–247, 2017.

[20] G. Guo and H. Wei, "Reconstruction of Surface Morphology and Roughness Detection Based on Shading Shape," *Tool technology*, vol. 45, no. 6, pp. 98–102, 2011.

[21] W. Lun, W. Yong-tian, and L. Yue, "A Robust Approach Based on Photometric Stereo for Surface Reconstruction," *Acta Automatica Sinica*, vol. 39, no. 8, pp. 1339–1348, 2013.

[22] Q. Liu, X. Qin, and S. Ying, "Structural Parameter Design and Accuracy Analysis of Binocular Vision Measuring System," *China Mechanical Engineering*, vol. 19, no. 22, pp. 2728–2732, 2008.

[23] Z. Huang and X. Xu, "Research on precision of 3D restoration based on horopter and structural light," *Transducer and Microsystem Technologies*, vol. 37, no. 5, pp. 16–22, 2018.

[24] Y. Yin, D. Xu, and Z. Zhang, "Plane measurement based on monocular vision," *Journal of Electronic Measurement & Instrument*, vol. 27, no. 4, pp. 347–352, 2013.

Hindawi
Advances in Multimedia
Volume 2018, Article ID 7479316, 8 pages
https://doi.org/10.1155/2018/7479316

*Research Article*

# Region Space Guided Transfer Function Design for Nonlinear Neural Network Augmented Image Visualization

**Fei Yang**,[1,2] **Xiangxu Meng**,[1] **JiYing Lang**,[2] **Weigang Lu**,[3] **and Lei Liu**[4]

[1]*School of Computer Science and Technology, Shandong University, Jinan 250101, China*
[2]*School of Mechanical, Electrical & Information Engineering, Shandong University, Weihai, 264209, China*
[3]*Department of Educational Technology, Ocean University of China, Qingdao, 266100, China*
[4]*The Institute of Acoustics of the Chinese Academy of Sciences, Beijing, 100190, China*

Correspondence should be addressed to Xiangxu Meng; mxx@sdu.edu.cn

Received 6 July 2018; Accepted 12 September 2018; Published 1 November 2018

Guest Editor: Shengping Zhang



Visualization provides an interactive investigation of details of interest and improves understanding the implicit information. There is a strong need today for the acquisition of high quality visualization result for various fields, such as biomedical or other scientific field. Quality of biomedical volume data is often impacted by partial effect, noisy, and bias seriously due to the CT (Computed Tomography) or MRI (Magnetic Resonance Imaging) devices, which may give rise to an extremely difficult task of specifying transfer function and thus generate poor visualized image. In this paper, firstly a nonlinear neural network based denoising in the preprocessing stage is provided to improve the quality of 3D volume data. Based on the improved data, a novel region space with depth based 2D histogram construction method is then proposed to identify boundaries between materials, which is helpful for designing the proper semiautomated transfer function. Finally, the volume rendering pipeline with ray-casting algorithm is implemented to visualize several biomedical datasets. The noise in the volume data is suppressed effectively and the boundary between materials can be differentiated clearly by the transfer function designed via the modified 2D histogram.

## 1. Introduction

Since there are the two characteristics of visibility of object and clear detail revealing, visualization has been proven to be of paramount important for exploring meaningful properties of volume data [1]. Because of the ability of obtaining the two-dimensional rendering results on the screen directly from the data field without building a network model in advance, volume rendering is thus certified to be an effective visualization method of extracting underlying information of interest from volumetric data using interactive graphics and imaging [2, 3]. Kniss et al. [4] visualized the muscle, soft tissues, and the bone from the visible male head data using the volume rendering method and produced a set of direct manipulation widgets to make exploring such features convenient. Ching and Chang [5] rendered the feature of interest in CT (Computed Tomography) images and generated a large wide-angle perspective projection view in an endoscopy to help a physician in diagnosis. Zhang et al. [1] synchronized the dual-modality of cardiac MRI (Magnetic Resonance Imaging) and 3D ultrasound volumes and visualized the dynamic heart by 4D cardiac image rendering. Based on volume rendering. Zhang and Wang et al. developed a platform integrating multivolume visualization method for both heart anatomical data and electrophysiological data visualization [6]. Hsieh et al. [7] visualized the three-dimensional (3D) geometry of the ear ossicle with the segmented ossicle computer tomography (CT) slices, which presented the spatial relation with the temporal bone to diagnose middle ear disease. To visualize brain activity conveniently, Holub and Winer [8] performed 3D and 4D volume ray casting on a tablet device in real-time.

Transfer function plays a fundamental role in visualization for its capability of classifying and segmenting features of volume data, which may affect the quality of rendering image and the perception of users to volume data. To measure through-plane MR flow. Thunberg et al. [9] presented a visualization method which combined the magnitude and

velocity images into one single image. By using the transfer function, the velocities are color-coded and set to a predefined opacity. How the measured blood flow was related to the underlying anatomy can thus be understood. Zhang presented a statistics-based method to visualize 3D cardiac volume data set [10] and further proposed a novel transfer function design approach for revealing detailed structures in the human heart anatomy via perception-based lighting enhancement [11]. Yang presented a fusion visualization framework to combine the cardiac electrophysiology pattern with the anatomy pattern through a novel multidimensional fusion transfer function [12].

Ebert et al. [13] studied the accuracy of volume rendering for arterial stenosis measurement and the results suggested that the choice of transfer function parameters greatly affects the accuracy of volume rendering, while accurate transfer function parameters selection is still a challenge due to the lack of meaningful guidance information and intuitive user interface. The present methods for this problem mainly are object-centric, image-centric, and data-centric [14]. Object-centric approach first classifies or segments the volume data through clustering, probability, and machine learning which covers artificial neural network, support vector machine, and hidden Markov model [15–18]. Then the optical parameters are specified based on the classification result.

Different from the object-centric method, the image centric transfer function is designed on the rendered images. Through the evaluation of the projective images, parameters of transfer function are automatically adjusted and reapplied to the original data recursively until the satisfied rendering result is achieved. Based on a set of rendered images, He et al. [19] presented the stochastic method to search the satisfied transfer function. Marks et al. [20] proposed the Design Gallery method to provide the user varieties of ordered graphics or animations with different perceptions, which are generated automatically by a series of transfer functions given input parameter vector. Users then explore these images space to search for the satisfactory transfer function.

In data-centric approach, parameters of transfer function are specified by analyzing the volume data. Generally, collecting additional information related to the data prior to confirming transfer function makes the design more convenient. Scalar value of volume data is commonly considered for deriving 1D transfer function. The gradient [21, 22] and curve [23, 24] are introduced as the second variable for the two-dimensional transfer function. Roettger et al. [25] extended the variable of transfer function to spatial information. Spatial regions connected with each other were grouped and thus classified. Huang et al. [26] added spatial information into the transfer function domain and extended the number of dimensions to three. Material boundaries are accurately revealed by taking advantage of a designed cost function in three dimensions to imply regional growth algorithm. Some approaches were proposed to classify the topological structure of volume data for the transfer function design [27, 28]. Through the continuous scale-space analysis and detection filters, Correa et al. [29] obtained the 3D scale fields which represent the scale of every voxel. The size-based transfer function is then proposed using the scale fields and maps the scale of local features to color and opacity. Thus the features with similar scalar values in the complex data can be classified based on the relative size. With the increasing dimensions of transfer function, specifying its parameters properly becomes a more difficult and tedious task.

When the dimension is more than 2, it is difficult to specify the parameters for the higher dimensional transfer functions. The histograms are often used to find satisfied transfer functions. Based on the first and second derivatives in the volume, Kindlmann and Durkin [22] built a 2D histogram and the object boundaries appear as arcs in the histogram. A feature-sensitive transfer function can then be semiautomatically generated according to the arcs to reveal features of interest. Lum et al. [30] employed gradient-aligned samples instead of first derivatives as the first property for creating a variant of 2D histogram. The transfer function designed through the histogram classifies the voxels with different degree of homogeneity by mapping them to different optic parameters. However, with an increasing number of boundaries, their separation based on above methods becomes more difficult due to intersections and overlaps.

In this paper, firstly a neural network volume data preprocessing approach for slice denoising is implemented to improve the quality of 3D biomedical data. Then a two-dimensional transfer function with the preprocessed data is designed based on a modified 2D histogram, which is created using a novel region space based method with depth information. The features of interest in the data are thus exactly explored. In Section 2 of this paper, the method for denoising is described and a two-dimensional transfer function based on 2D histogram is designed. Efficiency and practicability of the presented method are further shown in Section 3. Finally, conclusions are discussed in Section 4.

## 2. Transfer Function Design

*2.1. Augmentation on Slice Data.* Biomedical volume data produced by current noninvasive devices such as CT and MRI scanners are usually accompanied by noisy, partial effect, and bias. Data with serious noise or error message which causes low SNR (Signal Noise Ratio) will directly affect transfer function specification and cause the objects obscuring in the resulting image.

Spatial mean low-pass filtering such as General Median filtering and Gaussian smoothing has the advantage of reducing the amplitude of noise fluctuations. While the filtering blurs the details in the data such as the line or edge and does not focus on processing regional boundary or tiny structures, which makes the resulting image too fuzzy. This is a hamper to effectively enhance boundary for those noisy data containing lots of details.

Although nonlinear filtering has the achievement of reserving the edge, it produces the loss of resolution due to the suppression of details. To solve this problem, the nonlinear enhancement algorithm uses information of boundary and the neighbor of a pixel to preprocess the image data [14, 31], which effectively removes the noise region with homogeneous physical properties and significantly improves

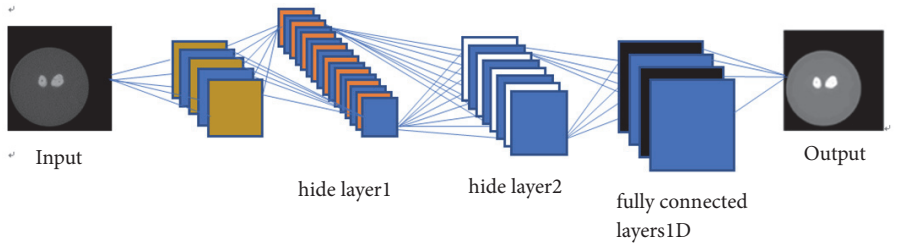



Figure 1: The architecture of the MLP network.

the image quality. Loss of information can thus be minimized by reserving the object boundary and the detailed structure and the shape is enhanced by discontinuous sharpening. Since the anisotropic diffusion filtering smoothes the image along the edge direction rather than in the orthogonal direction to the edge, location and intensity of the edge can be retained. Unlike traditional methods, neural network can learn more features beneficial to the task through hierarchical structure. Based on a multilayer perceptron neural network, Burger et al. [32] presented a denoising algorithm that is learned on a large dataset for image denoising. For the multilayer perceptron (MLP), it can be represented as a nonlinear function which maps a noisy image to a noise-free image:

$$h(\mathrm{u}) = \alpha_3 + W_3 \cdot \Theta\left(\alpha_2 + W_2 \cdot \Theta\left(\alpha_1 + W_1 \cdot \mathrm{u}\right)\right) \quad (1)$$

Here the network has three hidden layers. $\alpha_1$, $\alpha_2$, $\alpha_3$ are vector-valued biases. The weight matrices of the structure are $W_1$, $W_2$, $W_3$. The function $\Theta$ operates component-wise. In order to realize image denoising, the clean images are selected from the image dataset and the input noise level is employed to produce the corresponding noisy images. The MLP parameters are then estimated by the back propagation algorithm satisfying:

$$\arg\min_{\alpha,W} \|h(\mathrm{u}) - \mathrm{v}\|^2 \quad (2)$$

where $\mathbf{u}$ is the vector-valued noisy images input and $h(\mathbf{u})$ is the mapped vector-valued denoised images output. $\mathbf{v}$ is the clean images. The architecture of the network is as Figure 1.

During application for image denoising, MLP uses fully connected neural network to process image fragments and then splits and combines all processed image segments to form a denoising image. First the noisy image is split into overlapping patches and each patch $\mathbf{u}$ is denoised separately. Then the denoised patches $h(\mathbf{u})$ are placed at the locations of their noisy counterparts. The denoised image is thus obtained by averaging on the overlapping regions.

*2.2. Region Space Guided Transfer Function Design.* Kindlmann et al. [22] added higher-order derivatives of the voxel to transfer function domain in his presented approach. Those extracted boundaries appeared as arches in the derived histogram with axes representing scalar value and gradient magnitude. Although using this histogram can improve selection of boundaries, intersection or overlapping of two arches caused by different feature voxels sharing the same scalar value and gradient magnitude may result in ambiguities in classification of boundaries. Sereda et al. [33] proposed a multidimensional transfer function based on the LH histogram to facilitate separation of features which are represented by the arches.

LH Histogram based method computes low and high values of each sample voxels which are labeled as *FL* and *FH* respectively. For each sample voxel, if the gradient is less than the threshold, the voxel is identified to be internal sample. Otherwise the voxel is considered to be the boundary element. *FL* and *FH* of the internal voxel are equal and the value is the scalar of the voxel. For the voxels which are supposed to belong to boundaries, integration is implemented along the gradient and reverse direction in gradient field until the gradient is less than the threshold. *FL* and *FH* can then be found. The *FL* and *FH* values of all voxels are expressed in the same coordinate system, and the LH histogram is thus obtained. Since the value of *FL* is not more than *FH*, points in the LH histogram are only located above the diagonal line. The points on the diagonal indicate the internal voxels; that is, the *FL* and *FH* values are equal. The rest represent the boundary voxels and the corresponding *FL* and *FH* values are the scalar value of the two materials of the boundary respectively.

In volume rendering, using LH method to design transfer function can not only reduce the dependence on image segmentation, but also include voxel gradient information and boundary gray information. Due to the characteristics of medical data and clinical application, centralization of the voxel is required.

A proper region space $\Omega$ is selected and a voxel is compared with the voxels in $\Omega$. Let $V(p)$ be the scalar or intensity value of a voxel $p$. The intensity mean $m$ and variance $v$ of all voxels within $\Omega$ are given the following, respectively:

$$m = \frac{1}{n}\left(\sum_{p_i \in \Omega} V(p_i) + V(p)\right) \quad (3)$$

$$v = \frac{1}{n}\left(\sum_{p_i \in \Omega} \|V(p_i) - m\| + \|V(p) - m\|\right) \quad (4)$$

where $p_i$ represents the adjacent voxel in the region space of voxel $p$ and $n$ is the number of voxels in $\Omega$. The criteria for

**Input:** Anisotropic diffused volume data.
**Output:** Visualization result of biomedical volume data.
**for** each voxel $p(x,y,z)$ **do**
  chose adjacent voxels in the region space $\Omega$;
  calculate the intensity median $m$ of $p$;
  calculate the variance $v$ of $p$;
  set the value of estimation radius $r$;
  **if** $v < r$ **then**
   $p$ is marked as an inner voxel;
   $FL = FH = f(p)$;
  **else**
   $p$ is considered to be the boundary voxel;
   use the second order Runge-Kutta method to search $FL$ and $FH$ value of $p$;
  **end**
**end**
compute the depth information for each voxel;
construct the depth LH histogram and design the transfer function;
visualize the volume data according to the transfer function;

Algorithm 1: Region space guided visualization.

identification boundary voxels can then be formulated as in (5):

$$\begin{aligned} p &\in S_{inner} \quad v < r \\ p &\in S_{boundary} \quad otherwise \end{aligned} \tag{5}$$

where $S_{inner}$ is the set of voxels inside the materials and $S_{boundary}$ is the set of voxels on the boundaries. Thus the voxel that difference between it and the voxels in $\Omega$ falls out of the range of $r$ is considered to the boundary voxel.

In the region where the complex boundary exists, using the single criterion will result in boundary determination error. Since some boundaries only appear at a certain depth and then disappear when they reach a certain depth, the complex boundaries can be further differentiated according to the depth information. Then a modified 2D histogram is created using the region space based method with depth information. In this paper the points in the original histogram are further grouped according to the corresponding depth.

A 2D transfer function can then be specified based on the created LH histogram by selecting relevant areas and by assigning them color and opacity. The corresponding features in the volume data can thus be explored. The details of the proposed method are given in Algorithm 1.

## 3. Results and Discussion

In this section, some data sets are used as the test data, including tooth data and sheep heart data, to evaluate the performance of the proposed transfer function. The size of data set is 256×256×161 and 352×352×256, respectively. All the experiments are carried out on the computer with Intel Core i5 2.66G, 4.00G RAM and graphics card of NVIDIA GeForce GT 650.

Biomedical volume data produced by current noninvasive devices such as CT and MRI scanners are usually accompanied by serious noise, which will generate poor visualized image and cause the blur objects in the resulting image. Thus the MLP neural network is implemented to denoise the volume data. In the experiments, the union of the LabelMe dataset is used to train MLP which contains approximately 150,000 images. Before training, the data are filled with padding operation and each pixel is filled with 6 pixel sizes. The noise level $\sigma$ is set to 10. We use a patch of size 39×39 to generate the predicted patch and then adopt a filter of size 9×9 to average the output patches, thus its effective patch size is 47×47. In the experiment the learning rate $r$ in each layer is equal to $r/\mathrm{N}$ and N is the number of input units of current layer. The basic learning rate was set to 0.1. To improve results slightly we use the sliding window method with stride size of 3 which weights denoised patches with a Gaussian window instead of using all possible overlapping patches. Figure 2 compares the image denoising results of nonlinear enhancement and MLP neural network on the tooth data. The Peak Signal to Noise Ratio (PSNR) with the nonlinear enhancement filtering is 41.4294, and the Structural Similarity Index (SSIM) is 0.9325. The PSNR with the MLP method is 41.4294, and SSIM is 0.9325. As one can see, the MLP network produces more visually pleasant results. Compared with original data, noise in the homogeneous region is suppressed and the quality of the image is improved.

In LH histogram, points around diagonal represent interior of materials. Regions including those points are thus assigned to lower opacity in the transfer function to fade unimportant information out. Remainder regions are the accumulation of boundary voxels which contain features of interest. Figure 3 shows LH histogram and rendering result of the original tooth data and MLP denoised data. Figure 3(a) describes the created LH histogram (left) and the corresponding rendering result with original tooth data set (right). Figure 3(b) shows the LH histogram (left) based on the denoised data. From Figure 3(b), we can see the more compact separation of points in the LH histogram, which is

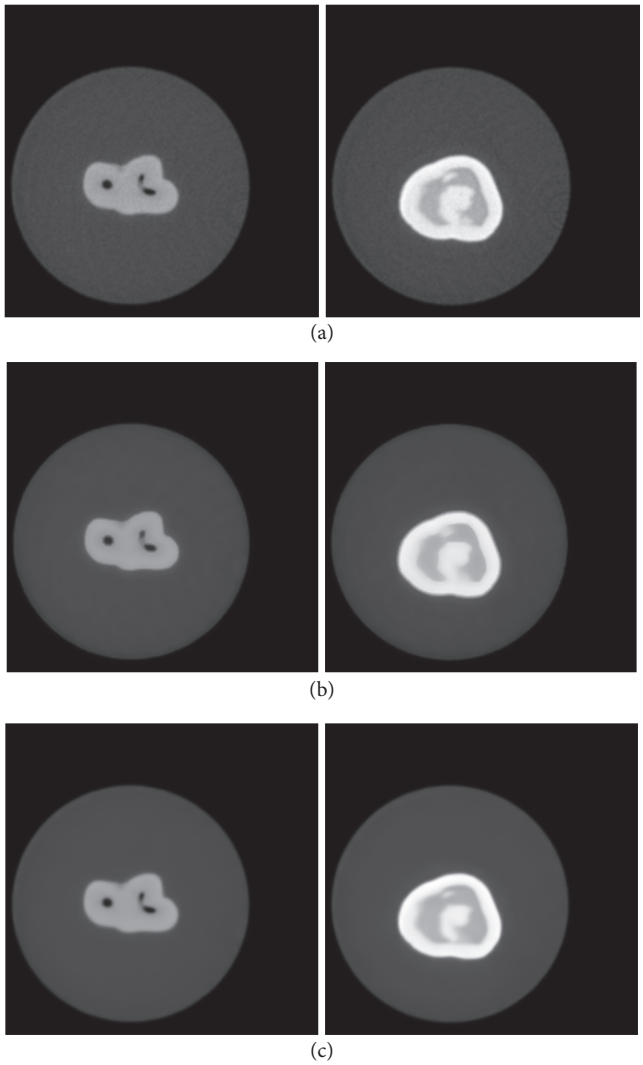


Figure 2: Preprocessing results of tooth dataset. From top to bottom: original slice data, denoised images by the nonlinear enhancement algorithm, and denoised images by MLP. (a) Original slice data; (b) nonlinear enhancement denoised result; (c) MLP denoised result.

on account of the suppression of noise in the homogeneous region and enhancement of the boundaries between materials. Since the noise is suppressed effectively, boundaries between different materials can be visualized more clearly. As shown in Figure 3(b), the fact which is specifically manifested through the experiment result is that dentine-enamel (yellow) is explored exactly and noises around the root of the dentine are considerably removed.

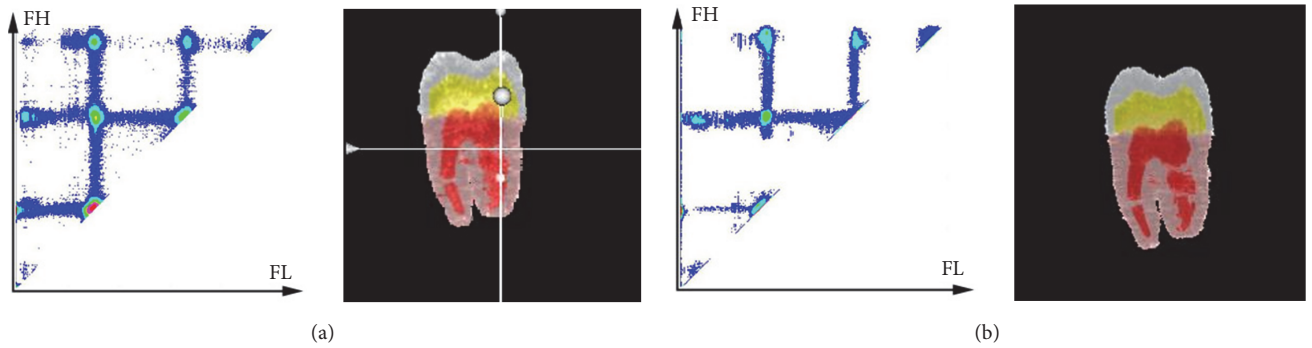
Figure 4 shows the created LH histogram and rendering result on the anisotropic diffusion enhanced tooth volume data using the conventional and regional criteria based method. The number of iteration of the nonlinear filtering is the process ordering parameter. Figure 4(a) presents the visualization result of enhanced data using the conventional method. The gradient magnitude threshold for investigating the *FL* and *FH* intensity profile is set to 10. Figure 4(b) shows the LH histogram constructed and corresponding rendering images through regional criteria based method. Here the region range $r$ is set to 1.6. As shown in Figure 4(b), since the noise is suppressed and the path tracing for *FL* and *FH* value starts with regional criteria, the distribution of separated parts of points that correspond to different features is more concentrate in LH histogram, which thus ensures a more accurate identification of boundary voxels. The fact which is specifically manifested through two experiment results is that noises around the root of the dentine are considerably removed, and the phenomenon of blurred boundary is removed and various boundaries of the tooth, i.e., enamel-air (white), dentine-enamel (yellow), pulp-dentine (red), and dentine-air (pink) boundary, are revealed clearly in the final image.

Figure 5 shows the rendering result after introducing depth information into region space with the MLP augmented data. Classifying boundary voxels through conventional LH histogram will result in the confused boundary exploration. From Figure 4, we can see that there exists a visible discontinuity in the pulp-dentine boundary. This discontinuity is due to classifying those boundary voxels of the dentine tissue falsely. Because the boundary appears at a specific depth, for example, the tooth enamel is in the region near the human eye, while the medulla is at the depth far away from the human eye, thus we add depth information for the histogram construction and can obtain the distinct boundary. In our experiment the depth of enamel is about 120 and the depth of medulla is about 80. As shown in the result image in Figure 5(a), since more voxels are classified into boundary voxels via the transfer function which is designed based on LH histogram with depth information in region space properly, the discontinuity of the pulp-dentine boundary (red) in Figure 4 is corrected. And the exact pulp-dentine boundary is revealed in the rendering result image. The rendering time of the proposed method is 1.1s, which enriches real-time interaction property of visualization.

Figure 6 shows the denoising result of sheep heart slice data and gives the rendering result via the proposed region space guided transfer function. Figure 6(a) shows the original slices. In Figure 6(b), the two corresponding denoised results are presented. It is obvious that preprocessing for tissues of the sheep heart such as the muscle and the fat is effective in decreasing noise and the boundary details are consequently remained. Fine structural features of interesting in the sheep heart are thus clearly visualized through the region space based transfer function with the augmented data and, as in Figure 6(c), the fat of sheet heart is colored in yellow and the muscle is red. The profile of sheep heart is colored by white. From the result, structures of the sheep heart are thus exactly explored and the shape, spatial position, and relationship of the tissues can be observed without ambiguity. The rendering time of the sheep heart data set is 2.6s.

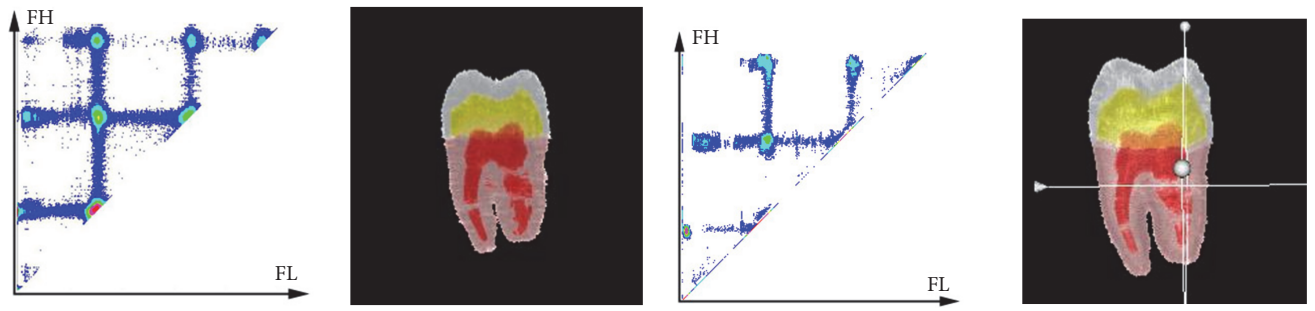
## 4. Conclusion

Transfer function in performance of volume rendering plays a crucial role for exploring directly detail information hiding



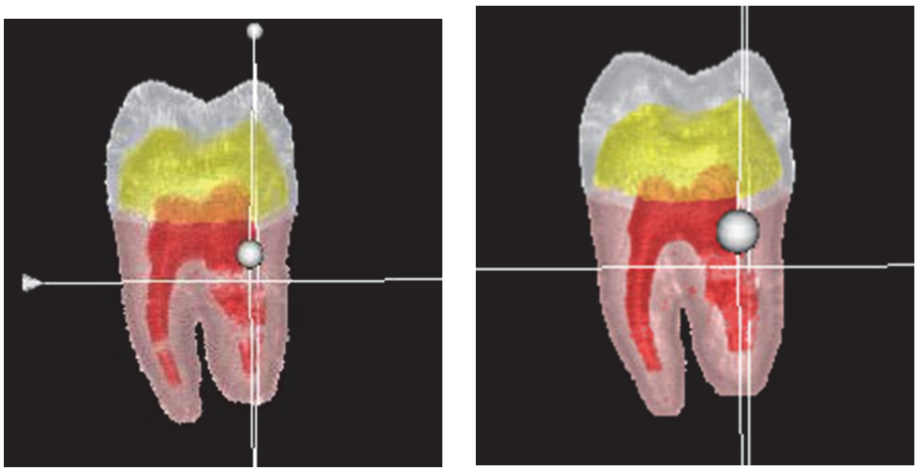

(a) (b)

Figure 3: LH histogram and rendering result of the original tooth data and MLP denoised data: (a) LH histogram and the corresponding rendering result of original data and (b) LH histogram and corresponding rendering result of denoised data.




(a) (b)

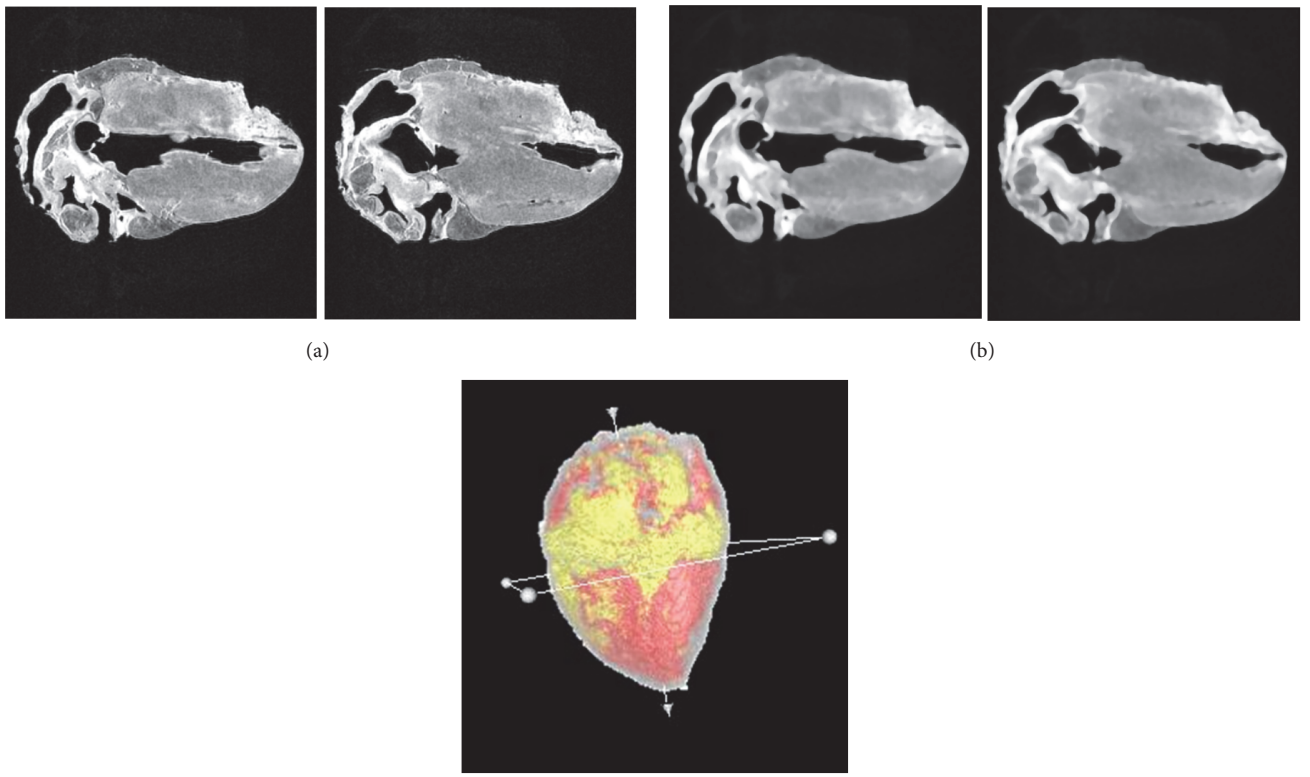
Figure 4: LH histogram and corresponding rendering result with nonlinear enhanced tooth dataset through the conventional and region criteria based method: (a) LH histogram based on conventional method with gradient threshold of 5 and the corresponding rendering result and (b) LH histogram based on region criteria based method and the rendering result.



(a)

Figure 5: Rendering result of MLP augmented tooth dataset with two methods: (a) rendering result based on region criteria based method and (b) rendering result based on depth enhanced method.

in data as well as enhancing important boundaries. In this work we first implement the MLP neural network on volume data to denoise while preserve the boundary. This method can considerably improve quality of volume data acquired by devices. Then we improve the LH method by combining the regional depth information to achieve the transfer function semiautomatic generation. This method can avoid the influence of noise and make the voxels more centralized. In the LH histogram the voxel distribution at the diagonal line is more concentrated, and the boundary of important



(a) (b)

(c)

Figure 6: Visualization result of augmented sheep heart dataset with region space guided transfer function: (a) the original sheep heart data; (b) the denoised data; (c) rendering result of sheep heart with the denoised data.

objects are effectively emphasized. The features of interest in the data can thus be found exactly by mapping scalar value of boundary voxels which correspond to the points in LH histogram to appropriate opacity and color.

## Data Availability

The two datasets are both open data which are available at http://visual.nlm.nih.gov/.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

The work was supported by the National Natural Science Foundation of China (NSFC) under Grant no. 61502275 and the Postdoctoral Science Foundation of China (no. 2017M622210). This work was also supported in part by the Natural Science Foundation of Shandong of Grant no. ZR2017MF051, the National Natural Science Foundation of China (NSFC) of Grant no. 61501450, and the MOE (Ministry of Education in China) Project of Humanities and Social Sciences of Grant no. 16YJC880057.

## References

[1] Q. Zhang, R. Eagleson, and T. M. Peters, "GPU-based visualization and synchronization of 4-D cardiac MR and ultrasound images," *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, no. 5, pp. 878–890, 2012.

[2] H. Pfister, B. Lorensen, C. Bajaj et al., "The transfer function bake-off," *IEEE Computer Graphics and Applications*, vol. 21, no. 3, pp. 16–22, 2001.

[3] Q. Zhang, R. Eagleson, and T. M. Peters, "Volume visualization: A technical overview with a focus on medical applications," *Journal of Digital Imaging*, vol. 24, no. 4, pp. 640–664, 2011.

[4] J. Kniss, G. Kindlmann, and C. Hansen, "Multidimensional transfer functions for interactive volume rendering," *IEEE Transactions on Visualization and Computer Graphics*, vol. 8, no. 3, pp. 270–285, 2002.

[5] Y.-T. Ching and C.-L. Chang, "A volume rendering technique to generate a very large wide-angle endoscopeic view," *Journal of Medical and Biological Engineering*, vol. 22, no. 2, pp. 109–112, 2002.

[6] L. Zhang, C. Gai, K. Wang, W. Lu, and W. Zuo, "GPU-based high performance wave propagation simulation of ischemia in anatomically detailed ventricle," in *Proceedings of the Computing in Cardiology Conference (CinC '11)*, pp. 469–472, Hangzhou, China, September 2011.

[7] M. S. Hsieh, F. P. Lee, and M. D. Tsai, "A virtual reality ear ossicle surgery simulator using three-dimensional computer tomography," *Journal of Medical and Biological Engineering*, vol. 30, no. 1, pp. 57–63, 2010.

[8] J. Holub and E. Winer, "Enabling Real-Time Volume Rendering of Functional Magnetic Resonance Imaging on an iOS Device," *Journal of Digital Imaging*, vol. 30, no. 6, pp. 738–750, 2017.

[9] P. Thunberg and A. Kähäri, "Visualization of Through-Plane Blood Flow Measurements Obtained from Phase-Contrast MRI," *Journal of Digital Imaging*, vol. 24, no. 3, pp. 470–477, 2011.

[10] L. Zhang, K. Wang, F. Yang et al., "A Visualization System for Interactive Exploration of the Cardiac Anatomy," *Journal of Medical Systems*, vol. 40, no. 6, 2016.

[11] L. Zhang, K. Wang, H. Zhang, W. Zuo, X. Liang, and J. Shi, "Illustrative cardiac visualization via perception-based lighting enhancement," *Journal of Medical Imaging and Health Informatics*, vol. 4, no. 2, pp. 312–316, 2014.

[12] F. Yang, W. G. Lu, L. Zhang, W. M. Zuo, K. Q. Wang, and H. G. Zhang, "Fusion visualization for cardiac anatomical and ischemic models with depth weighted optic radiation function," in *Proceedings of the Computing in Cardiology Conference (CinC '15)*, pp. 937–940, IEEE, Nice, France, September 2015.

[13] D. S. Ebert, D. G. Heath, B. S. Kuszyk et al., "Evaluating the potential and problems of three-dimensional computed tomography measurements of arterial stenosis," *Journal of Digital Imaging*, vol. 11, no. 3, pp. 151–157, 1998.

[14] G. Gerig, O. Kubler, R. Kikinis, and F. A. Jolesz, "Nonlinear anisotropic filtering of MRI data," *IEEE Transactions on Medical Imaging*, vol. 11, no. 2, pp. 221–232, 1992.

[15] F.-Y. Tzeng and K.-L. Ma, "A cluster-space visual interface for arbitrary dimensional classification of volume data," in *Proceedings of the in Proceedings of the 6th Joint Eurographics-IEEE TCVG Symposium on Visualization*, pp. 17–24, Konstanz, Germany, 2004.

[16] P. Šereda, A. Vilanova, and F. A. Gerritsen, "Automating transfer function design for volume rendering using hierarchical clustering of material boundaries," in *Proceedings of the in Proceedings of the 8th Joint Eurographics-IEEE TCVG Symposium on Visualization*, pp. 243–250, Lisbon, Portugal, 2006.

[17] F.-Y. Tzeng, E. B. Lum, and K-L. Ma, "A novel interface for higher-dimensional classification of volume data," in *Proceedings of the IEEE Visualization Conference (VIS '03)*, pp. 505–512, Seattle, WA, USA, 2003.

[18] F.-Y. Tzeng, E. B. Lum, and K.-L. Ma, "An intelligent system approach to higher-dimensional classification of volume data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 11, no. 3, pp. 273–283, 2005.

[19] T. S He, L. C. Hong, A. Kaufman, and et al, "Generation of transfer functions with stochastic search techniques," in *Proceedings of the Seventh Annual IEEE Visualization '96*, pp. 227–234, San Francisco, CA, USA.

[20] J. Marks, B. Mirtich, B. Andalman et al., "Design Galleries: A general approach to setting parameters for computer graphics and animation," in *Proceedings of the 1997 Conference on Computer Graphics, SIGGRAPH*, pp. 389–400, August 1997.

[21] M. Levoy, "Display of surfaces from volume data," *IEEE Computer Graphics and Applications*, vol. 8, no. 3, pp. 29–37, 1988.

[22] G. Kindlmann and J. W. Durkin, "Semi-automatic generation of transfer functions for direct volume rendering," in *Proceedings of the 1998 IEEE Symposium on Volume Visualization, VVS 1998*, pp. 79–86, USA, October 1998.

[23] J. Hladůvka, A. König, and E. Gröller, "Curvature-based transfer functions for direct volume rendering," in *Proceedings of the In Spring Conference on Computer Graphics 2000*, vol. 16, pp. 58–65, 2000.

[24] G. Kindlmann, R. Whitaker, T. Tasdizen, and T. Möller, "Curvature-Based Transfer Functions for Direct Volume Rendering: Methods and Applications," in *Proceedings of the VIS 2003 PROCEEDINGS*, pp. 513–520, USA, October 2003.

[25] S. Roettger, M. Bauer, and M. Stamminger, "Spatialized transfer functions," in *Proceedings of the In Eurographics, IEEE VGTC Symposium on Visualization*, pp. 271–278, 2005.

[26] R. Huang, . Kwan-Liu Ma, P. McCormick, and W. Ward, "Visualizing industrial CT volume data for nondestructive testing applications," in *Proceedings of the IEEE Visualization 2003*, pp. 547–554, Seattle, WA, USA.

[27] I. Fujishiro, T. Azuma, and Y. Takeshima, "Automating transfer function design for comprehensible volume rendering based on 3D field topology analysis," in *Proceedings of the IEEE Visualization '99*, pp. 467–470, October 1999.

[28] S. Takahashi, Y. Takeshima, and I. Fujishiro, "Topological volume skeletonization and its application to transfer function design," *Graphical Models*, vol. 66, no. 1, pp. 24–49, 2004.

[29] C. D. Correa and K.-L. Ma, "Size-based transfer functions: a new volume exploration technique," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1380–1387, 2008.

[30] E. B. Lum and K.-L. Ma, "Lighting transfer functions using gradient aligned sampling," in *Proceedings of the IEEE Visualization 2004 - Proceedings, VIS 2004*, pp. 289–296, USA, October 2004.

[31] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.

[32] H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with BM3D?" in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2012*, pp. 2392–2399, USA, June 2012.

[33] P. Šereda, A. V. Bartrolí, I. W. O. Serlie, and F. A. Gerritsen, "Visualization of boundaries in volumetric data sets using lh histograms," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 2, pp. 208–217, 2006.

Hindawi
Advances in Multimedia
Volume 2018, Article ID 6710865, 10 pages
https://doi.org/10.1155/2018/6710865

*Research Article*

# Can Deep Learning Identify Tomato Leaf Disease?

**Keke Zhang**,[1] **Qiufeng Wu**,[2] **Anwang Liu**,[1] **and Xiangyan Meng**[2]

[1]*College of Engineering, Northeast Agricultural University, Harbin 150030, China*
[2]*College of Science, Northeast Agricultural University, Harbin 150030, China*

Correspondence should be addressed to Qiufeng Wu; qfwu@neau.edu.cn

Received 9 June 2018; Accepted 30 August 2018; Published 26 September 2018

Academic Editor: Alexander Loui



This paper applies deep convolutional neural network (CNN) to identify tomato leaf disease by transfer learning. AlexNet, GoogLeNet, and ResNet were used as backbone of the CNN. The best combined model was utilized to change the structure, aiming at exploring the performance of full training and fine-tuning of CNN. The highest accuracy of 97.28% for identifying tomato leaf disease is achieved by the optimal model ResNet with stochastic gradient descent (SGD), the number of batch size of 16, the number of iterations of 4992, and the training layers from the 37 layer to the fully connected layer (denote as "fc"). The experimental results show that the proposed technique is effective in identifying tomato leaf disease and could be generalized to identify other plant diseases.

## 1. Introduction

Tomato is a widely cultivated crop throughout the world, which contains rich nutrition, unique taste, and health effects, so it plays an important role in the agricultural production and trade around the world. Given the importance of tomato in the economic context, it is necessary to maximize productivity and product quality by using techniques. Corynespora leaf spot disease, early blight, late blight, leaf mold disease, septoria leaf spot, two-spotted spider mite, virus disease, and yellow leaf curl disease are 8 common diseases in tomato [1–8]; thus, a real time and precise recognition technology is essential.

Recently, since CNN has the self-learned mechanism, that is, extracting features and classifying images in the one procedure [9], CNN has been successfully applied in various applications, such as writer identification [10], salient object detection [11, 12], scene text detection [13, 14], truncated inference learning [15], road crack detection [16, 17], biomedical image analysis [18], predicting face attributes from web images [19], and pedestrian detection [20], and achieved the better performance. In addition, CNN is able to extract more robust and discriminative features with considering the global context information of regions [10], and CNN is scarcely affected by the shadow, distortion, and brightness of the natural images. With the rapid development of CNN, many powerful architectures of CNN emerged, such as AlexNet [21], GoogLeNet [22], VGGNet [23], Inception-V3 [24], Inception-V4 [25], ResNet [26], and DenseNets [27].

Training deep neural networks from scratch needs amounts of data and expensive computational resources. Meanwhile, we sometimes have a classification task in one domain, but we only have enough data in other domains. Fortunately, transfer learning can improve the performance of deep neural networks by avoiding complex data mining and data-labeling efforts [28]. In practice, transfer learning consists of two ways [29]. One option is to fine-tune the networks weights by using our data as input; it is worth nothing that the new data must be resized to the input size of the pretrained network. Another way is to obtain the learned weights from the pretrained network and apply the weights to the target network.

In this work, first, we compared the performance between SGD [30] and Adaptive Moment Estimation (Adam) [30, 31] in identifying tomato leaf disease. These optimization methods are based on the pretrained networks AlexNet [21], GoogLeNet [22], and ResNet [26]. Then, the network architecture with the highest performance was selected and experiments on effect of two hyperparameters (i.e., batch size and number of iterations) on accuracy were carried out. Next,

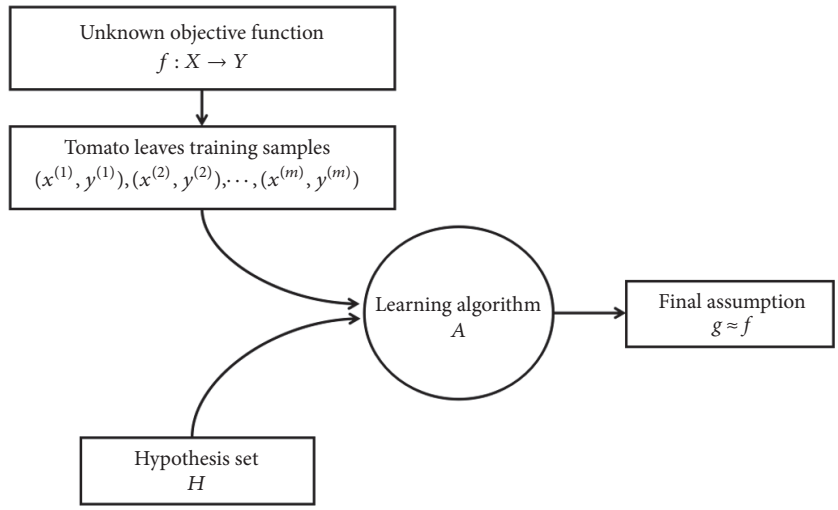



Figure 1: Proposed workflow diagram.

we utilized the network with the suitable hyperparameters, which was obtained from the previous experiments, to discuss the impact of different network structures on recognition tasks. We believe this makes sense for researchers who choose to fine-tune pretrained systems for other similar issues.

The rest of this paper is organized as follows. Section 2 displays an overview of related works. Section 3 introduces the dataset and three deep convolutional neural networks, i.e., AlexNet, GoogLeNet, and ResNet. Section 4 presents the experiments and results in this work. Section 5 concludes the paper.

## 2. Related Work

The research of agricultural disease identification based on computer vision has been a hot topic. In the early years, the traditional machine learning methods and shallow networks were extensively adopted in the agricultural field.

Sannakki et al. [32] proposed to use k-means based clustering performed on each image pixel to isolate the infected spot. They obtained the result that the Grading System they built by machine vision and fuzzy logic is very useful for grading the plant disease. Samanta et al. [33] proposed a novel histogram based scab diseases detection of potato and applied color image segmentation technique to exact intensity pattern. They got the best classification accuracy of 97.5%. Pedro et al. [34] applied fuzzy decision-making to identify weed shape, with fuzzy multicriteria decision-making strategy; they achieved the best accuracy of 92.9%. Cheng and Matson [35] adopted Decision Tree, Support Vector Machine (SVM), and Neural Network to identify weed and rice; the best accuracy they achieved is 98.2% by using Decision Tree. Sankaran and Ehsani [36] used quadratic discriminant analysis (QDA) and k-nearest neighbour (kNN) to classify citrus leaves infected with canker and Huanglongbing (HLB) from healthy citrus leaves; they got the highest overall accuracy of 99.9% by kNN.

Recently, deep learning methods have been applied in identifying plant disease widely. Cheng et al. [37] used ResNet and AlexNet to identify agricultural pests. At the same time, they carried out comparative experiments with SVM and BP neural networks; finally, they got the best accuracy of 98.67% by ResNet-101. Ferreiraa et al. [38] utilized ConvNets to perform weed detection in soybean crop images and classify these weeds among grass and broadleaf. The best accuracy they achieved is 99.5%. Sladojevic et al. [39] built a deep convolutional neural network to automatically classify and detect 15 categories of plant leaf diseases. Meanwhile, their model was able to distinguish plants from their surroundings. They got an average accuracy of 96.3%. Mohanty et al. [40] trained a deep convolutional neural network based on the pretrained AlexNet and GoogLeNet to identify 14 crop species and 26 diseases. They achieved an accuracy of 99.35% on a held-out test set. Sa et al. [41] proposed a novel approach to fruit detection by using deep convolutional neural networks. They adapted Faster Region-based CNN (Faster R-CNN) model, through transfer learning. They got the F1 score with 0.83 in a field farm dataset.

## 3. Materials and Methods

This paper concentrates on identifying tomato leaf disease by deep learning. In this section, the abstract mathematical model about identifying tomato leaf disease is displayed at first. Meanwhile, the process of typical CNN is described with formulas. Then, the dataset and data augmentation are presented. Finally, we introduced three powerful deep neural networks adopted in this paper, i.e., AlexNet, GoogLeNet, and ResNet.

The main process of tomato leaf disease identification in this work can be abstracted as a mathematical model (see Figure 1). First, we assume the mapping function from tomato leaves to diseases is $f : X \longrightarrow Y$ and then send the training samples to the optimization method. The hypothesis set $H$ means possible objective functions with different parameters; through a series of parameters update, we can get the final assumption $g \approx f$.

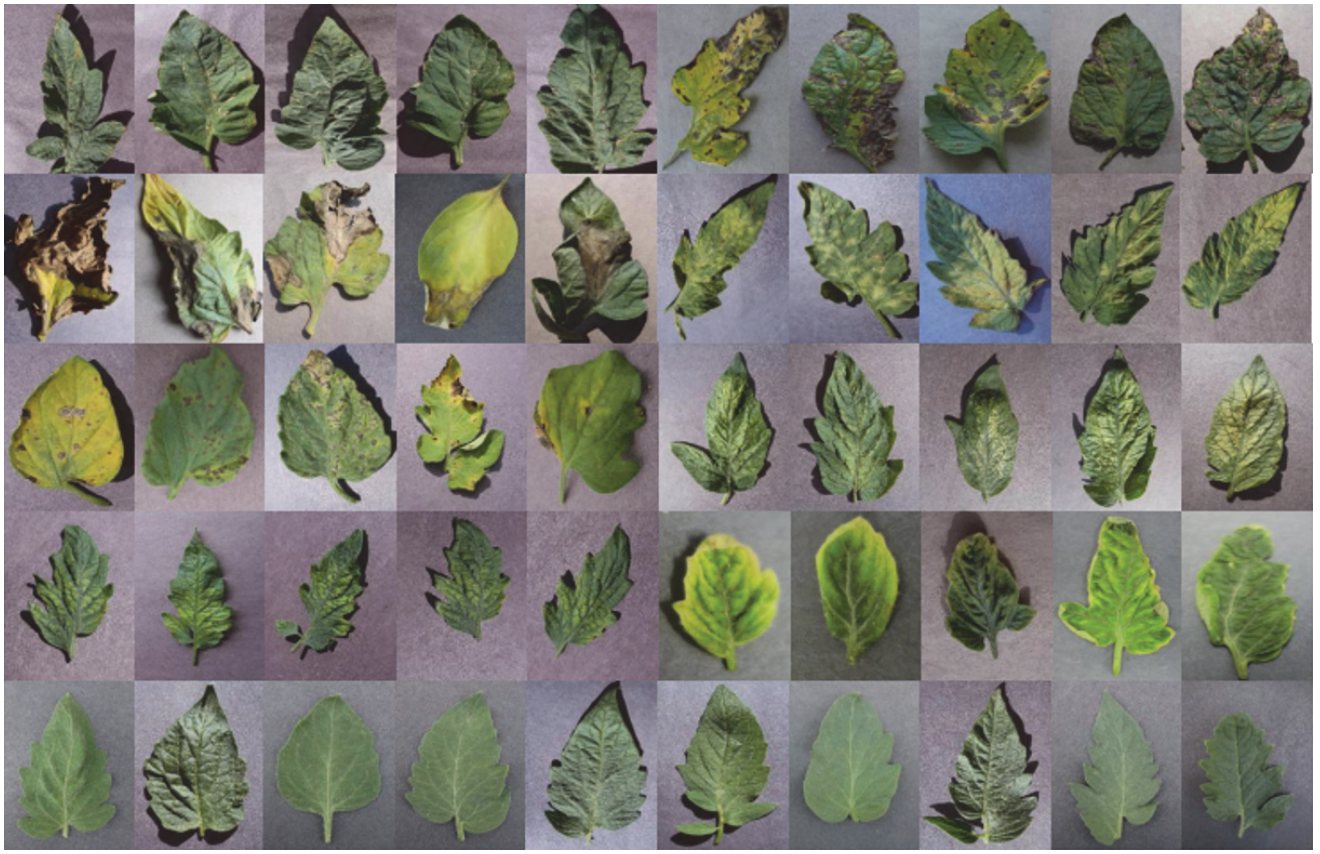


Figure 2: Raw tomato leaf images.

The typical CNN process can be represented with following formulas. Firstly, send the training samples (i.e., training tomato leaf images) to the classifier (i.e., AlexNet, GoogLeNet, and ResNet). Then, convolution operation is carried out; that is, a number of filters slide over the feature map of the previous layer, and the weight matrices do dot product.

$$M_j^l = f\left(\sum_{i \in N_j} M_i^{l-1} * w_j^l + b_j^l\right) \tag{1}$$

where $f(\cdot)$ is activation function, typically a Rectifier Linear Unit (ReLU) [42] function:

$$f(x) = \max(x, 0) \tag{2}$$

$N_j$ is the number of kernels of the certain layer, $M_i^{l-1}$ represents the feature map of the previous layer, $w_j^l$ is the weight matrix, and $b_j^l$ is the bias term.

Max-pooling or average pooling is conducted after the convolution operation. Furthermore, the learned features are sent to the fully connected layer. The softmax regression always follows the final fully connected layer, an input $x$ will get the probability of belonging to class $i$.

$$p(y = i \mid x; \theta) = \frac{e^{\theta_i^T x}}{\sum_{j=1}^{k} e^{\theta_i^T x}} \tag{3}$$

where $y$ is the response variable (i.e., predict label), $k$ is the number of categories, and $\theta$ is the parameters of our model.

*3.1. Raw Dataset.* The raw tomato leaf dataset utilized in this work comes from an open access repository of images, which focus on plant health [43]. Health and other 8 diseases categories are included (see Table 1, Figure 2), i.e., early blight (pathogen: Alternaria solani) [1], yellow leaf curl disease (pathogen: Tomato Yellow Leaf Curl Virus (Tylcv), Family Geminiviridae, Genus Begomovirus) [2], corynespora leaf spot disease (pathogen: Corynespora cassiicola) [3], leaf mold disease (pathogen: Fulvia fulva) [4], virus disease (pathogen: Tomato Mosaic Virus) [5], late blight (pathogen: Phytophthora Infestans)[6], septoria leaf spot (pathogen: Septoria lycopersici) [7], and two-spotted spider mite (pathogen: Tetranychus urticae) [8]. The total dataset is 5550.

*3.2. Data Augmentation.* Deep convolutional neural networks contain millions of parameters; thus, massive amounts of data is required. Otherwise, the deep neural network may be overfitting or not robust. The most common method to reduce overfitting on image dataset is to enlarge the dataset manually and conduct label-preserving transformations [21, 44].

In this work, at first, the raw image dataset was divided into 80% training samples and 20% testing samples, and then the data augmentation procedure was conducted: (1) flipping

Table 1: The raw tomato leaf dataset.

| Label | Category | Number | Leaf symptoms | Illustration |
|---|---|---|---|---|
| 1 | Corynespora leaf spot disease | 547 | Small brown spots appear, leaf spots have yellow halo. | See Figure 1 first row No.1-No.5 |
| 2 | Early blight | 405 | Black or brown spots appear, leaf spots often have yellow or green concentric ring pattern. | See Figure 1 first row No.6-No.10 |
| 4 | Late blight | 726 | Water-soaked area appears and rapidly enlarges to form purple-brown, oily-appearing blotches. | See Figure 1 second row No.1-No.5 |
| 5 | Leaf mold disease | 480 | Irregular yellow or green area appears. | See Figure 1 second row No.6-No.10 |
| 6 | Septoria leaf spot | 734 | Round spots, marginal brown, chlorotic yellow, appear. | See Figure 1 third row No.1-No.5 |
| 7 | Two-spotted spider mite | 720 | Show white or yellow spots, blade back netting. | See Figure 1 third row No.6-No.10 |
| 8 | Virus disease | 481 | Develop yellow or green, slightly shrinking. | See Figure 1 forth row No.1-No.5 |
| 9 | Yellow leaf curl disease | 814 | Develop small and curl upward, crumpling, and marginal yellowing, bushy appearance. | See Figure 1 forth row No.6-No.10 |
| 3 | Health | 643 | | See Figure 1 fifth row |
| Total | | 5550 | | |

the image from left to right; (2) flipping the image from top to bottom; (3) flipping the image diagonally; (4) adjusting the brightness of image, setting the max delta to 0.4; (5) adjusting the contrast of image, setting the ratio from 0.2 to 1.5; (6) adjusting the hue of image, setting the max delta to 0.5; (7) adjusting the saturation of image, setting the ratio from 0.2 to 1.5; (8) rotating the image by 90° and 270°, respectively. The final dataset is shown in Table 2, and the label in the first row represents the disease categories which are given in Table 1.

### 3.3. Deep Learning Models

*3.3.1. AlexNet.* AlexNet is the winner of ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) 2012, a deep convolutional neural network, which has 60 million parameters and 650,000 neurons [21]. The architecture of AlexNet utilized in this paper is displayed in Figure 3. The AlexNet architecture consists of five convolutional layers (i.e., conv1, conv2, and so on), some of which are followed by max-pooling layers (i.e., pool1, pool2, and pool5), three fully connected layers (i.e., fc6, fc7, and fc8), and a liner layer with softmax activation in output. In order to reduce overfitting in the fully connected layers, a regularization method called "dropout" is used (i.e., drop6, drop7) [21]. The ReLU activation function is applied to each of the first seven layers (i.e., relu1, relu2, and so on) [45]. In Figure 3, the notation $m \times m \times n$ in each convolutional layer represents the size of the feature map for each layer, 4096 represents the number of neurons of the first two fully connected layers. The number of neurons of the final fully connected layer was modified to 9, since the classification problem in this work has 9 categories. In addition, the size of input images must be shaped to $227 \times 227$, which meets the input pixel size requirement of AlexNet.

*3.3.2. GoogLeNet.* GoogLeNet is an inception architecture [22], which is the winner of ILSVRC 2014 and owns roughly 6.8 million parameters. The architecture of GoogLeNet is presented in Figure 4. The inception module is inspired by the network in network [46] and uses a parallel combination of $1 \times 1$, $3 \times 3$, and $5 \times 5$ convolutional layer along with $3 \times 3$ max-pooling layer [45]; the $1 \times 1$ convolutional layer before $3 \times 3$ and $5 \times 5$ convolutional layer reduces the spatial dimension and limits the size of GoogLeNet. The whole architecture of GoogLeNet is stacked by inception module on top of each other (See Figure 4), which has nine inception modules, two convolutional layers, four max-pooling layers, one average pooling layer, one fully connected layer, and a linear layer with softmax function in the output. GoogLeNet uses dropout regularization in the fully connected layer and applies the ReLU activation function in all of the convolutional layers [29]. In this work, the last three layers of GoogLeNet were replaced by a fully connected layer, a softmax layer, and a classification layer; the fully connected layer was modified to 9 neurons, which is equal to the

Table 2: The final tomato leaf dataset.

| Labels | Label1 | Label2 | Label3 | Label4 | Label5 | Label6 | Label7 | Label8 | Label9 | Total |
|---|---|---|---|---|---|---|---|---|---|---|
| Training set | 3933 | 2916 | 4626 | 5229 | 3456 | 5283 | 5184 | 3465 | 5859 | 39951 |
| Testing set | 110 | 81 | 129 | 145 | 96 | 147 | 144 | 161 | 163 | 1176 |

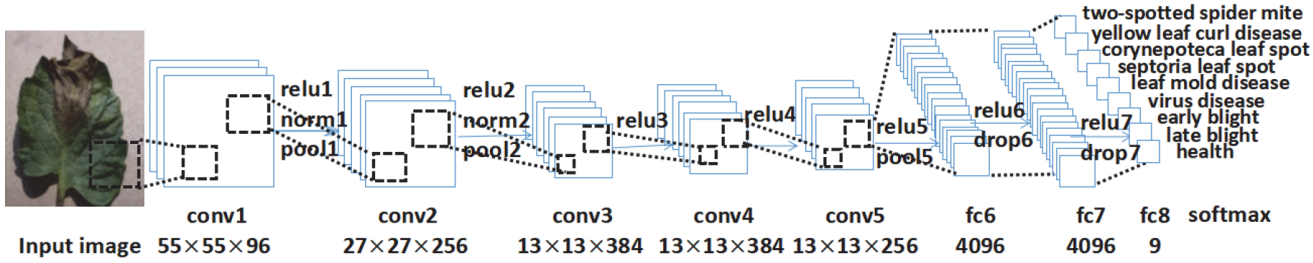



Figure 3: The architecture of AlexNet in this work.

categories in the tomato leaf disease identification problem. The size requested of input image of GoogLeNet is $224 \times 224$.

*3.3.3. ResNet.* The deep residual learning framework is proposed for addressing the degradation problem. ResNet consists of many stacked residual units, which won the first place in ILSVRC 2015 and COCO 2015 classification challenge with error rate of 3.57% [26]. Each unit can be expressed in the following formulas [47]:

$$y_l = h(x_l) + F(x_l, W_l) \tag{4}$$

$$x_{l+1} = f(y_l) \tag{5}$$

where $x_l$ and $x_{l+1}$ are input and output of the l-th unit, and $F$ is a residual function. In [26]$h(x_l) = x_l$ is an identity mapping and $f$ is a ReLU function [42]. A "bottleneck" building block is designed for ResNet (See Figure 5) and comprises two $1 \times 1$ convolutions with a $3 \times 3$ convolution in between and a direct skip connection bypassing input and output. The $1 \times 1$ layers are responsible for changing in dimensions. ResNet model has three types of layers with 50, 101, and 152. For saving computing resources and training time, we choose the ResNet50, which also has high performance. In this work, at first, the last three layers of ResNet were modified by a fully connected layer, a softmax layer, and a classification layer, the fully connected layer was replaced to 9 neurons, which is equal to the categories of the tomato leaf disease. We changed the structure of ResNet subsequently. The size of input image of ResNet should satisfy $224 \times 224$.

## 4. Experiments and Results

In this section, we reveal the experiments and discuss the experimental results. All the experiments were implemented in Matlab under Windows 10, using the GPU NVIDIA GTX1050 with 4G video memory or NVIDIA GTX1080Ti with 11G video memory. In this paper, overall accuracy was regarded as the evaluation metric in every experiment on tomato leaf disease detection, which means the percentage of samples that are correctly classified:

$$\text{accuracy} = \frac{\text{true positive} + \text{true negative}}{\text{positive} + \text{negative}} \tag{6}$$

where "true positive" is the number of instances that are positive and classified as positive, "true negative" is the number of instances that are negative and classified as negative, and the denominator represents the total number of samples. In addition, the training time was regarded as an additional performance metric of the network structure experiment.

*4.1. Experiments on Optimization Methods.* The first experiment is designed for seeking the suitable optimization method between SGD [30] and Adam [30, 31] in identifying tomato leaf diseases, combining with the pretrained network AlexNet, GoogLeNet, and ResNet, respectively. In this experiment, the hyperparameters were set as follows for each network: the batch size was set to 32, the initial learning rate was set to 0.001 and dropped by a factor of 0.5 every 2 epochs, and the max epoch was set to 5; i.e., the number of iterations is 6240. So far as SGD optimization method, the momentum was set to 0.9. For Adam, the gradient decay rate $\beta_1$ was set to 0.9, the squared gradient decay rate $\beta_2$ was set to 0.999, and the denominator offset $\varepsilon$ was set to $10^{-8}$ [31]. The accuracy of different networks is displayed in Table 3. In addition, we choose the better results in each deep neural network to show the training loss against number of iterations during the fine-tuning process (See Figure 6). The words inside parenthesis indicate the corresponding optimization method.

In Table 3, the ResNet with SGD optimization method gets the highest test accuracy 96.51%. In identifying tomato leaf diseases, the performance of Adam optimization method is inferior to the SGD optimization method, especially in combining with AlexNet. In the following paper, AlexNet (SGD), GoogLeNet (SGD), and ResNet (SGD) are referred to as AlexNet, GoogLeNet, and ResNet, respectively.

As it can be seen in Figure 6, the training loss of ResNet drops rapidly in the earlier iterations and tends to stable after 3000 iterations. Consistent with Table 3, the performance of

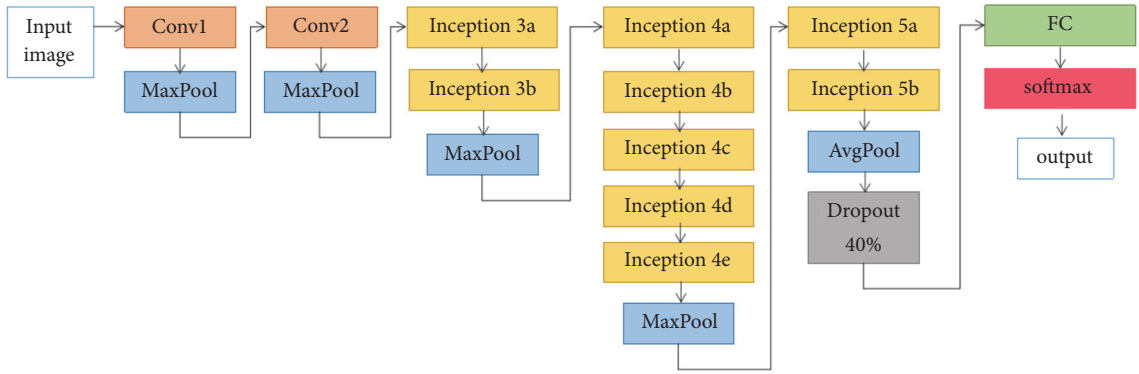



Figure 4: The architecture of GoogLeNet [22, 45].

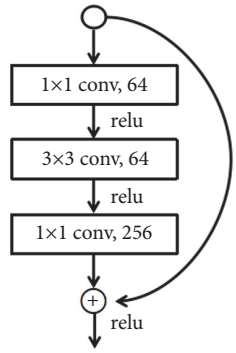



Figure 5: ResNet bottleneck residual building block [26].

Table 3: Model recognition accuracy.

| Model | Accuracy |
|---|---|
| AlexNet (SGD) | 95.83% |
| AlexNet (Adam) | 13.86% |
| GoogLeNet (SGD) | 95.66% |
| GoogLeNet (Adam) | 94.06% |
| **ResNet (SGD)** | **96.51%** |
| ResNet (Adam) | 94.39% |

AlexNet and GoogLeNet is similar and both inferior to the ResNet.

*4.2. Experiments on Batch Size and Number of Iterations.* From the experiment on optimization methods, the ResNet obtains the highest classification accuracy. Next, we evaluated the effects of batch size and the number of iterations on the performance of the ResNet. The batch size was set to 16, 32, and 64, respectively. Meanwhile, the number of iterations was set to 2496, 4992, and 9984. The classification accuracy of different training scenarios is given in Table 4. At the same time, the classification accuracy of each label's representative leaf disease category (See Table 1) is given. In this experiment, the initial learning rate was set to 0.001 and dropped by a factor of 0.5 every 2496 iterations.

In Table 4, the best overall classification accuracy 97.19% is got by the ResNet combining with batch size 16 and

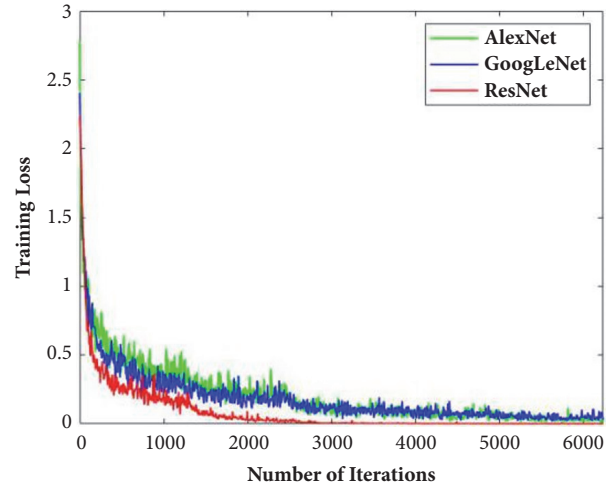



Figure 6: The training loss during the fine-tuning process.

iterations 4992. As shown in Table 4, whether increasing the number of iterations or batch size, the performance of corresponding models has not been improved significantly in identifying tomato leaf disease. A small batch size with a medium number of iterations is quite effective in this work. Moreover, a larger batch size and number of iterations increases the training duration. We have not tried higher or lower values for the attempted parameters, since different classification task may have various suitable parameters, and it is hard to give a certain rule in setting hyperparameters.

*4.3. Experiments on Full Training and Fine-Tuning of ResNet.* This section is designed for exploring the performance of CNN by changing the structure of the models. In practical, a deep CNN always owns a large size which means a large number of parameters. Thus, full training of a deep CNN requires extensive computational resources and is time-consuming. In addition, full training of a deep CNN may led to overfitting when the training data is limited. So we compared the performance of the pretrained CNN through full training and fine-tuning their structures.

We changed the structure of ResNet, and combination of the best parameters from the front experiments was utilized.

Table 4: Classification accuracies with different parameters during fine-tuning of the ResNet. The numbers inside parenthesis indicate batch size and number of iterations.

| Networks | label1 | label2 | label3 | label4 | label5 | label6 | label7 | label8 | label9 | overall |
|---|---|---|---|---|---|---|---|---|---|---|
| ResNet (16,2496) | 90.91% | 88.89% | 100% | 100% | 96.88% | 100% | 90.28% | 88.20% | 97.55% | 94.98% |
| **ResNet (16,4992)** | **98.18%** | **98.77%** | **100%** | **98.62%** | **96.88%** | **100%** | **96.53%** | **88.82%** | **98.77%** | **97.19%** |
| ResNet (16,9984) | 98.18% | 97.53% | 100% | 98.62% | 96.88% | 100% | 97.22% | 86.96% | 98.77% | 96.94% |
| ResNet (32,2496) | 97.27% | 95.06% | 100% | 97.93% | 96.88% | 100% | 95.14% | 86.34% | 99.39% | 96.34% |
| ResNet (32,4992) | 97.27% | 95.06% | 100% | 97.24% | 96.88% | 100% | 96.53% | 86.96% | 99.39% | 96.51% |
| ResNet (32,9984) | 96.36% | 96.30% | 100% | 99.31% | 96.88% | 100% | 94.44% | 88.20% | 98.77% | 96.60% |
| ResNet (64,2496) | 93.64% | 93.83% | 100% | 97.24% | 96.88% | 100% | 94.44% | 87.58% | 99.39% | 95.92% |
| ResNet (64,4992) | 94.55% | 95.29% | 100% | 96.55% | 95.83% | 100% | 95.83% | 86.96% | 99.39% | 95.83% |
| ResNet (64,9984) | 95.45% | 93.83% | 100% | 97.93% | 96.88% | 100% | 94.44% | 87.58% | 99.39% | 96.17% |

ResNet50 has 177 layers if the layers for each building block and connection are calculated. In this experiment, the last three layers of ResNet were modified to a fully connected layer (denoted as "fc"), a softmax layer, and a classification layer, and the fully connected layer owns 9 neurons. The structure was changed by freezing the weights of a certain number of layers in the network by setting the learning rate in those layers to zero. During training, the parameters of the frozen layers are not updated. Full training and fine-tuning are defined by the number of training layers, i.e., full training (1-"fc"), fine-tuning (37-"fc", 79-"fc", 111-"fc", 141-"fc", 163-"fc"). The accuracy and training time of different network structure are presented in Table 5. At first, the batch size and 4992 iterations were combined, the initial learning rate was set to 0.001 and dropped by a factor of 0.1 every 2496 iterations. In order to get more convincing conclusions, ResNet (16, 9984), which gets the second place in Table 4, was also used to execute the experiments.

In Table 5, the accuracy and training time of different network structures are presented. In two cases, i.e., the 4992 iterations and 9984 iterations of ResNet, the accuracy of the model from the 37 layer fine-tuning structure are higher than that of the full training model. In the case where the number of iterations is 4992, the accuracy of the model from the 79 layer fine-tuning structure is equal to that of the full training model. The final column of the Table 5 represents the training time of the corresponding network, and it is clear that the training time of the fine-tuning models is greatly lowered than the full training model. Because the gradients of the frozen layers do not need to be computed, freezing the weights of initial layers can speed up network training. We observe that the moderate fine-tuning models (37-"fc", 79-"fc", 111-"fc") always led to a performance superior or approximately equal to the full training models. Thus, we suggest that, for practical application, the moderate fine-tuning models may be a good choice. Especially for the researcher who holds massive data, the fine-tuning models may achieve good performance while saving computational resources and time.

Moreover, the features of the final fully connected layer of ResNet (16, 4992, 37-"fc") were examined by utilizing the t-distributed Stochastic Neighbour Embedding (t-SNE) algorithm (see Figure 7) [48]. 1176 test images were used to extract the features. In Figure 7, different colors represent different labels; the corresponding disease categories of the labels were listed in Table 1. As shown in Figure 7, 9 different color points are clearly separated, which indicates that the features learned from the ResNet with the optimal structure can be used to classify the tomato leaf disease precisely.

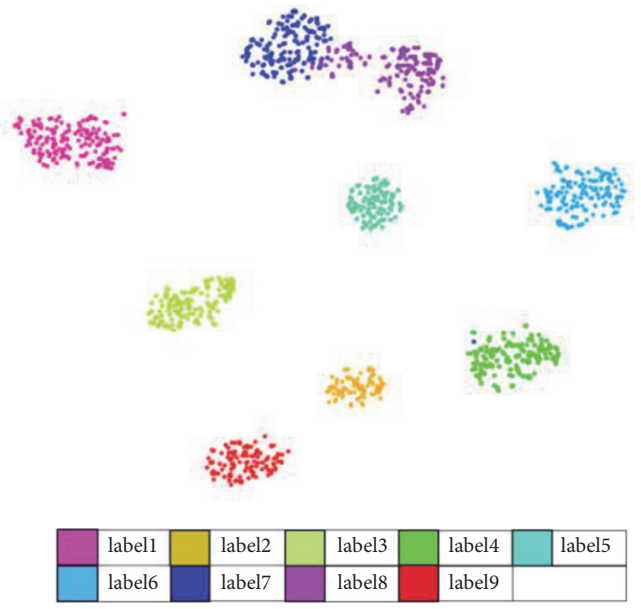



Figure 7: Two-dimensional scatter plot of high-dimensional features generated with t-SNE.

## 5. Conclusion

This paper concentrates on identifying tomato leaf disease using deep convolutional neural networks by transfer learning. The utilized networks are based on the pretrained deep learning models of AlexNet, GoogLeNet, and ResNet. First we compared the relative performance of these networks by using SGD and Adam optimization method, revealing that the ResNet with SGD optimization method obtains the highest result with the best accuracy, 96.51%. Then, the performance evaluation of batch size and number of

Table 5: Accuracies and training time in different network structures. The values inside parenthesis denote batch size, number of iterations, and training layers.

| Network topology | Accuracy | Time (min:sec) |
|---|---|---|
| ResNet (16, 4992, 1-"fc") | 96.43% | 59min 30sec |
| **ResNet (16, 4992, 37-"fc")** | **97.28%** | **44min 13sec** |
| ResNet (16, 4992, 79-"fc") | 96.43% | 37min 27sec |
| ResNet (16, 4992, 111-"fc") | 95.75% | 30min 6sec |
| ResNet (16, 4992, 141-"fc") | 95.32% | 24min 15sec |
| ResNet (16, 4992, 163-"fc") | 92.69% | 19min 31sec |
| ResNet (16, 9984, 1-"fc") | 96.94% | 118min 32sec |
| **ResNet (16, 9984, 37-"fc")** | **97.02%** | **92min 53sec** |
| ResNet (16, 9984, 79-"fc") | 96.77% | 72min 23sec |
| ResNet (16, 9984, 111-"fc") | 96.26% | 58min 40sec |
| ResNet (16, 9984, 141-"fc") | 95.75% | 47min 22sec |
| ResNet (16, 9984, 163-"fc") | 93.96% | 39min 32sec |

iterations affecting the transfer learning of the ResNet was conducted. A small batch size of 16 combining a moderate number of iterations of 4992 is the optimal choice in this work. Our findings suggest that, for a particular task, neither large batch size nor large number of iterations may improve the accuracy of the target model. The setting of batch size and number of iterations depends on your data set and the utilized network. Next, the best combined model was used to fine-tune the structure. Fine-tuning ResNet layers from 37 to "fc" obtained the highest accuracy 97.28% in identifying tomato leaf disease. Based on the amount of available data, layer-wise fine-tuning may provide a practical way to achieve the best performance of the application at hand. We believe that the results obtained in this work will bring some inspiration to other similar visual recognition problems, and the practical study of this work can be easily extended to other plant leaf disease identification problems.

## Data Availability

The tomato leaf data supporting this work are from previously reported studies, which have been cited. The processed data are available from the corresponding author request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This study was supported by the National Science and technology support program (2014BAD12B01-1-3), Public Welfare Industry (Agriculture) Research Projects Level-2 (201503116-04-06), Postdoctoral Foundation of Heilongjiang Province (LBHZ15020), Harbin Applied Technology Research and Development Program (2017RAQXJ096), and Economic Decision Making and Early Warning of Soybean Industry in Technology Collaborative Innovation System of Soybean Industry in Heilongjiang Province (20170401).

## References

[1] R. Chaerani and R. E. Voorrips, "Tomato early blight (Alternaria solani): The pathogen, genetics, and breeding for resistance," *Journal of General Plant Pathology*, vol. 72, no. 6, pp. 335–347, 2006.

[2] A. M. Dickey, L. S. Osborne, and C. L. Mckenzie, "Papaya (Carica papaya, Brassicales: Caricaceae) is not a host plant of tomato yellow leaf curl virus (TYLCV; family Geminiviridae, genus Begomovirus)," *Florida Entomologist*, vol. 95, no. 1, pp. 211–213, 2012.

[3] G. Wei, L. Baoju, S. Yanxia, and X. Xuewen, "Studies on pathogenicity differentiation of corynespora cassiicola isolates, against cucumber, tomato and eggplant," *Acta Horticulturae Sinica*, vol. 38, no. 3, pp. 465–470, 2011.

[4] P. Lindhout, W. Korta, M. Cislik, I. Vos, and T. Gerlagh, "Further identification of races of Cladosporium fulvum (Fulvia fulva) on tomato originating from the Netherlands France and Poland," *Netherlands Journal of Plant Pathology*, vol. 95, no. 3, pp. 143–148, 1989.

[5] K. Kubota, S. Tsuda, A. Tamai, and T. Meshi, "Tomato mosaic virus replication protein suppresses virus-targeted posttranscriptional gene silencing," *Journal of Virology*, vol. 77, no. 20, pp. 11016–11026, 2003.

[6] M. Tian, B. Benedetti, and S. Kamoun, "A second Kazal-like protease inhibitor from Phytophthora infestans inhibits and interacts with the apoplastic pathogenesis-related protease P69B of tomato," *Plant Physiology*, vol. 138, no. 3, pp. 1785–1793, 2005.

[7] L. E. Blum, "Reduction of incidence and severity of Septoria lycopersici leaf spot of tomato with bacteria and yeasts," *Ciência Rural*, vol. 30, no. 5, pp. 761–765, 2000.

[8] E. A. Chatzivasileiadis and M. W. Sabelis, "Toxicity of methyl ketones from tomato trichomes to Tetranychus urticae Koch," *Experimental and Applied Acarology*, vol. 21, no. 6-7, pp. 473–484, 1997.

[9] M. Anthimopoulos, S. Christodoulidis, L. Ebner, A. Christe, and S. Mougiakakou, "Lung Pattern Classification for Interstitial Lung Diseases Using a Deep Convolutional Neural Network," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1207–1216, 2016.

[10] Y. Tang and X. Wu, "Text-independent writer identification via CNN features and joint Bayesian," in *Proceedings of the 15th*

*International Conference on Frontiers in Handwriting Recognition, ICFHR 2016*, pp. 566–571, Shenzhen, China, October 2016.

[11] Y. Tang and X. Wu, "Saliency Detection via Combining Region-Level and Pixel-Level Predictions with CNNs," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 1608–05186, 2016.

[12] Y. Tang and X. Wu, "Salient object detection with chained multi-scale fully convolutional network," *ACM Multimedia (ACMMM)*, pp. 618–626, 2017.

[13] Y. Tang and X. Wu, "Scene text detection and segmentation based on cascaded convolution neural networks," *IEEE Transactions on Image Processing*, vol. 26, no. 3, pp. 1509–1520, 2017.

[14] Y. Tang and X. Wu, "Scene Text Detection using Superpixel based Stroke Feature Transform and Deep Learning based Region Classification," *IEEE Transactions on Multimedia*, vol. 20, no. 9, pp. 2276–2288, 2018.

[15] Y. Yao, X. Wu, Z. Lei, S. Shan, and W. Zuo, "Joint Representation and Truncated Inference Learning for Correlation Filter based Tracking," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 1–14, 2018.

[16] L. Zhang, F. Yang, Y. Daniel Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *Proceedings of the 23rd IEEE International Conference on Image Processing, ICIP 2016*, pp. 3708–3712, Phoenix, AZ, USA, September 2016.

[17] D. Xie, L. Zhang, and L. Bai, "Deep learning in visual computing and signal processing," *Applied Computational Intelligence and Soft Computing*, vol. 2017, Article ID 1320780, 13 pages, 2017.

[18] Z. Zhou, J. Shin, L. Zhang, S. Gurudu, M. Gotway, and J. Liang, "Fine-tuning convolutional neural networks for biomedical image analysis: Actively and incrementally," in *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, pp. 4761–4772, USA, July 2017.

[19] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of the 15th IEEE International Conference on Computer Vision, ICCV 2015*, pp. 3730–3738, Santiago, Chile, 2015.

[20] W. Ouyang and X. Wang, "Joint deep learning for pedestrian detection," in *Proceedings of the 14th IEEE International Conference on Computer Vision (ICCV '13)*, pp. 2056–2063, Sydney, Australia, December 2013.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS '12)*, pp. 1097–1105, Lake Tahoe, Nev, USA, December 2012.

[22] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '15)*, pp. 1–9, IEEE, Boston, Mass, USA, June 2015.

[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," https://arxiv.org/abs/1409.1556, 2015.

[24] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," https://arxiv.org/abs/1512.00567, 2015.

[25] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," https://arxiv.org/abs/1602.07261, 2016.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 770–778, July 2016.

[27] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten, "Densely connected convolutional networks," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2016.

[28] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[29] M. Mehdipour Ghazi, B. Yanikoglu, and E. Aptoula, "Plant identification using deep neural networks via optimization of transfer learning parameters," *Neurocomputing*, vol. 235, pp. 228–235, 2017.

[30] S. Ruder, "An overview of gradient descent optimization algorithms," https://arxiv.org/abs/1609.04747, 2017.

[31] D. P. Kingma and J. L. Ba, "Adam: a method for stochastic optimization," https://arxiv.org/abs/1412.6980, 2017.

[32] S. S. Sannakki, V. S. Rajpurohit, V. B. Nargund, R. Arun Kumar, and S. Prema Yallur, "Leaf disease grading by machine vision and fuzzy logic," *International Journal of Computer Technology and Applications*, vol. 2, no. 5, pp. 1709–1716, 2011.

[33] D. Samanta, P. P. Chaudhury, and A. Ghosh, "Scab diseases detection of potato using image processing," *International Journal of Computer Trends and Technology*, vol. 3, pp. 109–113, 2012.

[34] P. J. Herrera, J. Dorado, and Á. Ribeiro, "A novel approach for weed type classification based on shape descriptors and a fuzzy decision-making method," *Sensors*, vol. 14, no. 8, pp. 15304–15324, 2014.

[35] B. Cheng and E. T. Matson, "A feature-based machine learning agent for automatic rice and weed discrimination," *International Conference on Artificial Intelligence and Soft Computing*, pp. 517–527, 2015.

[36] S. Sankaran and R. Ehsani, "Comparison of visible-near infrared and mid-infrared spectroscopy for classification of Huanglongbing and citrus canker infected leaves," *Agricultural Engineering International: CIGR Journal*, vol. 15, no. 3, pp. 75–79, 2013.

[37] X. Cheng, Y. Zhang, Y. Chen, Y. Wu, and Y. Yue, "Pest identification via deep residual learning in complex background," *Computers and Electronics in Agriculture*, vol. 141, pp. 351–356, 2017.

[38] A. dos Santos Ferreira, D. Matte Freitas, G. Gonçalves da Silva, H. Pistori, and M. Theophilo Folhes, "Weed detection in soybean crops using ConvNets," *Computers and Electronics in Agriculture*, vol. 143, pp. 314–324, 2017.

[39] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification," *Computational Intelligence and Neuroscience*, vol. 2016, Article ID 3289801, 11 pages, 2016.

[40] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in Plant Science*, vol. 7, article no. 1419, 2016.

[41] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "Deepfruits: A fruit detection system using deep neural networks," *Sensors*, vol. 16, article no. 1222, no. 8, 2016.

[42] V. Nair and G. E. Hinton, "Rectified linear units improve Restricted Boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML '10)*, pp. 807–814, Haifa, Israel, June 2010.

[43] D. P. Hughes and M. Salathe, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," https://arxiv.org/abs/1511.08060, 2016.

[44] D. Ciresan, U. Meier, J. Masci, and J. Schmidhuber, "A committee of neural networks for traffic sign classification," in *Proceedings of the 2011 International Joint Conference on Neural Networks (IJCNN 2011 - San Jose)*, San Jose, CA, USA, July 2011.

[45] P. Pawara, E. Okafor, O. Surinta, L. Schomaker, and M. Wiering, "Comparing Local Descriptors and Bags of Visual Words to Deep Convolutional Neural Networks for Plant Recognition," in *Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods*, pp. 479–486, Porto, Portugal, Feburary 2017.

[46] M. Lin, "Network in Nnetwork," https://arxiv.org/abs/1312.4400, 2014.

[47] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proceedings of the European Conference on Computer Vision*, pp. 630–645, 2016.

[48] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2625, 2008.

Hindawi
Advances in Multimedia
Volume 2018, Article ID 4976372, 8 pages
https://doi.org/10.1155/2018/4976372

*Research Article*

# Performance Evaluation of Contour Based Segmentation Methods for Ultrasound Images

**R. J. Hemalatha [1], V. Vijaybaskar,[2] and T. R. Thamizhvani [3]**

[1]*Department of Biomedical Engineering, Sathyabama Institute of Science and Technology, Chennai, TamilNadu-600, India*
[2]*Department of Electronica and Telecommunication, Sathyabama Institute of Science and Technology, Chennai, TamilNadu-600, India*
[3]*Department of Biomedical Engineering, Vels Institute of Science, Technology and Advanced Studies, Chennai, TamilNadu-600117, India*

Correspondence should be addressed to R. J. Hemalatha; rjhemalatha@gmail.com

Received 26 May 2018; Revised 9 August 2018; Accepted 30 August 2018; Published 16 September 2018

Academic Editor: Huiyu Zhou



Active contour methods are widely used for medical image segmentation. Using level set algorithms the applications of active contour methods have become flexible and convenient. This paper describes the evaluation of the performance of the active contour models using performance metrics and statistical analysis. We have implemented five different methods for segmenting the synovial region in arthritis affected ultrasound image. A comparative analysis between the methods of segmentation was performed and the best segmentation method was identified using similarity criteria, standard error, and F-test. For further analysis, classification of the segmentation techniques using support vector machine (SVM) classifier is performed to determine the absolute method for synovial region detection. With these results, localized region based active contour named Lankton method is defined to be the best segmentation method.

## 1. Introduction

Musculoskeletal disorder (MSD), an epidemic disease, is now a significant health problem in emerging and most developing countries in the world. MSD remains the most prevalent disease in society due to its impact on mobility, ability to work, and life style. Arthritis is one of the prevalent MSD among all the age groups of people [1]. The most affected portion is the joint region. At relatively early stage of the disease the synovial membrane around the joints used to inflame and leads to degeneration of the joints. The main system to visualize the joint state is through ultra sound diagnosis (USD). The USD is less expensive and available with all the clinicians [2, 3]. The USD images represent different tissue regions with variations of gray shades. These images are further processed to segment the synovial region and to analyze the disease condition and further progression of the disease.

One of the essential tasks in image analysis is image segmentation. Segmentation is to extract objects from the images by dividing the image into set of regions with different properties. Segmentation plays an important role in automatic object recognition or pattern identification process to identify pathologies and medical diagnosis [4, 5]. The most challenging task is to extract the contour and boundaries of the desired region for dynamic analysis of anatomical structures. One of the most robust segmentation methods for medical images is active contour method (ACM).The method implies curve evolution to detect the region of interest in a given image [6–8]. The segmentation process is based on edge and region based approach. The geometric active contour model proposed by Caselles and Malladi et al. is an edge based approach which is based on evolution of curves and geometric flows [9]. Chan and Vese have proposed edge less active contour model which is one of the most well-known region based method. Bernard et al. have proposed a parameterized active contour method. More recently Li et al. and Lankton proposed a method which utilizes local region information for segmentation [9–14].

In this paper we have applied different segmentation methods to arthritis affected finger joint ultrasound images to segment the synovial region. The efficiency of these

methods is analyzed using performance analysis metrics and statistical analysis method. For performance analysis metrics we have used Dice coefficient and Hausdorff coefficient and, for statistical analysis method standard error, F-test were used. At the end classification is used to define the absolute active contour method for segmentation of synovial region by training and analyzing the performance metrics and statistical values. The rest of the paper is organized as follows Section 2. The Methods and Materials, Section 3 the results and discussion, and Section 4 the conclusion.

## 2. Methods and Materials

In medical imaging, the segmentation of regions with specific parameters is carried out with the help of active contour models. Because these models develop a contour around the target object and segregate it from the image, the segmented image possesses only the required information of the target object [15]. The level set segmentation methods like Caselles, Chan–Vese, Bernard, Li, and Lankton are applied on arthritis affected finger joint images obtained from the MEDUSA database http://medusa.aei.polsl.pl.[16–18]. Further using performance analysis metrics like dice coefficient and Hausdroff distance and statistical analysis metrics like standard error and F-test describes the significant difference between the techniques used for segmentation. Classification using SVM defines the best suited method for synovial region segmentation.

MEDUSA is a standardized and authorized database which consists of finger joint images of different grades (grade 0, grade 1, grade 2, and grade 3) of synovitis. Various studies related to arthritis and synovitis are performed using this database.

*2.1. Caselles.* Caselles is geodesic based active contour methods which largely depend on the level set functions that describe the specific regions in the image for segmentation. Contours are described based on the geometric flow of curve and detection of objects in the image [11]. This type of contour model modifies the curve in the plane by moving the points of the curve perpendicular. The motion of the points is at a speed proportional to the curvature of the region in the image. By adding an area of minimizing region (balloon force), propagation of contour occurs internally by minimization of the interior energy given by

$$\mathrm{E}(\mathrm{C}) = \int g(I(c(P)))\left\|C'(P)\right\| dP \tag{1}$$

$$g(I) = \frac{1}{1 + \|\nabla (G * I)\|^2} \tag{2}$$

I is image intensity, G is Gaussian Filter of unity variance, and C is derived parametric curve to regions with high gradient where set level function is executed as a signed distance function (P= ∅(x) )

Contour models use the energy forces for geometric flow curve description. Geometric contours can be obtained based on regions and edges in the curvature of the image [12].

*2.2. Chan–Vese.* Chan–Vese is a region based method which segments an image into two homogeneous regions. The method utilizes energy minimization technique defined by weighted values corresponding to the average value of sum of intensity difference from outside and inside the segmented region [9, 10]. Contours are based on either the variance inside and outside contour or the squared difference between average intensities inside and outside the contours along with the total contour length. This contour model helps to determine different image properties, not only edges, and it also includes regions based on texture and other geometrical features. Energy defines the entire region of interest from the image.

The total energy of the model is given in

$$\begin{aligned} \mathrm{E}_{\emptyset} = {} & \mu \int_{\Omega} \delta(\varphi(\mathrm{a}))\left|\nabla\varphi(\mathrm{a})\right| \mathrm{da} + \mathrm{v}\int_{\Omega} \mathrm{H}(\varphi(a))\,\mathrm{da} \\ & + \lambda_1 \int_{\Omega} \mathrm{H}(\varphi(\mathrm{a}))\left|\mathrm{f} - \mathrm{C}_1\right| \\ & + \lambda_2 \int_{\Omega} (1 - \mathrm{H}(\varphi(\mathrm{a})))\left|\mathrm{f} - \mathrm{C}_2\right| \mathrm{da} \end{aligned} \tag{3}$$

$\mu$, $\nu$, $\lambda_1$, $\lambda_2$: real parameters

$\mathrm{C}_1$, $\mathrm{C}_2$: constants determined for segmentation

$-1 \leq \varphi(\mathrm{a}) \leq 1$: level set function in which $\varphi(\mathrm{a})=0$ specifies the interface

f: original image

H: heavy side function in 1 dimension centered at 0 and $\delta = \mathrm{H}'$.

*2.3. Bernard.* Bernard method utilizes B-spline coefficients as energy minimization function. These utilize parameterized active contour method [12]. Spline coefficients define the contour models for the pixels of interest. The energy based functions inside and outside is described with these coefficients. Contour models describe the entire structures with inflation force that can overpower forces from weak edges, amplifying the issue with localization of initial guess. To speed up the process a linear combination of B-spline basis functions is used and given in

$$\mathrm{E}_{\emptyset} = \int_{\Omega} \mathrm{F}(\mathrm{I}(\mathrm{a}), \emptyset(\mathrm{a}))\,\mathrm{dx} \tag{4}$$

$$\begin{aligned} \mathrm{F}(\mathrm{I}(\mathrm{a}), \emptyset(\mathrm{a})) = {} & (\mathrm{I}(\mathrm{a}) - \upsilon)^2 \mathrm{H}(\emptyset(\mathrm{a})) \\ & + (\mathrm{I}(a) - \mu)^2 (\mathrm{I} - \mathrm{H}(\emptyset(\mathrm{a}))). \end{aligned} \tag{5}$$

Φ(a) is linear combination of B-spline basis functions.

*2.4. Chumming Li.* In order to separate the region into two homogenous regions this method utilizes using local neighbourhood statistics for each pixel given in (6). It uses local region information for segmentation [13]. The energy function of the region based active contour model is range of region based domain kernel function. By minimizing the

energy function, the region of elements of the target could be determined in images with contours.

$$\begin{aligned} E_{\emptyset} &= \lambda_1 \iint K_{\sigma}(a-b)\left|I(b)-f_1(a)\right|^2 H(\emptyset(a))\,db\,da \\ &+ \lambda_2 \iint K_{\sigma}(a-b)\left|I(b)-f_2(a)\right|^2 \\ &\cdot (1-H(\emptyset(a)))\,db\,da + \upsilon \int \delta(\emptyset(a)) \\ &\cdot \|\nabla\emptyset(a)\|\,da + \mu \int \frac{1}{2}\left(\|\nabla\emptyset(a)\|-1\right)^2 da \end{aligned} \tag{6}$$

I(a): pixel intensity at x

H: heavy side function

$K_{\sigma}$: Gaussian Kernel.

$$K_{\sigma}(\mu) = \frac{1}{(2\pi)^{\pi/2}\,\sigma^{\pi}} e^{-\|\mu\|^2/2\sigma^2} \tag{7}$$

*2.5. Lankton.* Lankton is a region based active contour method which segments non homogeneous objects. This method utilizes localizing region based energy which segments the region based on local information. It is not suitable for unsupervised image segmentation as it requires appropriate curve initialization [14]. These models form contour boundaries with energy forces required for the particular region of interest. The energy inside and outside depends on the local region pixels of the image that describes the required region. The energy equation is illustrated in

$$\begin{aligned} E_{\emptyset} &= \int_{\Omega} \delta(\emptyset(a)) \int_{\Omega} F(I(a), \emptyset(b)), B(a,b)\,db\,da \\ &+ \lambda \int_{\Omega} \|\nabla\emptyset(a)\|\,\delta(\emptyset(a))\,da \end{aligned} \tag{8}$$

$\delta$ is Dirac function

B is Ball of radius r centered at point x

*2.6. Performance Evaluation Metrics.* The segmentation methods are qualitatively and quantitatively assessed and compared with each other based on three kinds of criteria. Based on this the best suited algorithm is chosen for particular applications.

**Visual criteria:** The segmented region using the active contour methods is compared with the annotated images by expert radiologist as reference image. The segmented region obtained from level set function methods is compared with reference image.

**Computation time:** The time taken for each algorithm to segment the region represents the speed of the algorithm. The speed of the respective algorithm is compared.

**Similarity criteria:** This criterion measures the similarity between the reference and segmented image. The quality of the segmented image is measured by calculating the Dice coefficient, Hausdorff distance, and PSNR

Dice coefficient compares the segmented region with the reference region from the annotated image and provides the dice coefficient values ranging between 0 and 1. If it is 1 the segmented region is more similar and it is different when it is 0 [19].

$$\text{Dice} = \frac{2(A \cap B)}{A + B} \tag{9}$$

Hausdroff distance is a metric to measure dissimilarity between two point sets. Distance transform is used to compute the HD in an image. This is used to control the progress of level set based algorithms and to evaluate the quality of the clusters [20].

$$D_1(A,B) = \max_{x \in A}\left(\min_{y \in B}\left(\|x-y\|\right)\right) \tag{10}$$

*2.7. Statistical Analysis.* The features like mean, variance, and standard errors are calculated for the segmentation methods. Among the five segmentation methods the best suited method was identified using statistical analysis.

**Mean:** The mean value is termed as average value which is computed by taking sum of all perceived outcomes divided by overall number of gray levels. The following shows mathematical expression for mean represented as $\overline{x}$:

$$\text{Mean, } \overline{x} = \frac{1}{n}\sum_{i=1}^{n} x \tag{11}$$

where n is sample size and x is observed value.

**Variance** is study of deviation of actual value versus predicted value. The deviation from actual and predicted indicates the performance of the methods used.

**Standard error** is defined as the measure of prediction's accuracy. Estimated standard error is related to sum of squared deviations of prediction (that is sum of squares error), described

$$\sigma_{est} = \sqrt{\frac{\sum\left(Y - Y'\right)^2}{N}} \tag{12}$$

$\sigma_{est}$ is the standard error of the estimate, Y is an actual range, $Y'$ is a predicted range, and N is the number of pairs of scores. $\sum(Y - Y')^2$ is the sum of squared differences between the actual scores and the predicted scores.

*2.8. Classification.* Classification is a process to describe the effective type or class based on the features derived from the region of interest. Support vector machines (SVM) are machine learning models. SVM is the representation of observations as points that maps to form separate divisions and a clear boundary factor defined as decision boundary. Multiclass support vector machine classifies the types based on the kernel models. Multiclass support vector machine is used to illustrate the appropriate type of active contour technique for the segmentation of the synovial region.

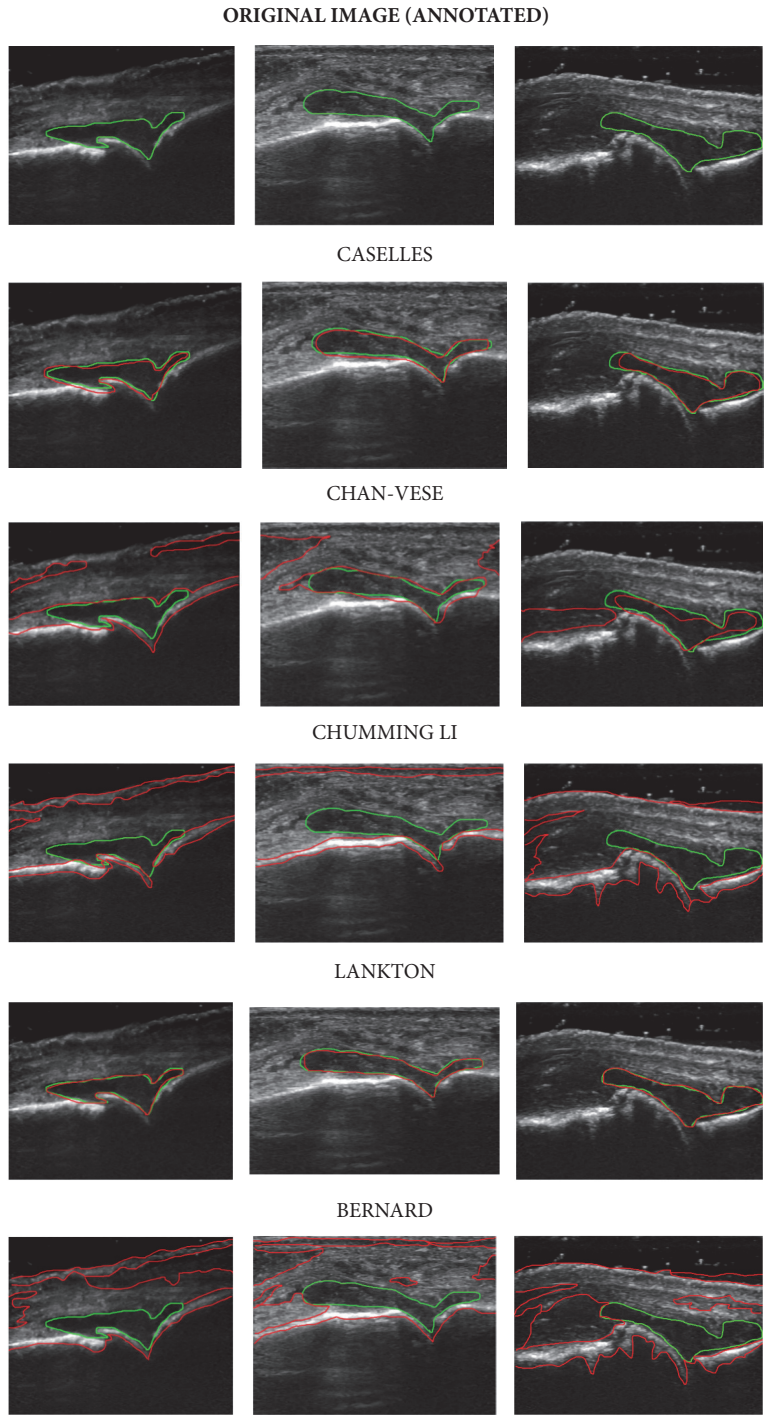



Figure 1: Different types of active contour segmentation techniques.

## 3. Results

In this method, different types of active contour segmentation techniques are used for the detection of synovial region. The segmentation methods were evaluated using performance metrics and statistical analysis. Ultrasound images from the database are used for the identification of synovial region. Fifty images of different grades are considered for segmentation of the synovial region from the database. Different active contour segmentation techniques are used to segment the synovial regions. Visual changes in the segmentation process are illustrated through the images in Figure 1.

In this figure, the annotated image is defined with green colour and the synovial region defined by the five different types of segmentation is displayed in red colour. The

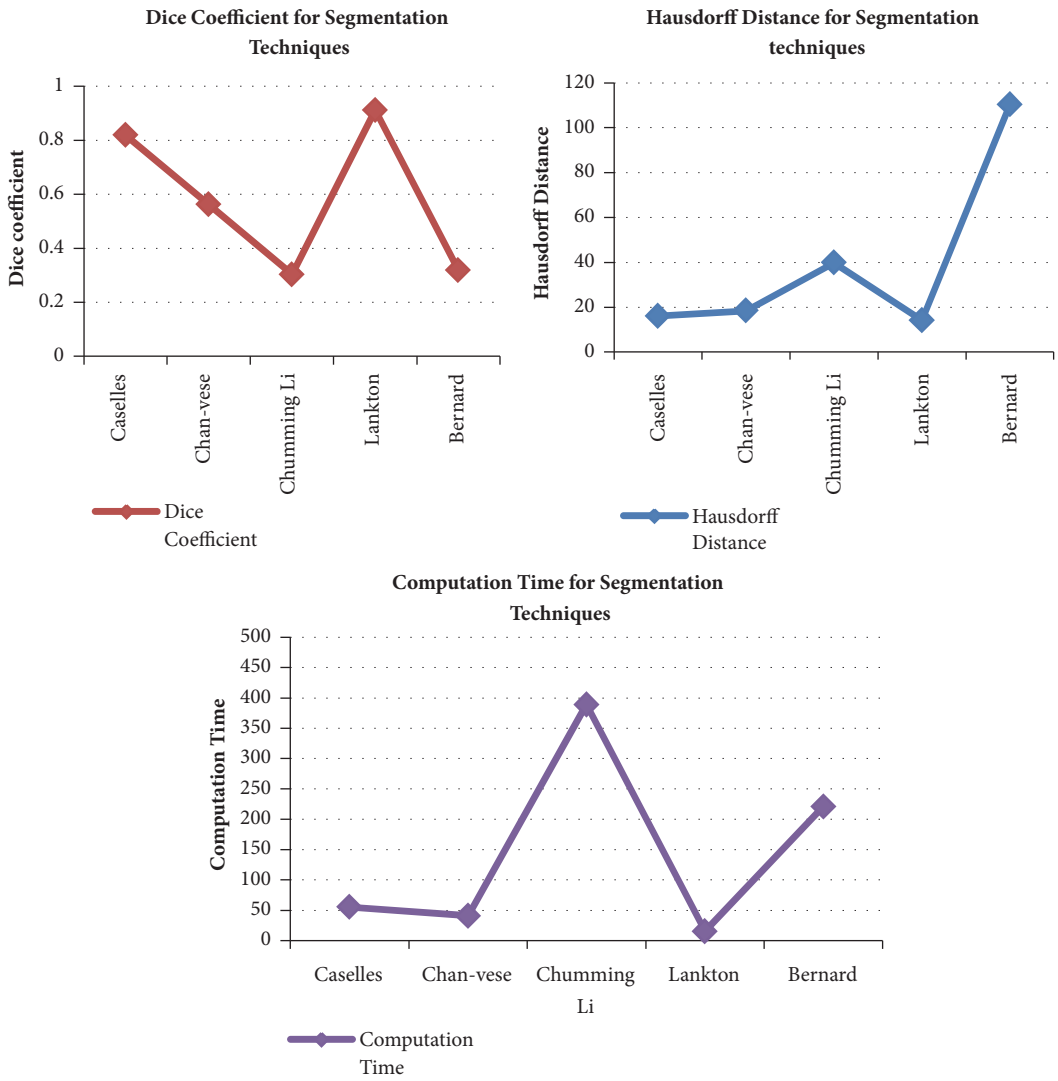



Figure 2: Graphical representation of the performance metrics of five different active contour techniques.

Table 1: Performance metrics value for several segmentation methods.

| Segmentation methods | Dice Coefficient | Hausdorff Distance | Computation Time |
|---|---|---|---|
| **Caselles** | 0.823 | 16.136 | 40.608 |
| **Chan-Vese** | 0.562 | 18.419 | 55.109 |
| **Chumming Li** | 0.303 | 39.841 | 388.815 |
| **Lankton** | 0.879 | 14.814 | 13.416 |
| **Bernard** | 0.204 | 88.601 | 402.057 |

segmented synovial region, when compared to the annotated images, visually defines the fact that Caselles and Lankton possess similarity. To analyze the absolute segmentation technique, further performance metrics description, statistical analysis, and classification are carried out. The performance analysis metrics like dice coefficient, Hausdorff distance, and computation time values of an image is tabulated in Table 1.

Comparison between each performance metric of the five different active contour techniques is graphically represented in Figure 2. These representations illustrate that the Caselles geodesic active contour and Lankton localized region based active contour methods have slightly similar values. From the dice coefficient values it is shown that Caselles and Lankton are more towards 1 that is the highest range. Hausdroff distance also shows the similarity between the two methods. Computation time in seconds defines the fact that the localized region based active contour Lankton is efficient. To further classify which is the best suitable segmentation technique we perform statistical analysis.

In statistical analysis, the features like standard error, average mean, and average variance values are derived from the similarity index parameters (Dice coefficient and Hausdroff distance) for the exact determination of the segmentation technique. In this Table 2 shows the statistical analysis of the similarity index parameters in performance metrics defining the average mean, standard error, and average variance values.

Table 2: Statistical values of performance metrics for segmentation techniques.

| Segmentation methods | Average Mean | | Standard Error | | Average Variance | |
|---|---|---|---|---|---|---|
| | Dice coefficient | Hausdorff distance | Dice coefficient | Hausdorff distance | Dice coefficient | Hausdorff distance |
| Caselles | 0.809±0.060 | 30±15.60 | 0.809±0.0837 | 30±11.18 | 0.809±0.0451 | 30±103.32 |
| Chan-vese | 0.579±0.150 | 90.9±25.40 | 0.579±0.1404 | 90.9±56.82 | 0.579±0.0233 | 90.9±46.10 |
| Chumming Li | 0.306±0.050 | 235.17±100 | 0.306±0.1829 | 235.17±6.570 | 0.306±0.0520 | 235.17±862 |
| Lankton | 0.873±0.005 | 18.7±0.010 | 0.873±0.0858 | 18.7±13.31 | 0.873±0.0403 | 18.7±37.92 |
| Bernard | 0.158±0.160 | 284.4±200 | 0.158±0.0907 | 284.4±11.18 | 0.158±0.0081 | 284.4±489 |

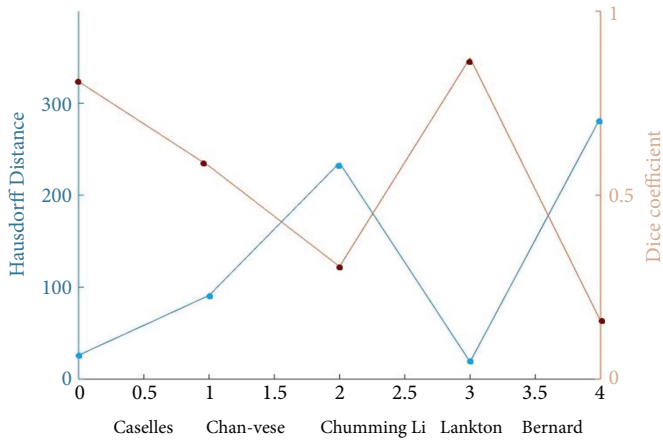



Figure 3: Graphical representation of mean average values of similarity index parameters.

Mean average value defines significant variations among all the types of active contour techniques used for the process of segmentation of the synovial region. But the values possess slight similarity between the two active contour methods. The slight deviation in the synovial segmentation between the two methods is illustrated in Figure 3 which represents the mean values of the similarity index parameters of the five active contour methods.

The standard error measurement describes the accuracy of the predicted value. The difference in standard error between Caselles and Lankton active contours is 0.0021, which is really insignificant. So there is a little more similarity among the geodesic and localized region based active contours.

Average variance is derived to find out the significant difference between the methods. The difference can be defined appropriately using F-test which is performed over the variance value. The F-test results shows that the Caselles geodesic active contour and Lankton localized region based active contour method are likely significant which shows that the region segmented using these methods is not similar. The result of the F-test of the geodesic active contour Caselles and localized region based active contour Lankton is shown in Table 3.

As a result of statistical analysis, Caselles and Lankton have slightly significant difference and prove to possess dissimilarity in the process of segmentation of synovial region. For further determination of the exact segmentation technique, the performance metrics and statistical features obtained from the region undergo classification process. In this process, support vector machine (SVM) classifier is used for training and identification of the significant segmentation technique for synovial region detection. Localized region based active contour Lankton is described as more significant with the help of the results of classification. These results are defined with confusion matrix and scatter plot of the trained features of the synovial region from the ultrasound images. In Figure 4, the confusion matrix is defined with accuracy rates of each category of segmentation. With these results, Lankton is determined as the significant nature of the method in detection of synovial region. Features extracted from the synovial region possess significant variations and localized region based active contour is described with the accuracy of the confusion matrix based on the true positive and false negative rates.

From these results, Localized region based active contour is more efficient method of active contours for synovial region segmentation by training and classifying the features like performance metrics and statistical values. These features define the appropriate classification of the region which is performed with this Lankton active contour.

## 4. Conclusion

In recent days arthritis has become a significant health problem. Early diagnosis and treatment help the patients to lead normal life. A method was presented to evaluate the active contour segmentation algorithms to segment the synovial region from arthritis affected finger ultrasound image. Performance analysis metrics like Dice coefficient and

Table 3: F-Test for Variances of Caselles and Lankton Method.

| F-Test Two-Sample for Dice coefficient Variances | | | F-Test Two-Sample for Hausdorff distance Variances | | |
|---|---|---|---|---|---|
| Mean | 0.845719 | 0.829609 | Mean | 19.47198 | 27.57708 |
| Variance | 0.04031 | 0.045104 | Variance | 37.91567 | 103.3178 |
| Observations | 8 | 8 | Observations | 8 | 8 |
| df | 7 | 7 | df | 7 | 7 |
| F | 0.893709 | | F | 0.366981 | |
| P(F<=f) one-tail | 0.442978 | | P(F<=f) one-tail | 0.104682 | |
| F Critical one-tail | 0.264058 | | F Critical one-tail | 0.264058 | |

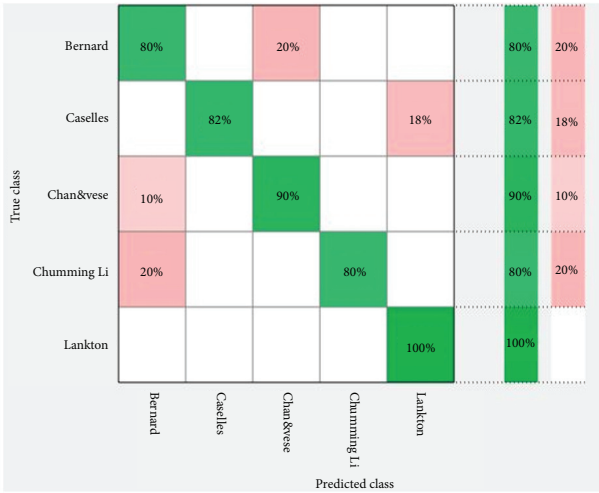



Figure 4: Confusion matrix of the classification of active contour segmentation techniques.

Hausdroff distance and statistical analysis metrics like standard error and F-test shows the significant difference between the two segmentation method (Caselles and Lankton) for synovial region. Further classification is performed for the derived features such as performance metrics and statistical values. Higher accuracy is described for Lankton as the result of classification process. Hence the output of the research work shows that Lankton method is the best method for synovial region segmentation from ultrasound images.

## Data Availability

The ultrasound images used to support the findings of this study were supplied by Krystian.Radlak under license agreement from MEDUSA Project and so cannot be made freely available. Requests for access to these data should be made to Krystian.Radlak, krsytian.radlak@polsl.pl.

## Additional Points

The complete research work concentrates on evaluation of contour based segmentation techniques.

## Conflicts of Interest

There are no conflicts of interest among the authors with regard to the proposed methodology and performance evaluation for contour based segmentation techniques for ultrasound images using statistical analysis.

## References

[1] R. Sharma, Ed., *Epidemiology of Musculoskeletal Conditions in India*, International Journal on Computer Science and Engineering (IJCSE), New Delhi, India, 2012.

[2] A. Bk, J. Segen, K. Wereszczyski, P. Mielnik, M. Fojcik, and M. Kulbacki, "Detection of linear features including bone and skin areas in ultrasound images of joints," *PeerJ*, vol. 2018, no. 3, 2018.

[3] G. Schett, "Synovitis–an inflammation of joints destroying the bone.," *Swiss Medical Weekly*, vol. 142, p. w13692, 2012.

[4] M. Krishnaveni, "Quantitative evaluation of Segmentation algorithms based on level set method for ISL datasets," *International Journal on Computer Science and Engineering (IJCSE)*, vol. 3, pp. 2361–2369, 2011.

[5] D. Barbosa, T. Dietenbeck, J. Schaerer, J. D'Hooge, D. Friboulet, and O. Bernard, "B-spline explicit active surfaces: an efficient framework for real-time 3-D region-based segmentation," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 241–251, 2012.

[6] D. Reska, C. Boldak, and M. Kretowski, "A Texture-Based Energy for Active Contour Image Segmentation," in *Image Processing Communications Challenges, Advances in Intelligent Systems and Computing*, R. Choras, Ed., vol. 313, Springer, Cham, 2015.

[7] M. Airouche, L. Bentabet, and M. Zelmat, "Image Segmentation Using Active Contour Model and Level Set Method Applied to Detect Oil Spills," in *Proceedings of the In Proceedings of the World Congress on Engineering*, vol. I, pp. 846–850, London, U.K, 2009.

[8] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky, "Fast geodesic active contours," *IEEE Transactions on Image Processing*, vol. 10, no. 10, pp. 1467–1475, 2001.

[9] T. Chan and L. Vese, "An active contour model without edges," in *Scale-Space Theories in Computer Vision*, vol. 1682 of *Lecture Notes in Computer Science*, pp. 141–151, Springer, Berlin, Germany, 1999.

[10] P. Getreuer, "Chan-Vese Segmentation," *Image Processing On Line*, vol. 2, pp. 214–224, 2012.

[11] V. Caselles, F. Catté, T. Coll, and F. Dibos, "A geometric model for active contours in image processing," *Numerische Mathematik*, vol. 66, no. 1, pp. 1–31, 1993.

[12] O. Bernard, D. Friboulet, P. Thevenaz, and M. Unser, "Variational B-spline level-set: a linear filtering approach for fast deformable model evolution," *IEEE Transactions on Image Processing*, vol. 18, no. 6, pp. 1179–1191, 2009.

[13] C. Li, C. Y. Kao, J. C. Gore, and Z. Ding, "Minimization of region-scalable fitting energy for image segmentation," *IEEE Transactions on Image Processing*, vol. 17, no. 10, pp. 1940–1949, 2008.

[14] S. Lankton and A. Tannenbaum, "Localizing region-based active contours," *IEEE Transactions on Image Processing*, vol. 17, no. 11, pp. 2029–2039, 2008.

[15] A. Khadidos, V. Sanchez, and C.-T. Li, "Active contours based on weighted gradient vector flow and balloon forces for medical image segmentation," pp. 902–906.

[16] MEDUSA, "Automated assessment of joint synovitis activity from medical ultrasound and power Doppler examinations using image processing and machine learning methods," http://eeagrants.org/project-portal/project/PL12-0015.

[17] K. Radlak, N. Radlak, and B. Smolka, "Automatic detection of bones based on the confidence map for Rheumatoid Arthritis analysis," in *Proceedings of the 5th Eccomas Thematic Conference on Computational Vision and Medical Image Processing, VipIMAGE 2015*, pp. 215–220, Spain, October 2015.

[18] P. Mielnik, M. Fojcik, J. Segen, and M. Kulbacki, "A Novel Method of Synovitis Stratification in Ultrasound Using Machine Learning Algorithms: Results From Clinical Validation of the MEDUSA Project," *Ultrasound in Medicine & Biology*, vol. 44, no. 2, pp. 489–494, 2018.

[19] T. Dietenbeck, M. Alessandrini, D. Friboulet, and O. Bernard, "Creaseg: A free software for the evaluation of image segmentation algorithms based on level-set," in *Proceedings of the 2010 17th IEEE International Conference on Image Processing, ICIP 2010*, pp. 665–668, Hong Kong, September 2010.

[20] A. A. Taha and A. Hanbury, "An efficient algorithm for calculating the exact hausdorff distance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 11, pp. 2153–2163, 2015.

Hindawi
Advances in Multimedia
Volume 2018, Article ID 3521720, 27 pages
https://doi.org/10.1155/2018/3521720

*Research Article*

# A New Semisupervised-Entropy Framework of Hyperspectral Image Classification Based on Random Forest

**Mengmeng Sun,[1] Chunyang Wang ,[1,2] Shuangting Wang,[1,2] Zongze Zhao,[1,2] and Xiao Li[1]**

[1]*School of Surveying and Land Information Engineering, Henan Polytechnic University, Jiaozuo 454000, China*
[2]*Henan Province Engineering Technology Research Center of Space Big-Data Acquisition Equipment Development and Application, Henan Polytechnic University, Jiaozuo 454000, China*

Correspondence should be addressed to Chunyang Wang; hpu_wcy@163.com

Received 26 May 2018; Accepted 14 August 2018; Published 4 September 2018

Academic Editor: Lei Zhang



The purposes of the algorithm presented in this paper are to select features with the highest average separability by using the random forest method to distinguish categories that are easy to distinguish and to select the most divisible features from the most difficult categories using the weighted entropy algorithm. The framework is composed of five parts: (1) random samples selection with (2) probabilistic output initial random forest classification processing based on the number of votes; (3) semisupervised classification, which is an improvement of the supervision classification of random forest based on the weighted entropy algorithm; (4) precision evaluation; and (5) a comparison with the traditional minimum distance classification and the support vector machine (SVM) classification. In order to verify the universality of the proposed algorithm, two different data sources are tested, which are AVIRIS and Hyperion data. The results show that the overall classification accuracy of AVIRIS data is up to 87.36%, the kappa coefficient is up to 0.8591, and the classification time is 22.72s. Hyperion data is up to 99.17%, the kappa coefficient is up to 0.9904, and the classification time is 8.16s. Classification accuracy is obviously improved and efficiency is greatly improved, compared with the minimum distance and the SVM classifier and the CART classifier.

## 1. Introduction

As shown in Figure 1, hyperspectral remote sensing image technology contains a lot of potential information and integrates the spectral and spatial dimensions [1, 2]. It has the characteristics of a continuous spectrum and a unified spectrum, and it can be used to interpret an object with high spectral diagnosis ability [3]. Considering these advantages, we use hyperspectral remote sensing image in the experiments in this paper.

Owing to the limitations of technology, the mining of spatial dimension information is obviously deficient. How to fully excavate a large amount of information hidden in hyperspectral remote sensing images is a key issue in the literature. To be of scientific merit, classification technology is the important technology for processing hyperspectral remote sensing images [4]. Using hyperspectral images to classify ground objects is one of the core contents of the application of hyperspectral remote sensing technology [5] and the classification results have great application value that may apply to land cover [6, 7], resource surveys [8–12], environment monitoring [13, 14], coverage prediction [15, 16], military exploration [17, 18], and other fields.

However, in the process of application, we mainly encounter the problems of the Hughes phenomenon, Bellman's disaster, nonlinear distribution of data in feature space, and so on, which may result in the information being ambiguous during the process of hyperspectral image classification [19]. What is more, it is really important to fine-tune the spectral features provided by spectral images considering that the traditional classification methods based on a single classifier cannot meet the classification needs of hyperspectral remote sensing [20] and most of the traditional algorithms take pixels as the basic unit for classifying [21, 22], without considering the spatial features of remote sensing images. This results in the algorithm not being able to deal with the "isomorphism problem" of the same objects effectively [23], and many noise points easily appear in the

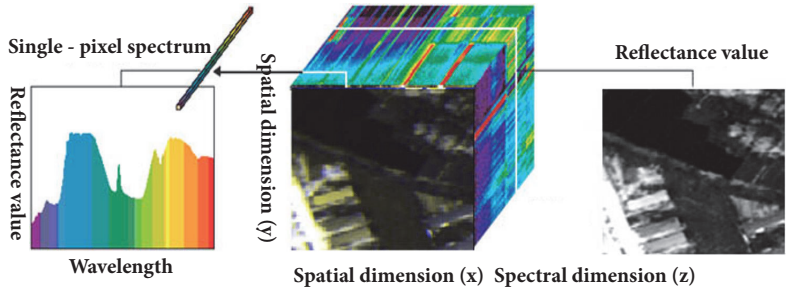



Figure 1: The example of hyperspectral image data cube.

interior of the ground objects in the classification results. The fine spectral features provided by spectral images can be used to distinguish objects with subtle differences, including those with high similarity to natural backgrounds, and the distribution of background information is different from the assumption of the model. The object size is affected by the subpixel level, and sometimes the false alarm rate is too high [24]. Therefore, hyperspectral image target detection technology has great potential value in the field of public security and national defense. Hyperspectral image target detection requires the diagnostic spectral characteristics of a target, and it is applied to many varieties of target spectrum in practice [25]. Therefore, it is necessary to develop a stable and reliable method.

To be of scientific merit, it is necessary to make full use of the rich space and spectral information of hyperspectral remote sensing images to interpret the objects of observation. This has become a hotspot in the research field and is the frontier field in recent years. What is more, it has great application value and broad development prospects in many related fields. As has been illustrated, it is essential to improve the extraction ability of ground object information [26].

In view of the characteristics of hyperspectral remote sensing images, the classification of random forest algorithms may be a good choice. The random forest algorithm is a supervised self-training classifier [27–30] that consists of many classification trees, each of which completes its own sorting operation. The final classification results are determined by the voting results of each classification tree [31]. The random forest algorithm is a classification method based on the principle of the classification and regression tree (CART) decision trees algorithm, which is composed of a series of CART decision trees. The classification results are voted on by the decision trees. The final feature category relies on the classification result with the largest number of votes [32]. The Gini index is used to measure the classification results.

As an excellent classifier model, random forest also provides a new idea for image classification [33]. Random forest can connect independent variables with dependent variables by generating a lot of classification trees, which can successfully calculate the nonlinear and interactive effects of variables, even when there is a high degree of interference. Considering those characteristics, random forest is a good choice that can deal with the problems of local extremum in hyperspectral remote sensing images, the difference between different categories of ground object, and the slow speed of running the operation. Many explanatory variables can be predicted [34]. This algorithm is a summary of classification trees and gives importance to all variables. However, it is still relatively robust in the face of data loss and imbalance.

In view of the large amount of information contained in hyperspectral remote sensing images and the fact that they are still a new subdivision of spectral imaging remote sensing technology, it is very difficult to fully exploit the potential information contained therein and to remove the difficulty in obtaining training samples. Some of the traditional supervised classification methods are not very practical; however, the unsupervised classification method does not require training samples owing to the limitation of its classification accuracy. Owning to the above reasons, it may be wise to use semisupervised classification to classify ground objects [13]. Semisupervised classification is a learning algorithm that is an active learning machine. Integrated learning methods are a very important research direction in the field of machine learning. Semisupervised classification is an active learning algorithm that focuses on the use of labeled and unlabeled samples [35] to obtain high-performance classifiers. The purpose of ensemble learning is to improve the accuracy of weak learning classifiers by integrating multiple learning devices. Semisupervised ensemble learning is a new machine learning method that combines semisupervised learning and integrated learning to improve the generalization performance of classifiers [36].

Semisupervised learning uses both labeled and unlabeled samples in the process of training. With the growth of information, the classification problem becomes more and more complicated, while the semisupervised classification algorithm obtains only a small number of classification samples. A small number of labeled samples are used to train the classification model in this classification algorithm.

Semisupervised learning is a self-training machine learning method that makes full use of a small number of labeled samples and a large number of unlabeled samples. In view of the high cost of sample marking, the rapid development of spectral imaging technology, and the emergence of hyperspectral images, it is very meaningful to study the classification method of semisupervised machine learning. This invention can optimize the classification performance of hyperspectral remote sensing images and improve their classification accuracy and efficiency.

Random forest not only shows high classification performance but also has fewer parameters to be adjusted, and it is fast and efficient in the field of machine learning [37]. There is no need to worry about overfitting and strong noise tolerance [38]. Excellent random performance makes it widely used in intelligent information processing, bioinformatics, finance, diagnosis of faults, recognition of images, industrial automation, and other fields [39]. It has attracted widespread attention and achieved great success. Industrial automation and other fields have widely used it and achieved great success, attracting extensive attention [37, 40]; although many scholars have conducted extensive research on random forests and achieved many remarkable things [41], there are still some limitations and shortcomings, leaving some room for improvement [16]. Therefore, in order to deal with the problems of the high dimensionality and large amount of data of hyperspectral remote sensing images and the difficulty in extracting samples' extraction characteristics [42, 43], a robust classification method with high accuracy is urgently needed [44].

In this paper, a semisupervised random forest hyperspectral remote sensing image classification method based on weighted entropy [45–49] is proposed. It can be classified by randomly selecting the number of training samples with 10% or 5% labeled samples. A classifier model [50–52] is constructed that uses a random forest based on a CART decision tree with probabilistic output; it is a supervised classifier [53] that integrates multiple weak classifiers. The classifier predicts ground objects according to the number of votes cast. Then, a weighted entropy algorithm is used to give the class with the highest weighted values and return weight values, which is predicted by the model. The objects with larger weighted entropy values are sorted, and objects that account for 5% or 10% of the total sample increase are added to the training samples to form new training samples; then, the prediction and classification steps are carried out again. The above steps are repeated until the conditions for iteration stop are satisfied or the labeled samples are used up; the remaining samples are used for accuracy evaluation, which uses classifier performance detection [50, 54].

This classification method is economical and suited to the properties of hyperspectral remote sensing images. It is of high value to researchers wishing to classify large, dense areas. The purpose of the algorithm presented in this paper is to select the features with the highest average separability by using the random forest method to distinguish the categories that are easy to distinguish and to select the most divisible features of the most difficult categories by using the weighted entropy algorithm. The framework is composed of five parts: (1) random samples selection with (2) probabilistic output initial random forest classification processing based on the number of votes; (3) semisupervised classification, which is the improvement of the supervision classification of random forest based on the weighted entropy algorithm; (4) precision evaluation; and (5) a comparison with the traditional minimum distance classification and SVM classification. In order to verify the universality of the proposed algorithm, two different data sources are tested: AVIRIS and Hyperion data. The results show that the overall classification accuracy of AVIRIS data is up to 87.36%, the kappa coefficient is up to 0.8591, and the classification time is 22.72s. The Hyperion data is up to 99.17% accurate, the kappa coefficient is up to 0.9904, and the classification time is 8.16s.

## 2. Materials and Methods

The imaging spectrometer acquiring the hyperspectral image data cannot be directly applied and classified analysis, which needs to be analyzed. Therefore, the preprocessing of hyperspectral remote sensing image in general includes atmospheric radiation correction, geometry correction, and noise removal [45, 55–57]. In the preprocessing of hyperspectral image, radiometric correction is the mainly steps.

In this experiment, there are three classification methods adapted, which are the minimum distance classification method, the support vector machine classification method, and the semisupervised classification method proposed in this paper which uses the features of the random samples and random band selection [58] of random forest based on the weighted entropy [31]. The semisupervised classifier [59] is trained with labeled samples data and determined the parameters of the classifier until it ensures that the training samples are matched with the verification samples. In the case of mutual independence, the fitting test of classification parameters is carried out to determine the applicability of the parameters [60]. The minimum distance classifier and support vector machine classifier are used to test the function of these classifiers with the same number of training samples. The accuracy evaluation is carried out to quantitatively evaluate results of the experimental method to determine the effect of the classifiers.

In order to verify the universality of the proposed algorithm, two different data sources are tested in this paper. In fact, there will be three parts in this chapter. First of all, the study areas and the training samples will be illustrated. In the second part, the selection of classification algorithms will be illustrated. In the third part, the processing of the algorithm that is proposed in the paper will be illustrated.

*2.1. Study Areas and the Training Samples.* In order to verify the universality of the proposed algorithm, two different data sources are tested in this paper, which are AVIRIS data and Hyperion data. The AVIRIS imaging spectrometer, also known as the airborne visible infrared imaging spectrometer, was developed in 1987 by NASA's Jet Propulsion Laboratory (JPL). It covers a wavelength range of 400,500 nm and it is almost the full wavelength of solar radiation. Because of its rich spectral information, AVIRIS provides a large amount of data for various science and applications. The images used in this paper are located at the Kennedy Space Center in Florida, USA. The acquisition time is March 1996. The image is 614 pixels wide and 512 pixels high, with a total of 15 pixels five bands with a spectral resolution of 10 nm and a spatial resolution of 18 meters. The training data are selected on the basis of images provided by Landsat Thematic Mapper.

The land cover types in the region are divided into 13 major categories, namely, scrub, willow, CP-Hammock,

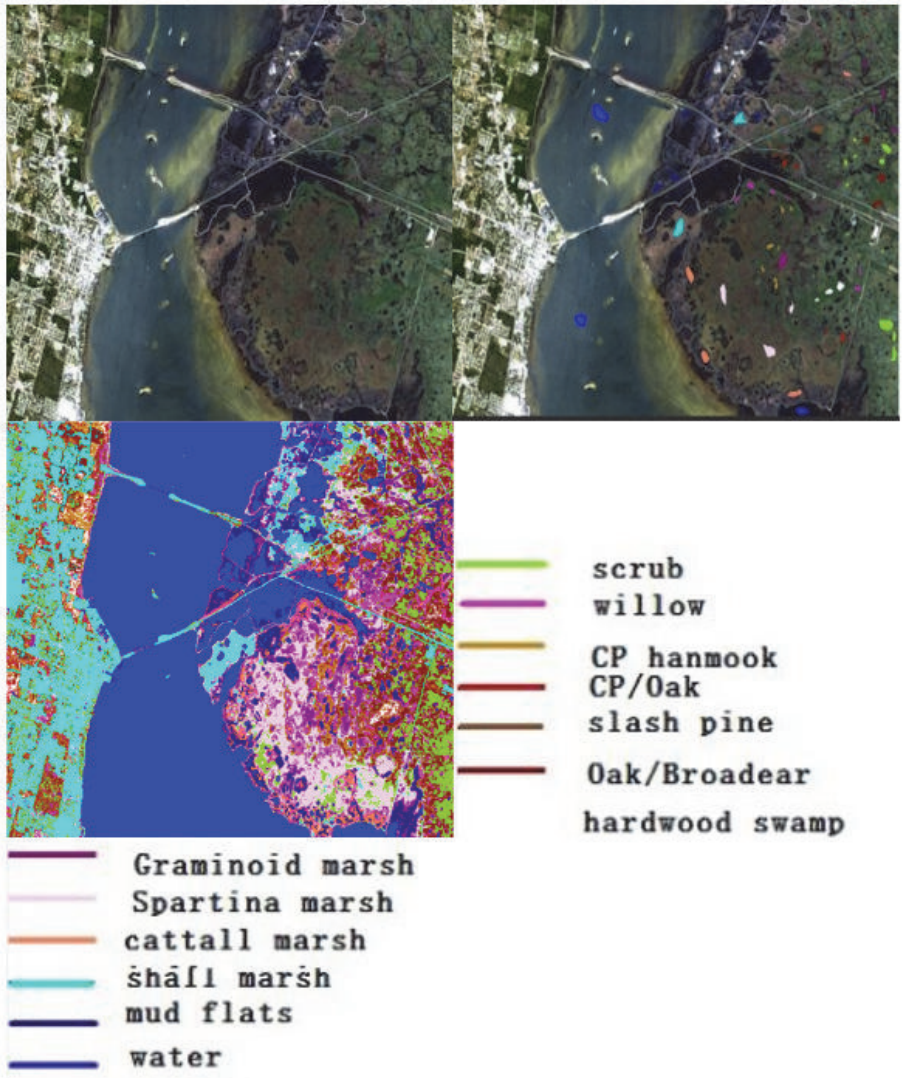



Figure 2: The synthesis true color image (bands 31, 21, and 11) and the distribution of training samples of AVIRIS data and the ground truth data.

CPP/Oak, slash-Pine, Oak/Broad-leaf, Hardwood, Graminoid-marsh, Spartina-marsh, Cattail marsh, Saltmarsh Mud flats, and water. In the classification of remote sensing, the selection of the training samples area directly determines the classification results. Using Google Earth to select the training samples area of remote sensing classification is an excellent way. The ground truth data can be used in two ways: one is the standard classification diagram and the other is the selected area of interest (validation samples area). In this paper, the standard classification diagram is used as the ground truth data. The true color composites images of different samples [27]. AVIRIS data, the distribution of training samples, and the ground truth data are shown in Figure 2.

The Hyperion imaging spectrometer, mounted on the EO-1 satellite platform, was launched by NASA on November 2000. It covers a wavelength range of 400 to 2500 nm, has 220 wavelengths, and has a spectral resolution of 10 nm and a spatial resolution of 30 meters. The image used in this paper is located in Dali city, Yunnan province, China. This product was created by US Geological Survey. The product contains EO-1 Hyperion data file, hierarchical data format, or geographical mark image file format (TIFF). EO-1 has launched a one-year technical demonstration validation mission. NASA (NASA) and the US Geological Survey (USGS) agreed to continue the EO-1 program as an extended mission. Information about EO-1 satellites and Hyperion sensors can be found at the USGS and NASA's Web site: http://eo1.usgs.gov http//eo1.gsfc.nasa.gov. The date of acquisition of this image is January 2014. In order to use it conveniently, this paper cuts out an experimental area of 529 pixels high and 256 pixels wide. After atmospheric correction and geometric correction, a total of 72 bands were selected for analysis after removing the low SNR band. The images of land cover mainly include bare land, low residents, low vegetation, broad-leaved forest, and six types of water body. In the classification of remote sensing, the selection of the

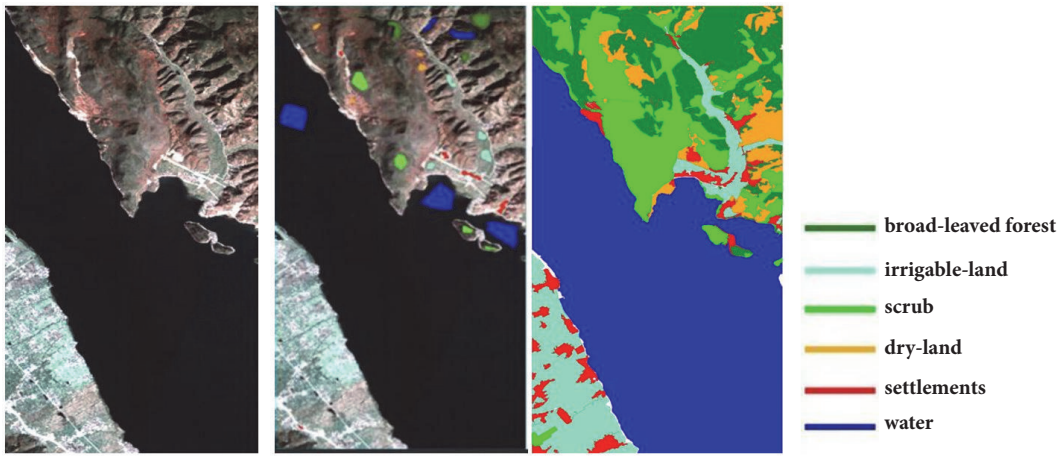



Figure 3: The synthesis true color image (bands 29, 20, and 18) and the overlaid image of training sets of Hyperion data and the ground truth data.

training samples area directly determines the classification results. Using Google Earth to select the training samples area of remote sensing classification is an excellent way. The ground truth data can be used in two ways: one is the standard classification diagram and the other is the selected area of interest (validation samples area). In this paper, the standard classification diagram is used as the ground truth data. Hyperion data true color composite image and the distribution of training samples and the ground truth data are shown in Figure 3.

Classification is one of the main problems in the field of remote sensing. As a tool to solve the problem, classifier is always a hot topic. The commonly used classifiers include decision tree, logical regression, Bayes, and neural network. These classifiers all have their own performance characteristics. In this paper, three kinds of classifiers are mainly used, that is, the traditional classifier based on minimum distance and the support vector machine classifier and the classifier proposed in this paper.

The essence of classification is to select the appropriate discriminant function according to the law of probability and statistics and to establish a reasonable discriminant model to separate the discrete clusters in the image and to make the judgment and classification. Through the statistics and calculation of the regions of interest, the mean and variance parameters of each category are obtained to determine a classification function, and then each pixel in the image to be classified is brought into the classification function of each category. The category with the largest return value of the function is regarded as the category of the scanned pixel to achieve the effect of classification.

There are two strategies for selecting separability judgments [61, 62]: (1) select the features with the highest average separability; (2) select the most divisible feature of the most difficult categories. The first strategy is difficult to take care of a more centralized category [63, 64]. If this strategy is used, the choice of balanced care of all types can make up for its shortcomings. Second strategies can take care of the most difficult categories, but it may miss some of the biggest feature which makes the separability and decreases the classification accuracy [65, 66].

In practical application, the thought of the two strategies should be integrated to achieve balance between efficiency and pattern distribution. If the distribution is more uniform, both strategies should be chosen; but if the pattern distribution is not uniform, to select the first strategy, we must consider the validity of the separability criterion and the most difficult category to improve the classification accuracy.

The aim of this algorithm presented in this paper is to select the features with the highest average separability by using the method of random forest and to distinguish the categories that are easy to distinguish. Then, the weighted entropy algorithm is used to select the most divisible features of the most difficult category. This algorithm of the random forest and the weighted entropy of the optimal parameter combination combined into the classifier proposed in this paper to classify not only can improve the purpose of the classification, but also can improve the accuracy and efficiency of the classification.

By using the traditional minimum distance and support vector machine (SVM) classification method, the AVIRIS and the Hyperion data are used to carry out many experiments on 30% of the total samples and 40% of the classification samples and 50% of the classification samples, which are randomly selected. The experiment is carried out with the classification algorithm proposed in this paper. Among them, there are two methods for sample selection.

Method 1: 5% samples are randomly selected as training samples [66, 67] and the rest samples are used as test data. Then, the performance of the classifier is reflected by the degree of fitting of the classifier and the real ground object [68]. The time of iteration is 1 and the initial number of each feature is 16. Then the most weighted feature category is taken as the category of the feature. Then the weighted entropy algorithm is used to give the category with the highest weighted, which returns the weight of the class predicted by the model [69–75]. 5% additional samples were selected and added to the training sample to form a new training sample,

and then the prediction classification was conducted again, so that the iteration took place until 10 iterations. Among them, 6 iterations, 8 times, and 10 times correspond to 30% of the total number of samples and 40% of the total number of samples and 50% of the total number of samples for many experiments. All of them pick out the total classification accuracy and kappa coefficient of the three corresponding numbers and calculate the corresponding time and average value in order to eliminate the random band. The random training samples corresponding to the minimum distance classifier and the support vector machine classifier are 30%, 40 %, and 50% of the training samples

Method 2 of the experiment: 10% samples were randomly selected as training samples, and the rest were used as test data. The time of iteration is 1 and the initial number of each feature is 16. Then the most weighted feature category is taken as the category of the feature. Then the weighted entropy algorithm is used to give the category with the highest weighted, which returns the weight of the class predicted by the model. 10% additional samples were selected and added to the training sample to form a new training sample, and then the prediction classification was carried out again, so that each iteration took place five times. The number of iterations 3 times 4 times and 5 times corresponds to 30% of the total number of samples and 40% of the total number of samples and 50% of the total number of samples. The mean value of the total classification accuracy, the kappa coefficient, and the corresponding time are calculated to eliminate the errors caused by the randomness, and the random training samples corresponding to the minimum distance classifier and the support vector machine classifier are 30 %, 40 %, and 50 %.

*2.1.1. Training Samples Selection and Testing Samples Selection.* The purpose of the training samples is to confirm the parameters of the mathematical model. After training, the model system can be regarded as having been established. The purpose of the test samples is to ascertain the function of the model and whether the degree of fitting between the model and real events is small.

In the classification of remote sensing, the selection of the training samples area directly determines the classification results. Using Google Earth to select the training samples area of remote sensing classification is an excellent way and the process is showed in the following:

(1) Put your research area boundary through ArcGIS tool to KML, the research area in the Google Earth display.

(2) Sketch the type of object you want on Google and sketch "right-click your folder-save location as", KML file.

(3) In ArcGIS, convert the KML to layer tool to the ArcGIS-recognized layer, and then the data export the shape file format (note similar dissolved and projection conversions).

(4) Open your samples vector file in Envi, then export it to the ROI file to the image of you want to classify, and select the attribute for your own defined the type of objects, when it converts to ROI.

(5) You can see ROI in the main image window you want to categorize.

*Training Samples Selection Principle*

(1) Samples distribution should be as wide as possible.

(2) Choose the pure pixels, not the areas where different features are transferred.

(3) The relationship between the number of samples and the type of the samples is twice or more.

(4) The real samples should be consistent with the experimental samples.

(5) The separability of the training samples is a parameter of reference value to judge the function of the training samples.

(6) The correlation coefficient within the class should be large, and the correlation coefficient between the classes should be small.

*Test Samples Selection Principles*

(1) Band selection should be consistent with the training samples.

(2) Find the region of interest in other places, which does not coincide with the area of interest of the training samples.

(3) The real samples should be consistent with the experimental samples.

(4) Separability is a parameter of reference value to judge the function of test samples.

(5) The correlation coefficient within a class should be large, and the correlation coefficient between classes should be small.

### *2.2. Classification Algorithms Selection*

*2.2.1. The Minimum Distance Classifier.* The nearest neighbor method classifies new samples from unknown categories according to a set of samples known to each class of samples, and the classification is based on calculating the distance between the features of the new samples and those of the samples in the set of samples in turn. The nearest neighbor algorithm is mainly based on a limited number of adjacent samples; thus, it is more suited to unclassified data sets with more overlapping parts. Although the nearest neighbor algorithm depends on the limitations of the theorem to some extent, it only needs to consider the adjacent samples' information in the classification, which can solve the problem of sample imbalance. The disadvantage of the neighbor algorithm is that the time complexity of the algorithm is high. The distance between the samples should be classified and each sample in the known samples space should be calculated in turn, and then they will be sorted. In this case, we can process the known samples the first time, remove some samples, and reduce the number of comparisons in the classification process, thus reducing the time consumed by the algorithm [11].

The minimum distance classification is the most basic classification method in the classifier. It is a classification method by calculating the distance between unknown class vector X and the center vector of each previously known class and then reducing the vector X to be classified as the smallest of these distances.

In an n-dimensional space, the minimum distance classification first calculates the mean values of each dimension

of each known class X (expressed as a vector). A mean value is formed, which is represented by a vector (the name of a class, the samples' feature set of category A, the first dimensional feature set of class A, and the mean value n of the first one-dimensional feature set as the total characteristic dimension). The mean value of another category is calculated (expressed as a vector) and used. It is regarded as a samples' feature vector x to be classified. The distance between the two samples is the variable to be calculated. The basic idea of the minimum distance classifier is to generate a central vector representing the class according to the arithmetic average of the training set (K:1,2,$\cdots$,M; M is the number of classes), for each data tuple to be categorized X, its distance from the uk is calculated, and finally, it is determined that X belongs to the nearest class.

Here are two values X=[x1,x2,$\cdots$,xn] and uk = [uk1,uk2,$\cdots$,ukn] and c represents category and it belongs to {cl,c2,$\cdots$,c}. Take the Euclidean distance as an example; the formula for calculating the distance is as follows:

$$\begin{aligned} d(x,u_i) &= |x-u_i|^2 = (x-u_i)^T (x-u_i) \\ &= x^T x - \left(x^T x + u_i x - {u_i}^T u_i\right), \end{aligned} \tag{1}$$

Then look for the minimum value in the two groups. If the former is the smallest, then X belongs to class A, and if the latter is small, then X belongs to class B.

There are many different methods for calculating the distance of classification at present and it is the most common method of calculating distance, Euclidean distance.

Euclidean distance is the most easily understood method of distance calculation, derived from the distance formula between two points in the Euclidean space.

The Euclidean distance between two points of A (x1,y1) and B(x2,y2) on a two-dimensional plane is as follows:

$$d_{1,2} = \sqrt{(x_1-x_2)^2 + (y_1-y_2)^2}, \tag{2}$$

Euclidean distance between two points of A (x1,y1,z1) and B(x2,y2,z2) on two points in three dimensional space is as follows:

$$d_{1,2} = \sqrt{(x_1-x_2)^2 + (y_1-y_2)^2 + (z_1-z_2)^2}, \tag{3}$$

Euclidean distance between two points of X1 (expressed as a vector) and X2 (expressed as a vector) on two n-dimensional vectors is as follows:

$$d_{1,2} = \sqrt{\sum_{k=1}^{n} (x_1-x_2)^2}, \tag{4}$$

The nearest neighbor method classifies new samples in unknown categories according to a set of samples known to each class of samples, and the classification is based on calculating the distance between the features of the new samples and those of the known set of samples in turn. The nearest neighbor algorithm is mainly based on a limited number of adjacent samples, so it is more suited to unclassified data sets with more overlapping parts. Although the nearest neighbor algorithm depends on the limitation theorem to some extent, it only needs to consider the adjacent samples' information in the classification.

The disadvantage of the neighbor algorithm is that the time complexity of the algorithm is high and the distance between the samples to be classified in the classification process and each sample in the known sample space should be calculated in turn. The nearest samples should be considered after they are sorted. In this case, we can preprocess the known samples' points and remove some samples to reduce the number of comparisons in the classification process, thus reducing the time consumed by the algorithm [11].

*2.2.2. Support Vector Machine Classifier.* To map the sample space to a high or even infinite dimensional feature space by means of a nonlinear mapping plane, SVM may be a good method. It can transform the nonlinear separable problem in the original sample space into one in the feature space. A linear separable problem involves scaling up and can be linearized. Raising the dimension entails mapping samples to a high-dimensional space. In general, this will increase the computational complexity and even cause a "dimensionality disaster". However, as a matter of classification, regression, and so on, a sample set may not be linearly processed in a low-dimensional sample space. Linear partitioning (or regression) can be realized on a linear hyperplane. The SVM method can solve this problem skillfully by applying the expansion theorem of the kernel function. There is no need to know the explicit expression of nonlinear mapping. Since the linear learning machine is built in the high-dimensional feature space, comparing it with the linear model, the computational complexity is almost not increased. The catastrophe of dimensionality can be avoided to some extent thanks to the expansion of the kernel function and the theory of calculation.

SVM is often used in classification scenarios, and its classification effect is very good. Compared with other classification methods, only when the training samples reach a certain number it can achieve a better result. The algorithm can also obtain satisfactory results on small samples when the number of samples is limited and the classification effect is difficult to guarantee.

The basic principles of the algorithm are as follows: different points and lines represent the classification target and the classification hyperplane, respectively [11]. The optimal hyperplane linear of two classification support vector machines aims to find the optimal hyperplane, so that the two types of distances are maximal and they can be separated correctly. Suppose two classes of samples sets are known: $\{(x_1,y_1),(x_2,y_2),\cdots(x_n,y_n)\}$, $(x_i,y_i) \mid x_i \in R^n$, $y \in \{-1,+1\}, i = 1\ldots n$

The decision function is

$$f(x) = \operatorname{sgn}\left(g(x)\right) \tag{5}$$

The optimal hyperplane description is

$$w \cdot x + b = 0 \tag{6}$$

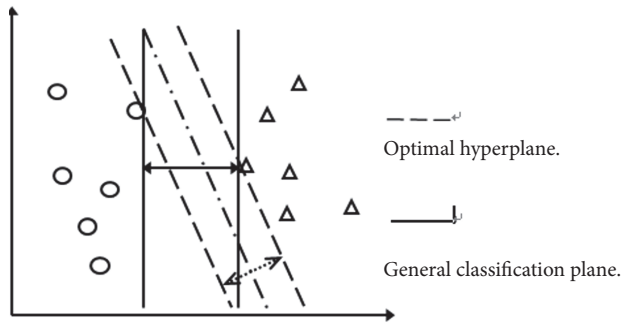



Figure 4: The comparison of optimal classification hyperplane and common classification plane.

The result of the classification is

$$\begin{aligned} w \cdot x + b \geq 0 \quad y_i = 1 \\ w \cdot x + b < 0 \quad y_i = 1 \end{aligned} \tag{7}$$

The function $w$ is the normal direction of hyperplane. So as to normalize it, the corresponding relation of classification hyperplane and hyperplane $w$ is established. Therefore, the distance between the nearest samples of judgment surface and judgment surface is $1/\|\mathbf{w}\|$, and the interval between the two types is $2/\|\mathbf{w}\|$. The samples data set is divided into two areas, i.e.,

$$\begin{aligned} \min_{i=1,2,\cdots n} \left|(\mathbf{w} \cdot \mathbf{x}_i) + b\right| = 1 \\ y_i \left[(\mathbf{w} \cdot \mathbf{x}_i) + b\right] = 1 \end{aligned} \tag{8}$$

The classification hyperplane $(\mathbf{w} \cdot \mathbf{x}) + b = 0$ can classify all samples correctly, that is,

$$y_i \left[(\mathbf{w} \cdot \mathbf{x}) + b\right] \geq 1, \quad i = 1, 2, \cdots n \tag{9}$$

To minimize $\|\mathbf{w}\|^2$, the classification surface is the optimal classification hyperplane. The comparison between the general classification surface and the optimal classification hyperplane is shown in Figure 4.

Solving the optimal hyperplane solution of two-class optimal hyperplane can be transformed into two-time programming problems. Constraint condition is

$$\min \Phi(\mathbf{w}) = \frac{1}{2} (\mathbf{w} \cdot \mathbf{w}) \tag{10}$$

$$y_i \left[(\mathbf{w} \cdot \mathbf{x}) + b\right] \geq 1, \quad i = 1, 2, \cdots n \tag{11}$$

Type (11) is a convex programming problem, using the Lagrange multiplier method to solve the upper formula, i.e.,

$$L(w, b, \alpha) = \frac{1}{2} (w \cdot w) - \sum_{i=1}^{n} \alpha_i \left\{ y_i \left[(w \cdot x_i + b\right] - 1 \right\} \tag{12}$$

Take the partial derivative of $w$ with respect to $b$, and set it to zero, and you get

$$\begin{aligned} \frac{\partial L}{\partial w} = 0 \Longrightarrow \\ w = \sum_{i=1}^{n} \alpha_i y_i x_i \end{aligned} \tag{13}$$

$$\begin{aligned} \frac{\partial L}{\partial b} = 0 \Longrightarrow \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \end{aligned} \tag{14}$$

Formula (13), (14) is brought into (12) and can be

$$W(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \left(x_i \cdot x_j\right) \tag{15}$$

According to the duality theory of Kuhn-tucker condition, the above problem can be transformed into dual problem, namely,

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \left(x_i \cdot x_j\right) \tag{16}$$

Constraint condition is

$$\begin{aligned} \alpha_i \geq 0, \\ \sum_{i=1}^{n} \alpha_i y_i = 0, \\ \alpha_i \left\{ y_i \left[w \cdot x_i + b\right] - 1 \right\} = 0 \end{aligned} \tag{17}$$

Solution Type (16): we can get the optimal classification function, namely,

$$\begin{aligned} f(x) &= \operatorname{sgn} \left\{(\mathbf{w} \cdot \mathbf{x}) + b\right\} \\ &= \operatorname{sgn} \left\{ \sum_{i=1}^{n} \alpha_i^* y_i \left(x_i \cdot x\right) + b^* \right\} \end{aligned} \tag{18}$$

In the result, most of $\alpha_i$ are equal to zero, and only the samples $x_i$ responding to the decision boundary distance of 1 are not equal to zero, so the samples $\alpha_i$ corresponding to this part of nonzero $\alpha_i$ are called support vectors. The training samples set will be only a small amount of samples, which can greatly reduce the process of construction and operation, so the efficiency and speed of the support vector machine (SVM) classification method are high.

When the training samples linear inseparable, namely, the training samples cannot be completely separated from the hyperplane of slack variables $\xi_i$, and error penalty adjustment parameters $C$ can be used to solve the transformation into and the optimal classification function is

$$\min \Phi(\mathbf{w}) = \frac{1}{2} (\mathbf{w} \cdot \mathbf{w}) + C \sum_{i=1}^{n} \xi_i \tag{19}$$

Constraint condition is

$$y_i\left(\mathbf{w}\cdot\mathbf{x}_i+b\right)-1+\xi_i\geq 0,\quad \xi_i\geq 0,\ i=1,2,\cdots,n \tag{20}$$

The function of error penalty adjustment parameter$C$ is to control the relationship between the upper bound of samples size and the complexity of the algorithm. The dual problem of transformation is exactly the same as that of the linear separable case but with different constraint conditions, i.e.,

$$\max_{\alpha} W(\alpha)=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j\left(x_i\cdot x_j\right) \tag{21}$$

Constraint condition is

$$\sum_{i=1}^{n}y_i\alpha_i=0;,\quad 0\leq\alpha_i\leq C,\ i=1,2,\cdots,n \tag{22}$$

So the final classification function is

$$f(x)=\operatorname{sgn}\{(\mathbf{w}\cdot\mathbf{x})+b\} \tag{23}$$

An advantage of SVM is that it does not require a lot of samples. This does not mean that the absolute number of training samples is very small, but it is smaller than other training classification algorithms. Under the same problem complexity, the number of samples required by SVM is relatively small, because of the introduction of the kernel function in SVM. Therefore, for high-dimensional samples, SVM is easy to deal with. The structural risk is minimal. This risk refers to the cumulative error between the approximation of the real model of the problem by the classifier and the real solution to the problem. SVM is good at dealing with the inseparability of sample data, mainly through relaxation variables (which are also called penalty variables) and kernel function technology.

*2.2.3. The Decision Tree Classifier.* The decision tree classification method is a kind of inductive classification algorithm which uses the learning of training samples to excavate the useful rules and use this rule to predict the new Xinji. Its rationale is for each input using a corresponding local model computed from the training data in the region [76]. The basic algorithm of decision tree classification is greedy algorithm, which is a top-down recursive method to construct decision. In each step, it takes the attribute of the best or optimal discrete value field in the current state, and it evaluates the attributes quantitatively by the information gain. The attribute is then established as a standard for partitioning until the data for each node belongs to the same category or no attributes can be used to split the data.

Conventional decision tree rules are generally based on experience and visual interpretation of artificial settings, subject to the influence of subjective factors, and classification and regression tree (Classification And Regression Trees, CART) method can automatically select the classification characteristics and determine the node threshold value. It is the representative of the decision tree model that can handle the nonnumeric data that other algorithms cannot handle [77].

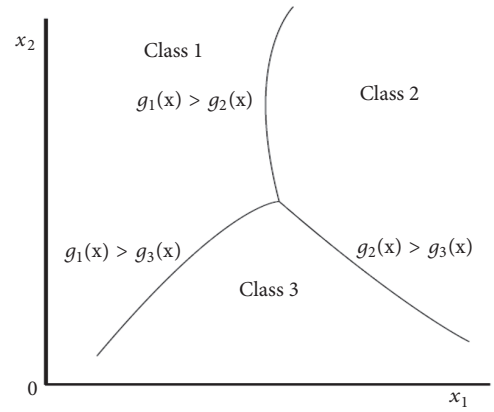



Figure 5: Class boundaries by all-at-once formulation.

The basic principle of the CART algorithm is a two-fork tree structure based on the cyclic analysis of the training dataset composed of the test variables and the target variables. CART is a supervised learning algorithm; that is, users must first provide a learning sample set (Learning samples) to build and evaluate the cart before using the cart for forecasting [78].The variable functions are as follows:

$$\mathrm{L}\coloneqq\left\{X_1,X_1,\cdots,X_m,Y\right\} \tag{24}$$

$$\begin{aligned}X_1&\coloneqq\left(x_{11},x_{11},\cdots,x_{1t_1}\right),\cdots,\\ X_m&\coloneqq\left(x_{m1},x_{m1},\cdots,x_{mt_n}\right),\end{aligned} \tag{25}$$

$$\mathrm{Y}\coloneqq\left(y_1,y_2,\cdots,y_k\right) \tag{26}$$

$X_1\sim X_m$ is called attribute vectors, and its properties are continuous and discrete. Y is called a label vector (label vectors) whose properties are contiguous and discrete. When $y_i$ is a continuous quantity value, it becomes a regression number. When $y_i$ is a discrete value, it becomes a classification tree.

Namely, we determine the decision functions:

$$\mathrm{gi}(x)>\mathrm{gj}(x),\quad \text{for j = i, j = 1,}\cdots\mathrm{n}. \tag{27}$$

In this formulation we need to determine n decision functions at all once. This results in solving a problem with a larger number of variables than the previous methods.

An example of class boundaries is shown in Figure 5. Unlike one-against-all and pairwise formulations, there is no unclassifiable region.

The way the CART algorithm chooses the split attribute is more interesting. The detailed steps are as follows: first, calculate the impurity and then use the Gini to compute the index. CART chooses a property with the highest information gain; the algorithm uses a greedy top-down approach and each internal node chooses the best classification attribute for the split node. Using the random forest proposed by Breiman is CART in the training process of the decision tree. The decision tree, which is attribute-value-based testing, will enter the training sets, which are divided into subsets, and each of them will be divided into a subset of a repeated

Table 1: The process of generate decision tree (using the given training data to produce a decision tree).

| **Algorithm:** Generate decision tree(using the given training data to produce a decision tree) |
|---|
| **Input:** Training data set samples, with discrete value attributes, the collection of candidate attributes Attribute list; |
| **Output:** A decision Tree; |
| **method:** (1) Create node N; |
| (2) If the samples are in the same Class C; |
| (3) returns N as a leaf node, with class C tag; |
| (4) If Attribute list is an empty then; |
| (5) returns N as the leaf node, marking the most common class in samples;//Majority Voting; |
| (6) Selecting the optimal classification attribute test attribute in Attribute list; Using information gain as attribute selection metric; |
| (7) The Mark node N is test attribute; |
| (8) for the known value AI in each test attribute; The Division of Samples |
| (9) It is grown from the node N with a condition of test attribute= ai; |
| (10) Set up SI as sample of test attribute=ai in samples; A partition; |
| (11) If si is an empty then; |
| (12) plus a leaf node, marked as the most common class in the samples;//majority vote |
| (13) else plus one by Generate decision tree (SI, Attribute_ List-test attribute) returns the node; |

recursively partitioned subset until the next node where all the elements have the same value or attribute value as given or some other stop conditions. We chose optimal segmentation nodes aimed at dividing the data set into homogeneous subsets as far as possible. Because entropy expresses the content of information, the smaller the entropy value the more ordered the subset, and a bigger entropy Gini means better homogeneity of the subsets.

Gini impurity is the expected error rate at which a certain result from a set is randomly applied to a data item in the set. It can be calculated as the sum of the product of each selected probability and the probability of this misdivision. If all data on the point belong to a certain target class, then the Gini impurity gets its minimum value of 0.

The classification algorithm of this paper uses random forest with probabilistic output to calculate the probability of each pixel in hyperspectral remote sensing image. The random forest using in this paper with probabilistic output is based on CART (Classification and Regression Tree) decision tree algorithm and the detailed steps of CART is showed at Table 1.

The processes of decision tree classification are as follows:

(1) The establishment of decision tree model; (2) decision tree classification in ENVI; (3) accuracy evaluation in ENVI. The steps are showed as in Figures 6, 7, and 8.

The advantages of CART: the decision tree classification method has the characteristics of clear structure, repeatable operation, high efficiency, flexibility, and intuition and has a good effect in remote sensing image classification. In the decision tree algorithm, there are C5.0 algorithm and classification regression tree CART (classification and regression trees) algorithm, and the classification accuracy of CART decision tree algorithm is better than that of C5.0 algorithm and has the advantages of clear structure and so on. Conventional decision tree rules are generally based on experience and visual interpretation of artificial settings,

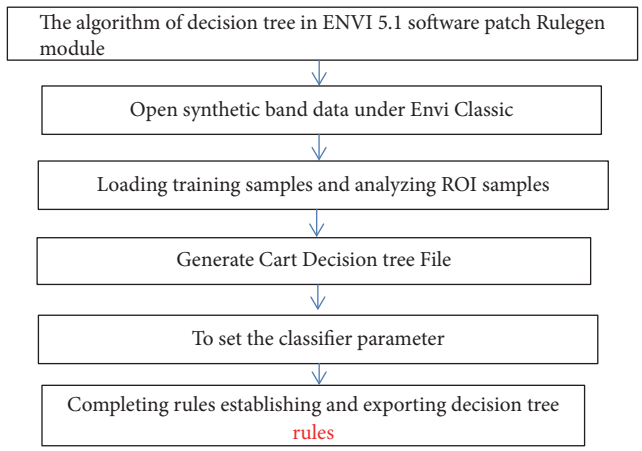



Figure 6: Flowchart of the establishment of decision tree model.

subject to the influence of subjective factors, and classification and regression tree (classification and regression Trees, CART) method can automatically select the classification characteristics and determine the node threshold value [79]. It is the representative of the decision tree model to deal with the nonnumeric data which other algorithms cannot handle.

*2.2.4. Semisupervised Classification of Random Forest Based on Weighted Entropy.* The process of random forest classification involves classifying each randomly generated decision tree classifier, input feature vector, and forest tree to classify the samples. Based on the weight of each tree, the final classification result is obtained. All tree training instances use the same parameters and a different training set; the error estimation of classifiers is based on out of bag. The method of bagging is used to generate different training sets. In other words, bootstrap sampling is used to generate new training sets from the original training set. For each new training set,

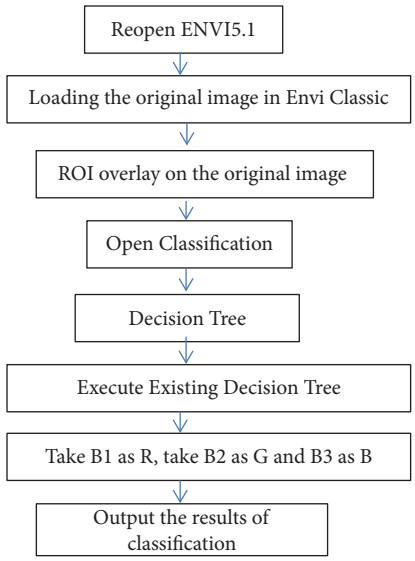



Figure 7: Flowchart of decision tree classification in Envi.

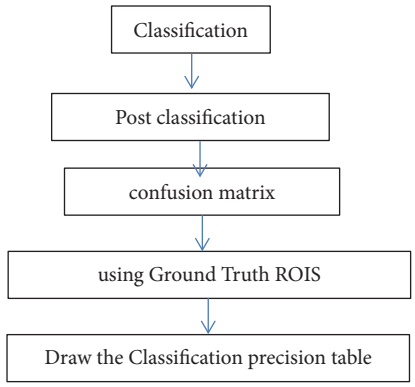



Figure 8: Flowchart of accuracy evaluation.

the random feature selection method is used to generate the decision tree, and the decision tree is not pruned during the growth process.

Random forest is a bagging integrated classifier with CART decision trees as weak classifiers. Boosting is usually used to iteratively call the weak classifier learning algorithm to construct a series of weak classifiers. Subsequently, each round gives greater weight to the failed samples of the last round. Bootstrap aggregating is improved by combining randomly generated training sets to select the training samples independently from the same distribution. There are many algorithms of decision trees for each weak classifier, such as iterative dichotomy (ID3), decision tree category methods (C4.5), and CART.

The core of the ID3 algorithm is the application of the information gain criterion to the decision tree nodes. The method of constructing the decision tree recursively is as follows: start from the root node, calculate the information gain of all possible features of the node and select the maximum characteristic of the information gain as the characteristic of the node, and establish the child nodes based on the different values of the feature. Then, recursively call the above method to the child node and construct the decision tree. Repeat the same steps until all characteristics of the information gain are very small or no characteristics can be selected. Finally, obtain a decision tree. The disadvantages of ID3 are that (1) when we select attributes with information gain, we prefer to select the attribute value with many values, that is, the attribute with more value; and (2) it cannot process contiguous properties.

The C4.5 algorithm is one of the top 10 algorithms in data mining. It is an improvement on the ID3 algorithm, with the following modifications:

(1) it can use the information gain ratio to select attributes;

(2) it can prune trees during the construction of decision trees;

(3) it can be processed for nondiscrete data; and

(4) it can handle incomplete data.

Because random forests are composed of a series of CART (classification and regression tree) decision trees, and vote by the decision tree, the attribute metric used is Gino index [67]. Suppose the information source x is a discrete random variable, and the value of X is $X = \{x_1, x_2, \ldots x_n\}$. If the probability of each message happening is $P = \{p_1, p_2, \ldots p_n\}$,in addition to $\sum_{i=1}^{n} p_i = 1$, Gini index concrete formula is as follows:

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2 \tag{28}$$

The smaller the probability of a category appearing in D means the smaller the Gino index value and the higher the "purity" of the sample. For attributes A in the training sample data set D that will be divided into D1 and D2, the following formula shows the Gini index for the given division of D:

$$Gini_A(D) = \left|\frac{D1}{D}\right| Gini(D1) + \left|\frac{D2}{D}\right| Gini(D2) \tag{29}$$

For discrete value attributes, recursive selection of this attribute produces a subset of the smallest Gini index as its split subset.

For continuous value attributes, all possible split points must be considered, and the decision is similar to the information Gini processing method introduced in ID3; its formula is as follows:

$$Gini_A = \sum_{i=1}^{v} \left|\frac{D_i}{D}\right| \times Gini(D_i) \tag{30}$$

The point where the given continuous attribute value produces the smallest Gini index is chosen as the split point of the attribute, which is the number of nodes [35]. When CART (classification and regression tree) is constructed, each node t is marked with a corresponding class, regardless of whether the nodes in the decision tree are divided or not. Inequality is used as the criterion of classification:

$$\frac{P_C(j \mid i) \cdot P(i) \cdot Ni_t}{P_C(i \mid j) \cdot P(j) \cdot Nj_t} > \frac{Ni}{Nj} \tag{31}$$

If all classes except node I are true, then node t is marked as class, where the a priori probability of class is represented by the a priori probability of class, which is the number of

classes in the sample of node t and the cost of dividing node I into classes. This can be found by looking up the decision tree matrix.

The probability formula x of a result variable of semisupervised classification random forest model with probability output represents a category set; and c solving formula is as shown in the following formula:

$$\widehat{c} = \arg mxp\left(c \mid x\right), \quad c \in \left(1, \cdots, N_C\right) \tag{32}$$

A semisupervised hyperspectral remote sensing image classification method based on weighted entropy is characterized in that the categories corresponding to the probabilistic output of each pixel category in the function (33) are

$$CLA\left(x_i\right) = \operatorname{argmax}\sum_{i=1}^{c} P \tag{33}$$

In the function of (29), the semisupervised random forest hyperspectral remote sensing image classification method based on weighted entropy outputs the classification results and evaluates the accuracy. If the first iteration is suitable for the results, the subsequent steps are carried out; otherwise, the results are compared with the previous output results. If the difference between the two is greater than the given threshold, the subsequent steps are conducted. If the difference is less than the given threshold, the final result is output.

The purpose of this paper is to transform the uncertainty label into a deterministic label that is expressed as an entropy value. In weighted implementation, the greater entropy is given the greater weight. This helps to distinguish the difficult surface categories. It regards the large entropy value as the key distinguishing object that can improve the efficiency of the classification.

In the steps of the semisupervised random forest hyperspectral remote sensing image classification method based on weighted entropy, the weighted entropy algorithm based on voting probability is used to assign different weights to ground objects according to the different needs of researchers [17]. The probability of each pixel in the remote sensing image is transformed into the uncertainty formula by converting the probability into uncertainty, as shown by the following formula:

$$W\left(X\right) = -\sum_{i=1}^{q} w_i p_i \log p_i \tag{34}$$

Suppose the information source x is a discrete random variable and the value of X is $X = \{x_1, x_2, \ldots x_n\}$. If the probability of each message happening is $P = \{p_1, p_2, \ldots p_n\}$, in addition to $\sum_{i=1}^{n} p_i = 1$, the function of the weighted entropy algorithm takes into account the degree of attention to information and the influence of events on people. When we calculate the weighted entropy value of pixel of hyperspectral image data, the maximum value will occur. In order to ensure the accuracy of the experimental results, a normalized treatment is adopted, as shown by the following formula:

$$R\left(X\right) = -p_i \sum_{i=1}^{q} w_i p_i \log p_i \tag{35}$$

We use the OA (the overall accuracy) and kappa to show the function of the classification, which are showed at formulae (36) and (37). OA is the ratio of the number of validation pixels that have been correctly classified to the total number of validation pixels used for all classes and is expressed as a percentage (%). Kappa is the proportion of correctly classified validation points after random agreements are removed and it expresses the extent to which the confusion matrix results are not obtained by chance or random, which is showed at the following formula:

$$OA = \frac{\sum_{i=1}^{m} tp_i}{n} \tag{36}$$

$$Kappa = \frac{n\sum_{i=1}^{m}\left(tp_i\right) - \sum_{i=1}^{m}\left(\left(tp_i + fp_i\right) \times \left(tp_i + fn_i\right)\right)}{n^2 - \sum_{i=1}^{m}\left(\left(tp_i + fp_i\right) \times \left(tp_i + fn_i\right)\right)} \tag{37}$$

In the function, $fp_i$ is the main diagonal element in $i_{th}$ row, which is the ratio of the number of validation pixels that have been correctly classified; $fp_i$ is computed from the sum of $i_{th}$ row column which is the number of real pixels in a class, excluding the main diagonal element; $fn_i$ is the sum along $i_{th}$ row, which is the number of classified pixels in this class, excluding the main diagonal element; m is the number of classes; and n is the total number of pixels in all surface real categories.

*2.3. The Processing of the Algorithm Proposed in the Paper.* The purpose of this improved algorithm is to select the most difficult category and find out the most divisible feature and then classify it and improve the accuracy of classification. The training convergence rate is slow and the performance of each classification varies greatly. The detailed steps of the algorithm are showed at Table 2.

Figure 9 shows the main technical flowcharts of this experiment, which is presented as in Figure 9.

## 3. Results

This experiment is based on the following computer hardware devices:

Inte: (R) Core(TM) i5 3230M CPU @ 2.6Ghz 2.6 Ghz

Install RAM Ram: 4.00GB

System type: 64-bit operating system

The software environment is as follows:

In the environment of Microsoft Windows 7, Envi/IDL5.2 and MATLAB 2017b are used to carry out the experiment.

In this experiment, AVIRIS and Hyperion data were used, respectively.

There will be two parts in this chapter. First of all, the discussion of parameters selection the parameters of random forest will be illustrated. In the second part, classification results and analysis will be showed.

*3.1. The Discussion of Parameters Selection.* In order to ensure the random forest classifier performs well, the number of variables used in a random forest decision tree (N-tree) and

Table 2: The process of semisupervised classification of random forest based on weighted entropy algorithm.

| **Algorithm:** Semi-supervised Classification of Random Forest Based on Weighted Entropy |
|---|
| **Input:** Hyper-spectral remote sensing image data; training sample set; the category set corresponding to the training sample. |
| Output: The results of classification images and confusion matrix; |
| method: |
| (1) Random forest with probability output is used to determine the expected value of the category of ground objects with the largest number of votes; |
| (2) Judging the type of the samples according to the results of output; |
| (3) Judging the accuracy of the output results according to the classified ground object classification data; |
| (4) Then the weighted entropy algorithm is used to give the category with the highest weighted return weight of the class predicted by the model; |
| (5) The ground objects which account for 5% or 10% of the total sample increase are selected and added to the training sample to form a new training sample, and then the prediction classification is carried out again; |
| (6) The unlabeled label pixel in the input hyper-spectral image is converted into the tag pixel according to the uncertainty evaluation value; |
| (7) A new tag is added to the original training set and a new training set is constructed; |
| (8) Run iteratively until termination requirements are met or unlabeled training samples run out. |
| (9) Using the remaining samples to test the performance of the classifier, that is, to evaluate the accuracy of the classifier; |
| (10) In order to test the classification performance and the universality of the classifier, the hyper-spectral remote sensing images of Hyperion and the hyper-spectral remote sensing images of AVIRIS type are classified, and the accuracy is evaluated. |

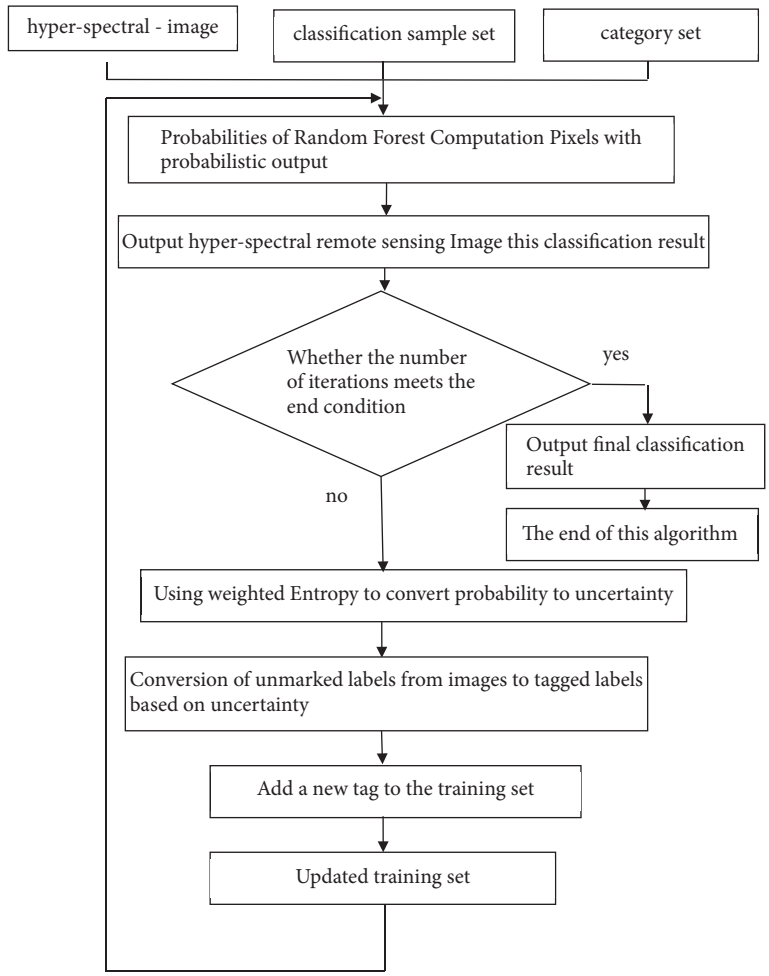



Figure 9: Flowchart of classification framework.

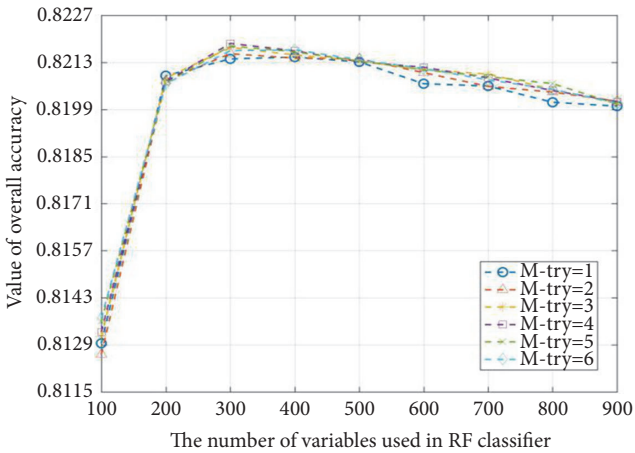



Figure 10: The tendencies of overall accuracy when the N_tree and M_try are setting different values for AVIRIS.

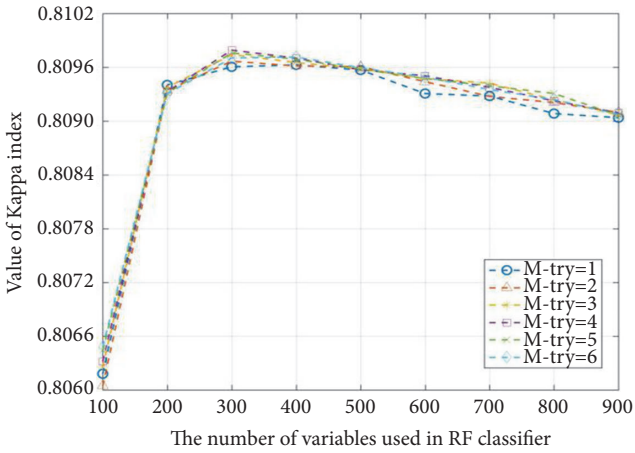



Figure 11: The tendencies of kappa when the N_tree and M_try are setting different values for AVIRIS.

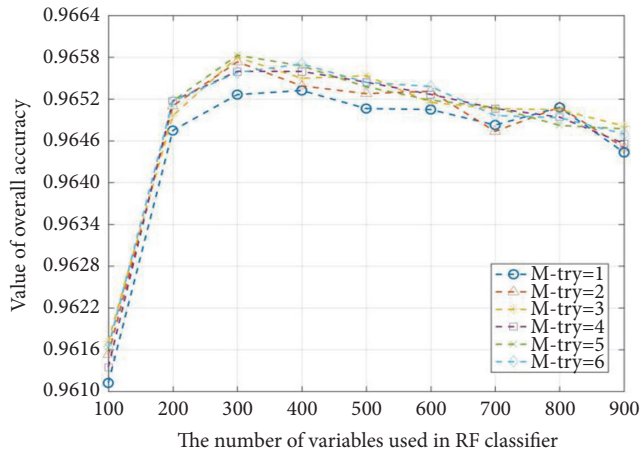



Figure 12: The tendencies of overall accuracy when the N_tree and M_try are setting different values for Hyperion.

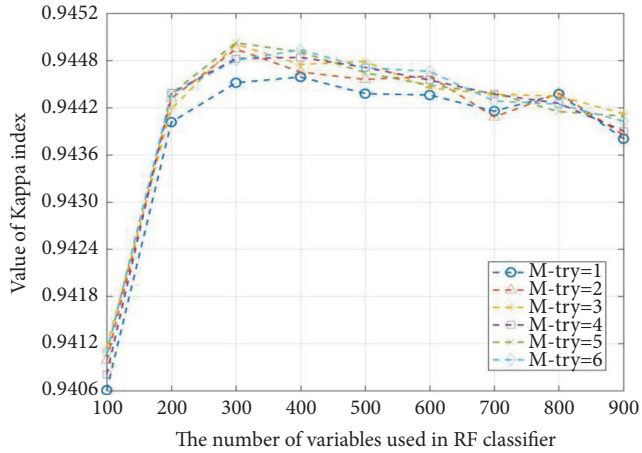



Figure 13: The tendencies of overall accuracy when the N_tree and M_try are setting different values for Hyperion.

the node number (M-try) should be selected. Among them, the characteristic number M-try decides to construct the correlation between the ability of the decision tree and the decision tree; the number of decision trees (N-tree) determines the number of votes and the accuracy of the random forests. Owing to the limitations of real conditions, the N-tree used in random forest changes from 100 to 1000, and M-try changes from 1 to 9. 90 when experiments are performed, aiming at selecting the most appropriate parameters for every data source used in the experiments.

The optimal combination of the number of decision trees and the number of nodes is selected. After testing 90 times, a set of optimal combinational parameters that are suitable for random forests are obtained; these are shown in Figures 10–13. Here, the most fitting parameters for random forest are selected. To be of scientific merit, the evaluation parameters must show the classification function of the classifiers, and their high values show that the fitting degree is good. The figures show that when the combination parameter in that decision tree's selection is 300 and the node number selection is 4, it makes the evaluation parameters of the results of the classification, and it has a high value for AVIRIS and Hyperion data. Therefore, we selected combination parameters where N-tree was 300 and M-try was 4 to carry out the random forest model before starting the weighted entropy algorithm, which ensured the semisupervised classification ran well.

*3.2. Classification Results and Analysis.* Conducting a qualitative and quantitative analysis of the classification results in this part arming at evaluating the accuracy of the classification.

*3.2.1. The Conduct of Qualitative Analysis of the Results of Classification.* While employing the traditional minimum distance and SVM classification method, the AVIRIS and Hyperion data were used to carry out many experiments on 30% of the total samples, 40% of the classification samples, and 50% of the classification samples, which were randomly selected. The mean value of the total classification accuracy,

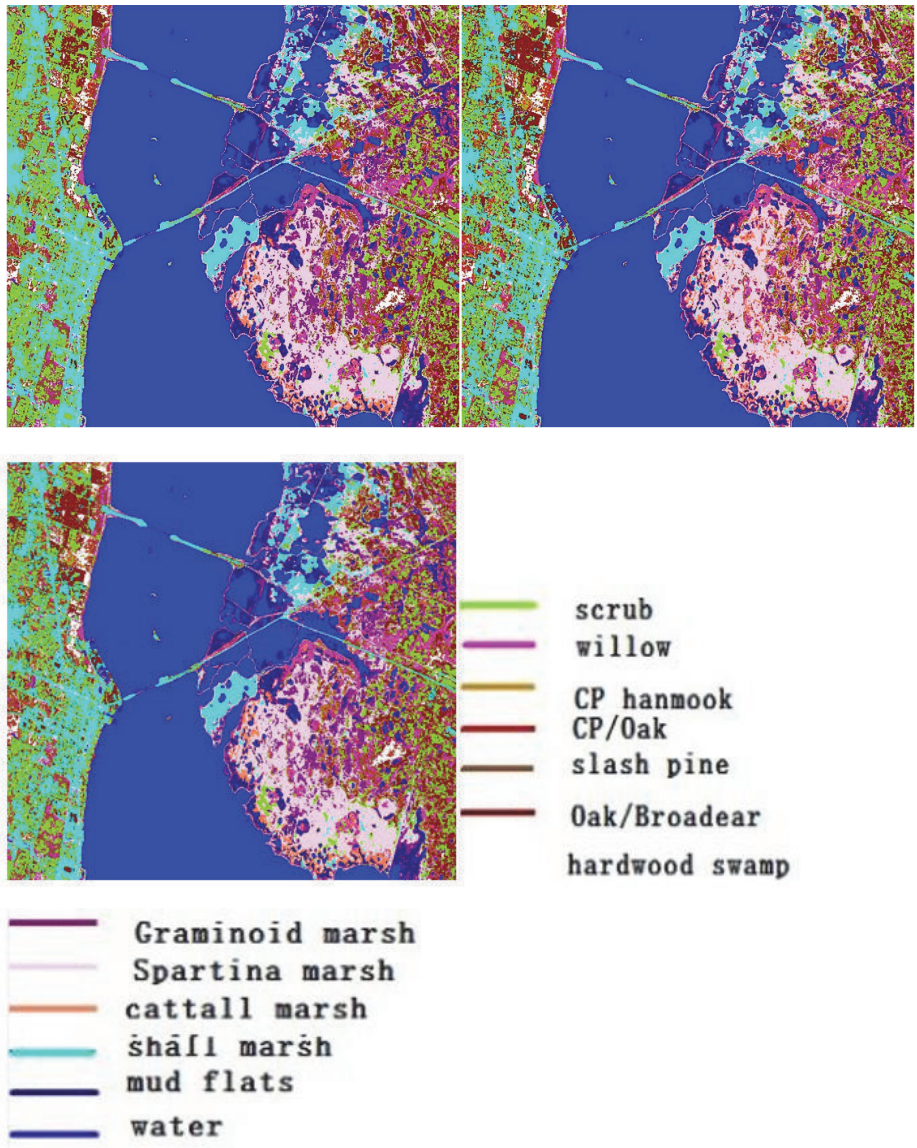



Figure 14: The classification results of minimum distance algorithm for AVIRIS data.

the kappa coefficient, and the corresponding time were calculated so as to eliminate the errors caused by the randomness, and the random training samples corresponding to the minimum distance classifier and the support vector machine classifier were 30%, 40%, and 50%. The classification results are showed from Figures 14–24.

From the map of the classification results, we can see that the classification results of the algorithm proposed in this paper were very consistent with the real surface situation, but showing the classification results more vividly is a problem. In order to show the results more vividly, a qualitative method is proposed, which is illustrated in the next part.

*3.2.2. The Conduct Quantitative Analysis of the Results of Classification.* The performance of the classifier can be demonstrated more intuitively according to the overall classification accuracy, kappa coefficient, and running time. Under a condition where the initial label selects 5% and 10% of the total samples, the total precision of the semisupervised classification method proposed in this chapter is shown in Figures 25–32 for 10 iterations and 5 iterations. The AVIRIS data for method 1 and method 2 are from Figures 25–28. The Hyperion data for method 1 and method 2 are from Figures 29–32.

Figure 25 clearly shows the trends in classification accuracy under different initial conditions and iterative states of the AVIRIS data. It can be seen from the diagram that the number of starting samples was selected as 5% of the total samples; that is, 16 ground objects of each sample were selected for the experiments on this algorithm.

The classification accuracy of the corresponding iteration was 80.02, and then the samples with larger entropy values were added in order from large to small, and 5% of the total number of samples were selected as the new training samples

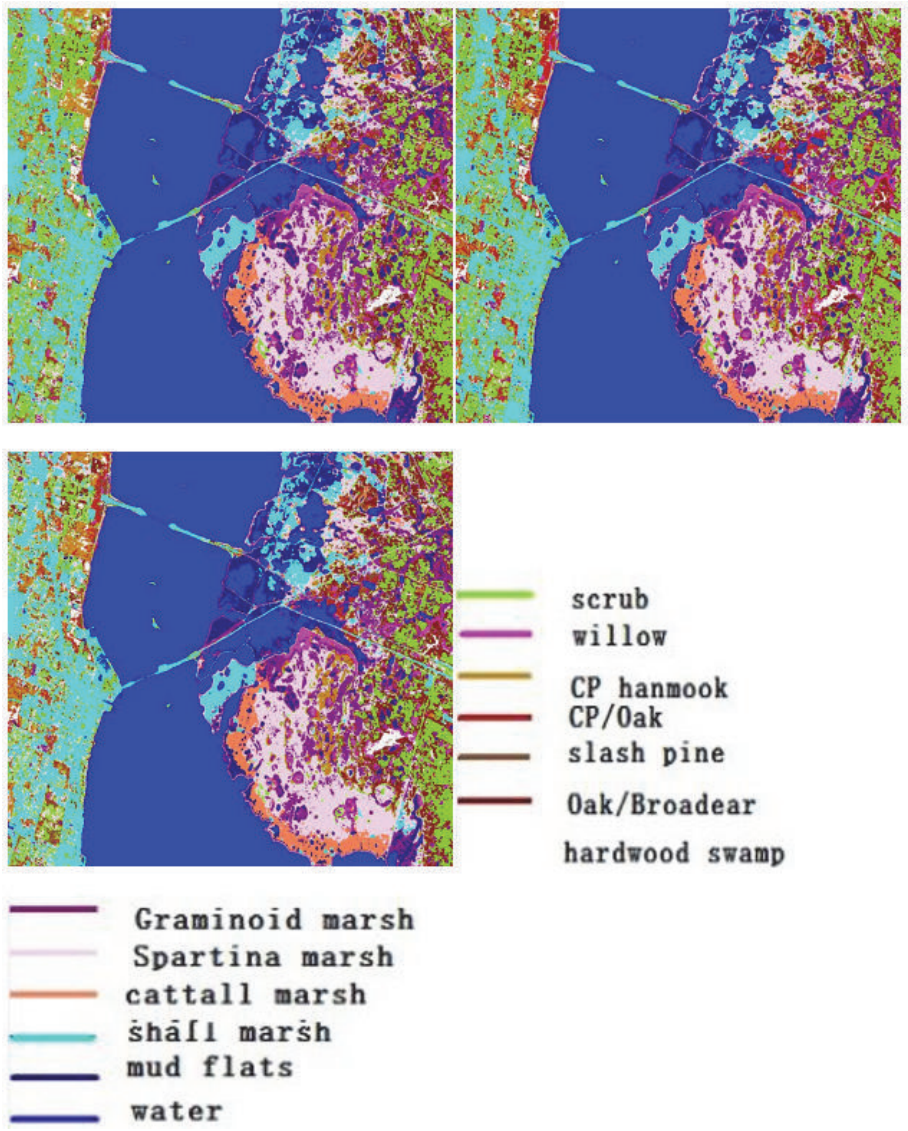



Figure 15: The classification results of support vector machine algorithm for AVIRIS data.

for the second iteration. Using the above rules, we found that, with the increase in the number of samples, the classification accuracy was obviously improved. After the 10th iteration, the kappa reached 0.8542.

The second group of experiments was intended to test the other steps when the initial sample was set at 10% of the total training samples. From Figure 26, we can see that the initial classification accuracy was 83.82, which means the number of training samples increased with the number of iterations. The overall precision increased rapidly and the precision curve tended to be gentle after the fourth iteration; the precision was 87.36 after the fifth iteration. Due to the control of the variables in the two experiments, we found that the classification accuracies of the different initial sample numbers were different: the classification accuracy was higher when the initial sample number was larger.

Compared with the same group of experiments, it was found that the overall accuracy of the classification obviously increased with the increase in iteration time, which indicates that the classification accuracy of the algorithm proposed in this paper has a great correlation with the growth of the samples set.

A comprehensive analysis of the classification results on the AVIRIS data showed that the algorithm proposed in this paper functions relatively well and the initial sample number and the growth of the samples set are closely related to the classification accuracy; also, the initial tag number corresponding to higher classification accuracy is significantly increased with an increase in training samples. What is more, when the samples set is increased to a certain point, the overall accuracy increase is no longer obvious.

Figure 27 clearly shows the trends in the kappa different initial and iterative states of the AVIRIS data. From the

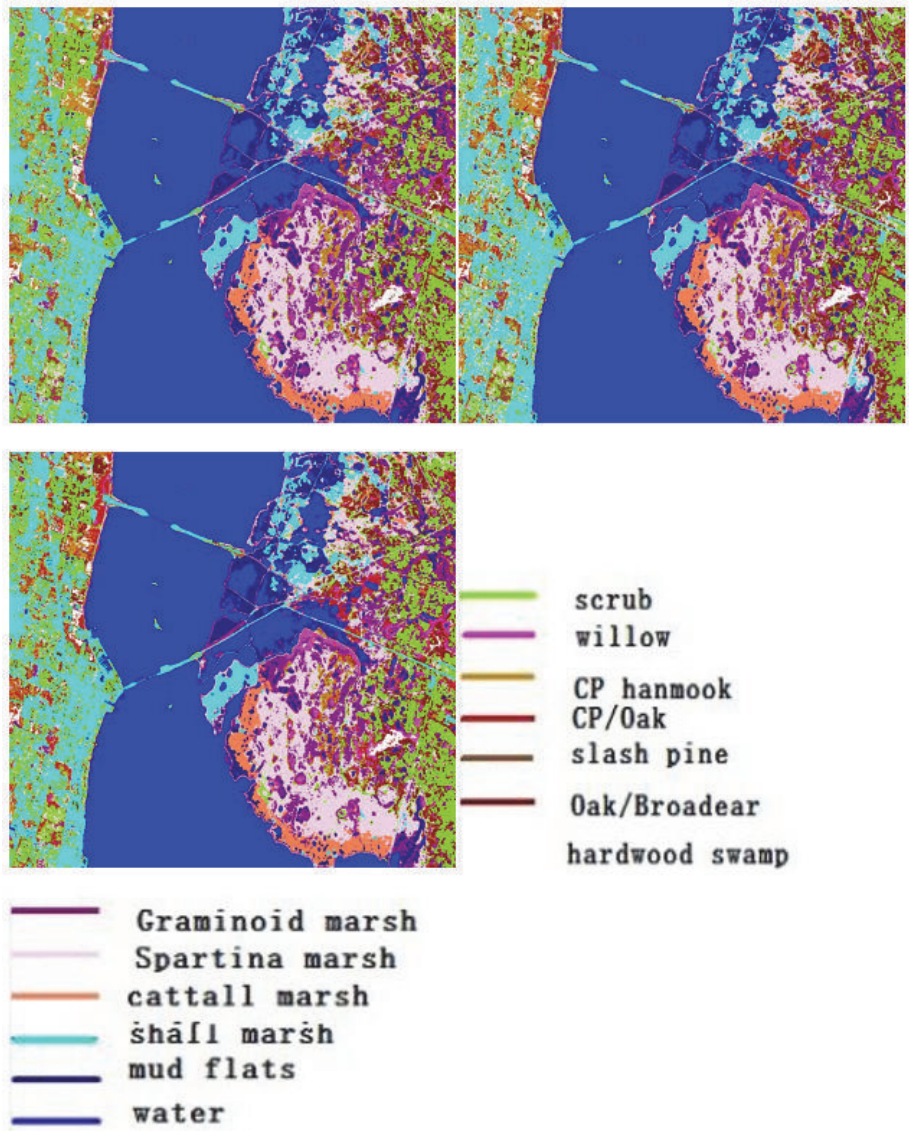



Figure 16: The classification results of CART algorithm for AVIRIS data.

diagram, we can see that the starting samples were 5% of the total samples; that is, 16 ground objects from each sample were selected for the experiment on this algorithm.

The corresponding kappa was 0.7798 at the first iteration, and then we added the larger samples sorted according to entropy value from large to small and selected 5% of the total samples as the new training samples for the second iteration to carry on the experiment. In accordance with the above rule, the kappa coefficient was obviously raised with the increase in the sample number, until the 10th iteration. After the 10th iteration, the kappa reached 0.8371.

The second group of experiments was intended to test the other steps when the initial sample was set as 10% of the total training samples. As shown in Figure 28, the initial kappa was 0.8199 with the increase in the number of iterations. What is more, the kappa coefficient increased rapidly with the increase in the number of training samples. After the fourth iteration, the trend of the kappa curve became stable and the kappa was 0.8591 after the fifth iteration. Also, compared with the experimental group, we found that kappa increased significantly with the increase in the number of iterations, which indicates that the kappa algorithm proposed in this paper is strongly related to the growth of samples set.

A comprehensive analysis showed that the algorithm proposed in this paper on the AVIRIS spectrometer data acquired the classification results and the initial number of samples and the samples set were closely related to the growth of the number of initial labels; also, the corresponding kappa was higher. At the same time, the corresponding kappa significantly increased with the increase in training samples. When the samples set increased to a certain point, the kappa increase was no longer so obvious.

Figure 29 clearly shows the trends in classification accuracy in different initial and iterative states of the Hyperion

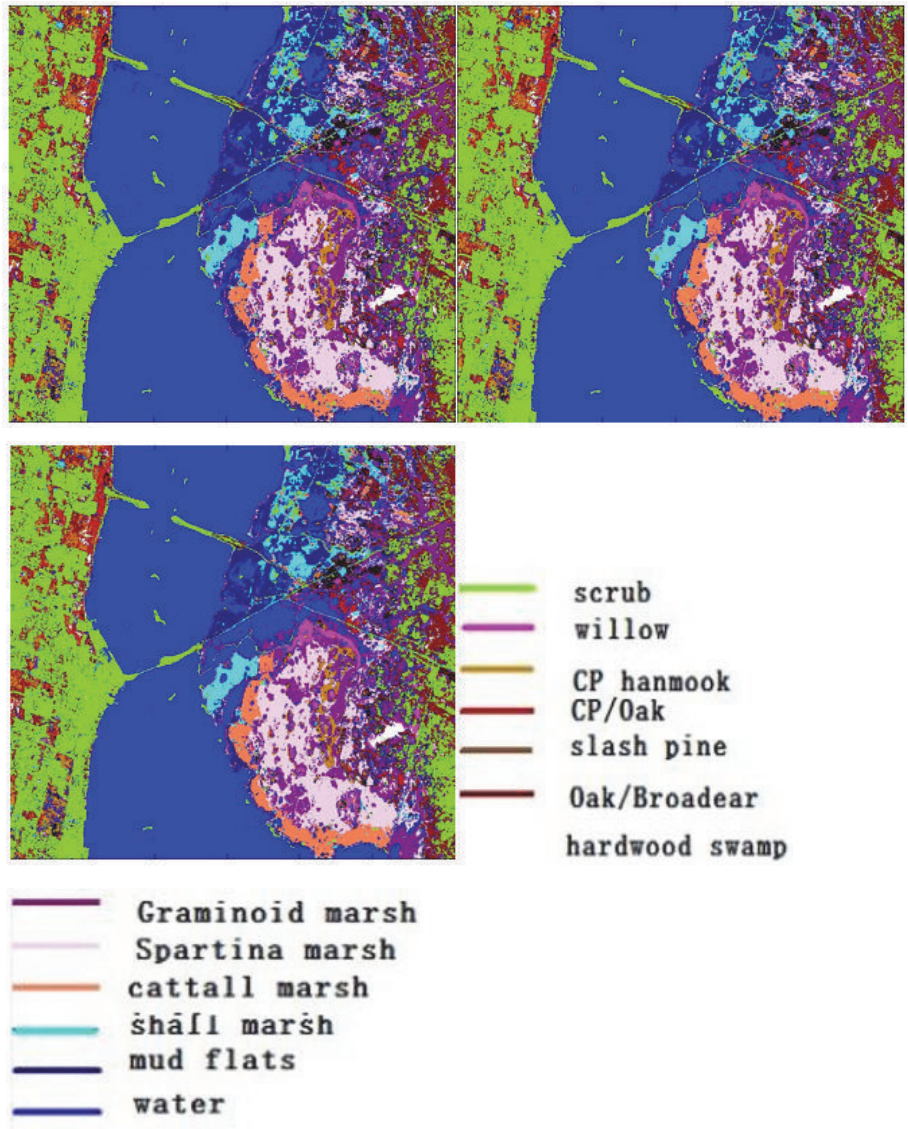



Figure 17: The classification results in the initial 5% label conditions of the algorithm for AVIRIS data.

data. It can be seen from the diagram that the starting sample was set as 5% of the total samples; that is, 16 ground objects from each sample were selected for the experiment on this algorithm. The total accuracy of the corresponding iteration was 0.9352, and then the samples with larger entropy values were added in order from large to small, and 5% of the total samples were selected as the new training samples for the second iteration. Using the above rules, the classification accuracy was obviously improved with the increase in the sample number up to the 10th iteration.

In the second group of experiments, the initial samples were set at 10% of the total. Looking at Figure 30, we can see that the initial classification accuracy was 98.20; that is, the number of training samples increased with the number of iterations. The overall precision increased rapidly, and the precision curve tended to be gentle after the fourth iteration and 99.17 after the fifth iteration. It was found that the classification accuracy differed based on the initial sample number: more initial samples meant more classification precision in the same conditions. The higher the degree of classification means the higher the classification accuracy, and the higher the number of iterations means the higher the classification accuracy, which indicates that the classification accuracy of the proposed algorithm is highly correlated with an increase in the sample set.

A comprehensive analysis showed that the classification effect of the proposed algorithm meant higher classification accuracy. The higher the number of training samples the higher the classification accuracy, and the higher the initial tag number the higher the classification accuracy. However, when the sample set was increased to a certain point, the increase in the total accuracy was no longer obvious.

Figure 31 shows the kappa trends in different initial and iterative states of the Hyperion data. From the diagram, we

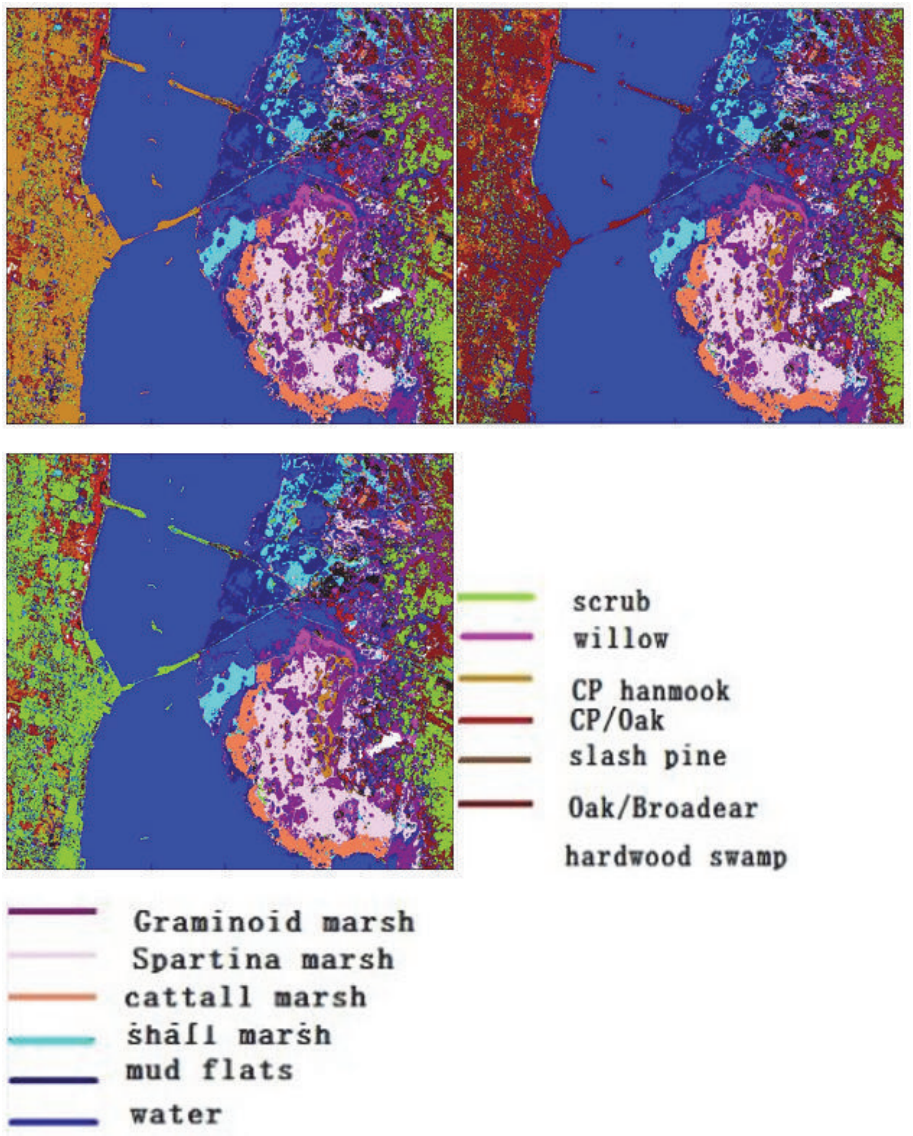



Figure 18: The classification results in the initial 10% label conditions of the algorithm for AVIRIS data.

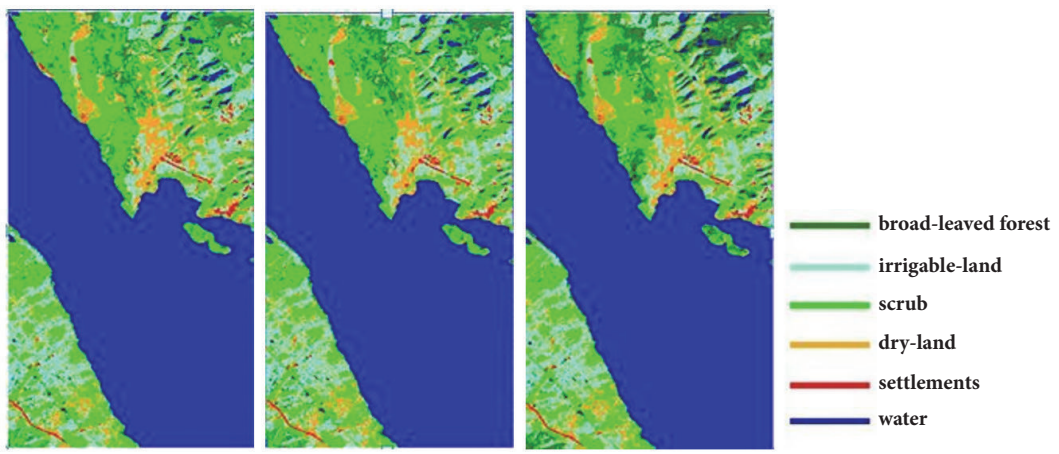



Figure 19: The classification results of minimum distance algorithm for the Hyperion data.

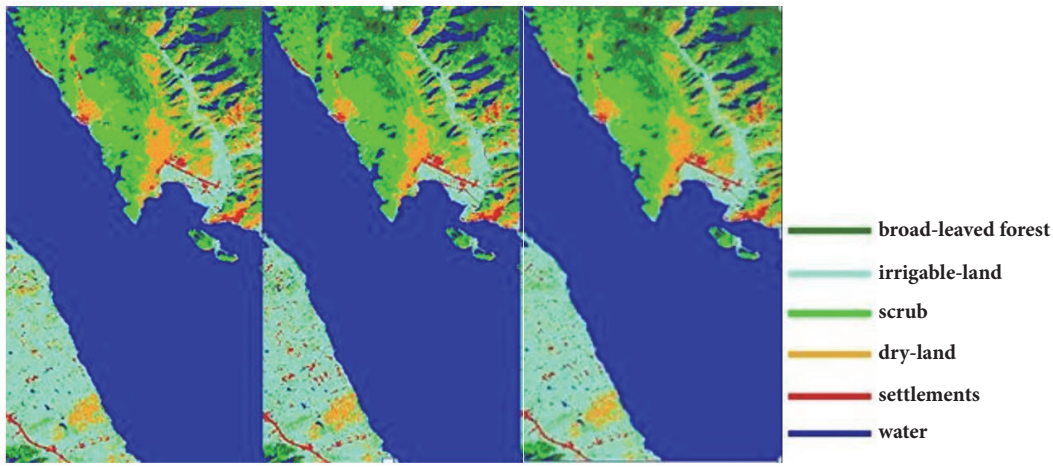



Figure 20: The classification results of support vector machine algorithm for the Hyperion data.

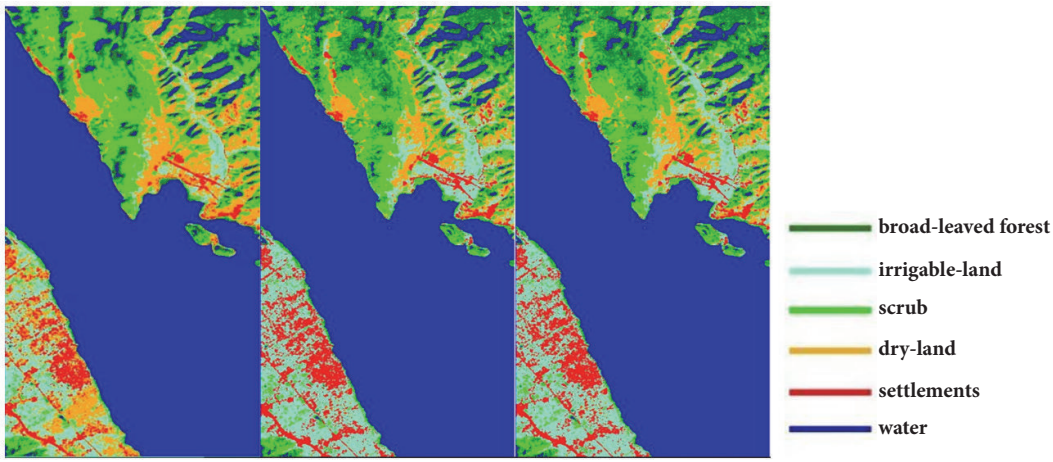



Figure 21: The classification results of support vector machine algorithm for the Hyperion data.

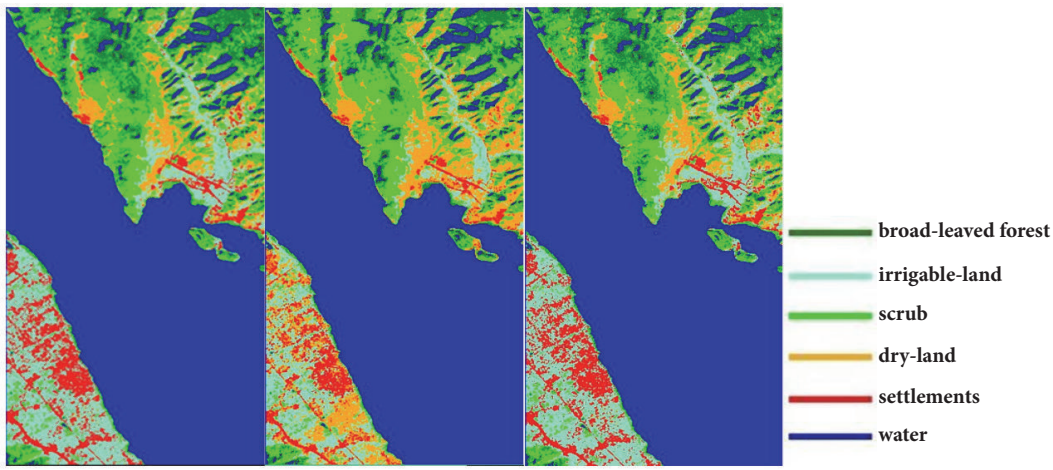



Figure 22: The classification results CART algorithm for the Hyperion data.

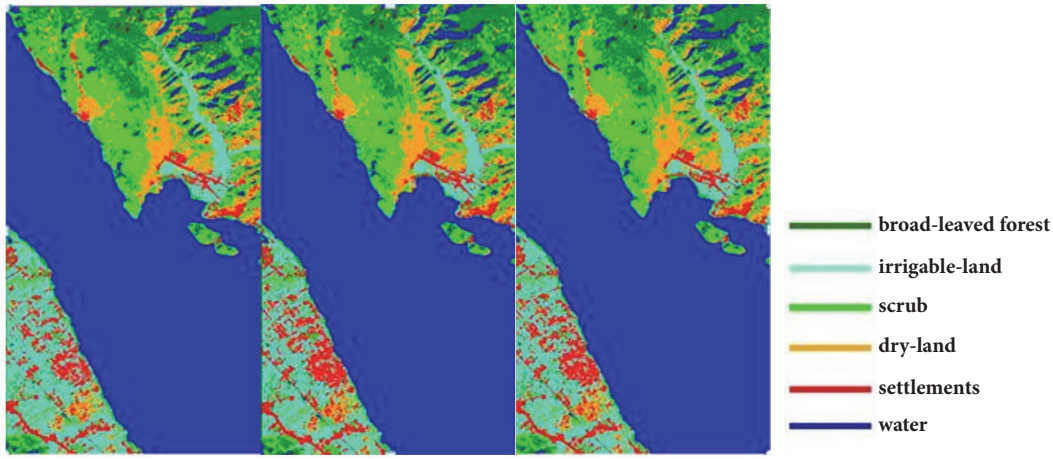



Figure 23: The classification results in the initial 5% label conditions of the algorithm for the Hyperion data.

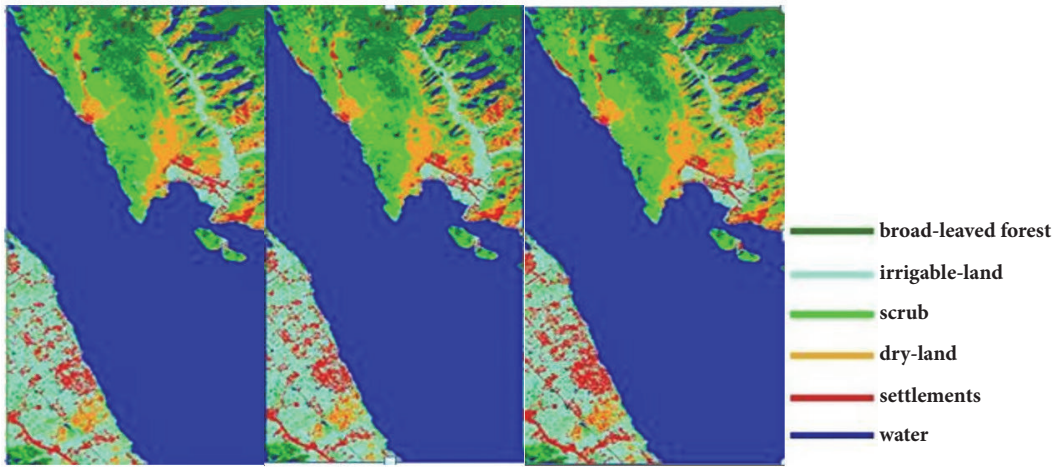



Figure 24: The classification results in the initial 10% label conditions of the algorithm for the Hyperion data.

can see that the starting samples were set at 5% of the total samples; that is, 16 ground objects from each sample were selected for the experiment on this algorithm. The kappa of the corresponding iteration was 0.8812. Then, the samples with larger entropy values were added in order from large to smaller, and 5% of the total samples were selected as the new training samples for the second iteration. Using the above rules, we found that the kappa coefficient increased with the number of samples until the 10th iteration. After the 10th iteration, the kappa reached 0.9888.

In the second group of experiments, the number of initial samples was set at 10% of the training samples. As shown in Figure 32, the initial kappa was 0.9255, and as the number of iterations increased, the final kappa reached 0.9901. The kappa coefficient increased rapidly when the number of training samples increased. By the fourth iteration, the kappa curve tended to be smooth, and its value was 0.9904. When the number of initial samples was higher, the kappa was different. Compared with the same group of experiments, it was found that the number of iterations obviously increased, which indicates that the kappa of the proposed algorithm has a greater correlation with the increase in the samples set.

Our analysis showed that the classification effect of the proposed algorithm for the Hyperion data was closely related to the initial sample number and the growth of the samples set. To some degree, the higher the initial tag numbers the higher the corresponding kappa. However, once the sample set was increased to a certain point, the increase in kappa was no longer so obvious.

The classification performance of the minimum distance classification algorithm, the SVM classifier, and the semisupervised classifier based on weighted entropy and stochastic random forest integration can be classified using the same validation data. The results on the overall classification accuracy, the kappa coefficient, and the running time are shown in Tables 3 and 4.

The tables show that when they are in the same conditions, the semisupervised classifier based on weighted entropy and the stochastic random forest integration proposed in the paper function well, which improves the overall

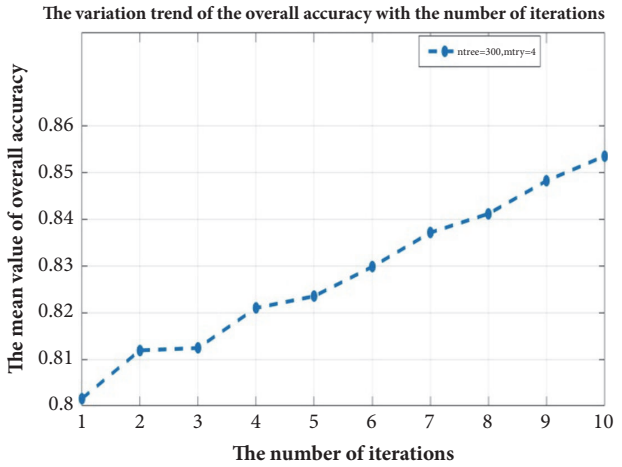



Figure 25: Classification accuracy trend of different initial and iterative condition of AVIRIS data for method 1.

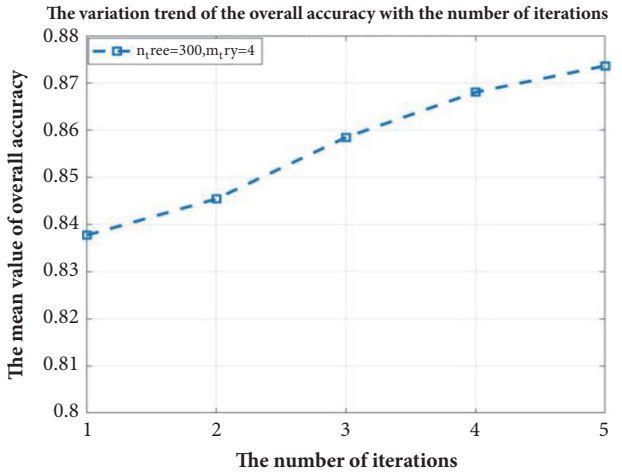



Figure 26: Classification accuracy trend of different initial and iterative condition of AVIRIS data for method 2.

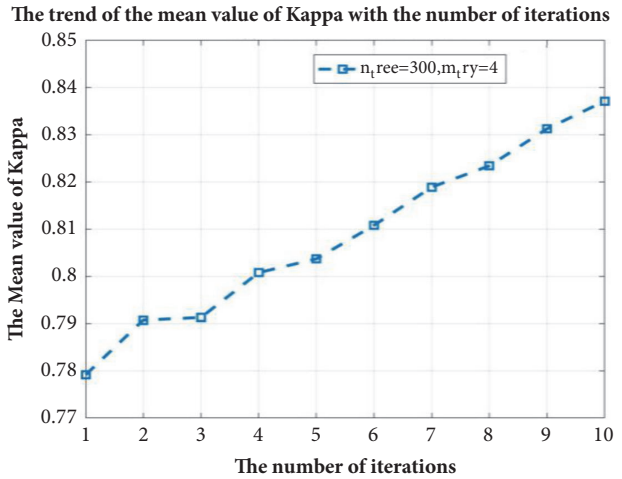



Figure 27: Classification kappa trend of different initial and iterative condition of AVIRIS data for method 1.

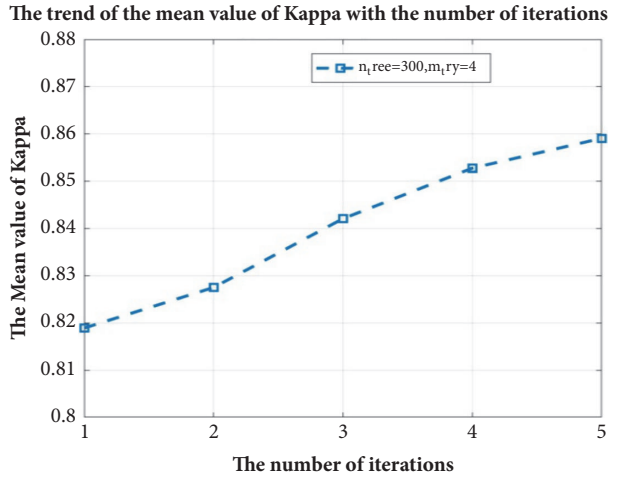



Figure 28: Classification kappa trend of different initial and iterative condition of AVIRIS data for method 2.

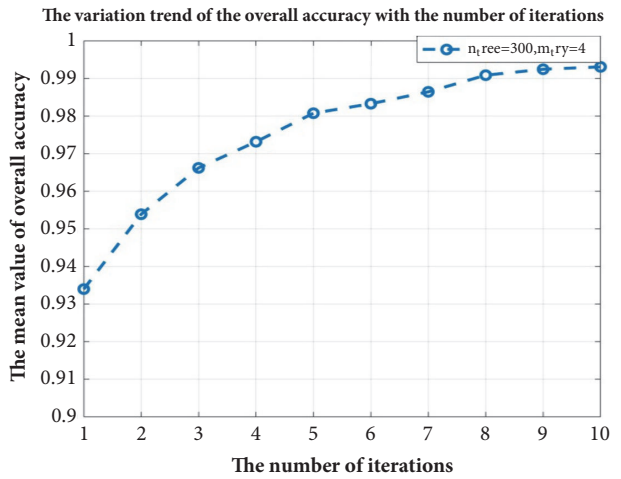



Figure 29: Classification accuracy trend of different initial and iterative condition of Hyperion data for method 1.

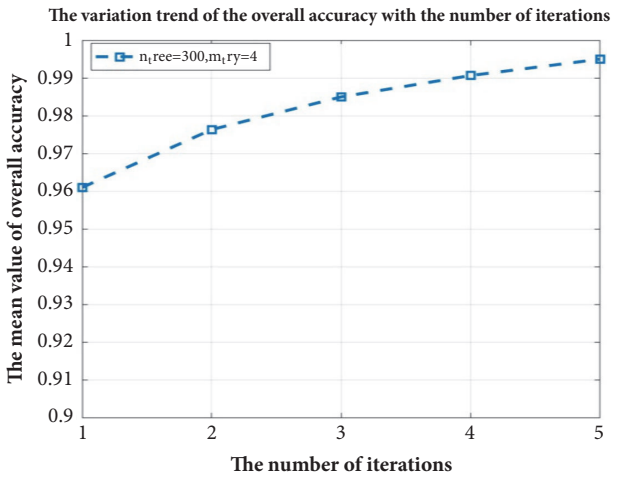



Figure 30: Classification accuracy trend of different initial and iterative condition of Hyperion data for method 2.

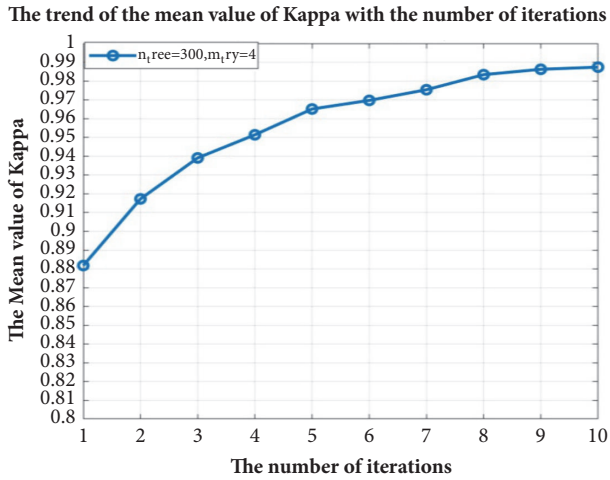



Figure 31: Classification kappa trend of different initial and iterative condition of Hyperion data for method 1.

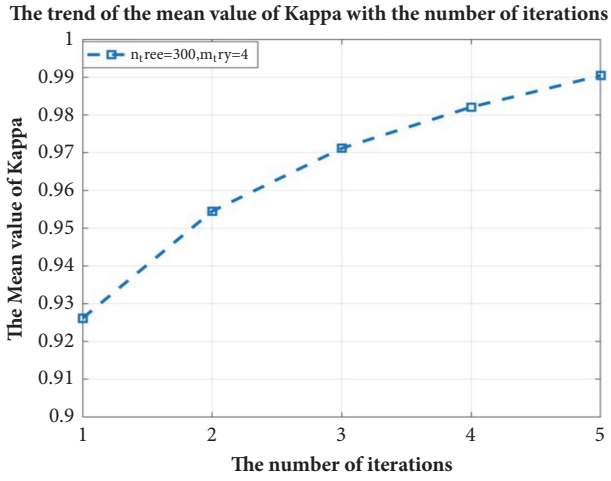



Figure 32: Classification kappa trend of different initial and iterative condition of Hyperion data for method 2.

classification accuracy and the kappa coefficient successfully; furthermore, when number of the labeled labels is 5% of the total, the overall accuracy of the AVIRIS data increased to 85.35%, which is about 20% higher than the minimum distance classification algorithm and 3% higher than that of the SVM classification algorithm, what is more, 7.02% higher than that of CART algorithm. The kappa coefficient of the AVIRIS data increased to 0.8591, which is about 0.22 higher than the minimum distance classification algorithm and 0.25 higher than that of the SVM classification algorithm, what is more, 0.08 higher than that of CART algorithm; when number of the labeled labels is 10% of the total, the overall accuracy of the AVIRIS data increased to 87.36%, which is about 22.15% higher than the minimum distance classification algorithm and 5.01% higher than that of the SVM classification algorithm, what is more, 9.03% higher than that of CART algorithm. The kappa coefficient of the AVIRIS data increased to 0.8591, which is about 0.2454 higher than the minimum distance classification algorithm and 0.055 higher than that of the SVM classification algorithm, what is more, 0.1 higher than that of CART algorithm.

When number of the labeled labels is 5% of the total, the overall accuracy of the Hyperion data increased to 98.83%, which is about 7.88% higher than the minimum distance classification algorithm and 3.92% higher than that of SVM classification algorithm, what is more, 4.71% higher than that of CART algorithm. The kappa coefficient of the Hyperion data increased to 0.9788, which is about 0.1836 higher than the minimum distance classification algorithm and 0.0959 higher than that of SVM classification algorithm, what is more, 0.0887 higher than that of CART algorithm; when number of the labeled labels is 10% of the total, the overall accuracy of the Hyperion data is up to 99.17%, which is about 8.22% higher than the minimum distance classification algorithm and 4.26% higher than that of SVM classification algorithm, what is more, 5.06% higher than that of CART algorithm. The kappa coefficient of the Hyperion data increased to 0.9904, which is about 0.1952 higher than the minimum distance classification algorithm and 0.1075 higher than that of SVM classification algorithm, what is more, 0.1003 higher than that of CART algorithm.

To sum up, the algorithm proposed in this paper can effectively improve the effect of classification. It is very convenient and fast compared with the minimum distance classification algorithm, the SVM, and regression trees (CART) classification algorithm in the same conditions.

## 4. Conclusions

After experimenting with two different data sources using the proposed method, the following conclusions can be drawn: through a large number of experiments, a set of optimal combination parameters suitable for random forest was obtained. That is, we set the number of decision trees at 300 and the number of nodes at 4. When the random forest parameters were optimal, samples of 5% and 10% of the total were selected for the experiment, and the samples of 5% and 10% were added each time. The weighted entropy algorithm was used to select samples with the largest entropy values to train the new training set proposed in this paper. The classifier used the remaining data as the test data to evaluate the performance of the classifier and to test the universality of the classifier. Compared with the traditional classifier based on supervised classification and SVM, we proved via a large number of experiments that the proposed weighted entropy semisupervised ensemble classifier based on random forest showed better classification performance and better universality.

However, the new algorithm still has inadequacies, such as its long running time and need for more computing power and hardware. Our future research will focus on optimizing the algorithm, lowering its running time, and improving its efficiency.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Table 3: The comparison of overall accuracy and Kappa coefficient under different classifier algorithm of AVIRIS data.

| Comparative classification method | Run time (seconds) | | | Overall Accuracy (%) | | | Kappa Coefficient | | |
|---|---|---|---|---|---|---|---|---|---|
| | T1 | T2 | T3 | T1 | T2 | T3 | T1 | T2 | T3 |
| Minimum distance classification algorithm | 7 | 10 | 11 | 65.46 | 63.06 | 65.21 | 0.6164 | 0.5895 | 0.6137 |
| Support vector machine classification algorithm | 62 | 72 | 86 | 78.47 | 80.17 | 82.35 | 0.7609 | 0.7704 | 0.8041 |
| Classification and regression trees (CART) | 17 | 21 | 26 | 75.04 | 78.0412 | 78.3285 | 0.7227 | 0.7557 | 0.7591 |
| This-algorithm classifies( 5% of the-labeled labels) | 16.21 | 16.63 | 17.69 | 82.98 | 84.19 | 85.35 | 0.8108 | 0.8234 | 0.8371 |
| This-algorithm classifies(10% labeled labels) | 20.65 | 21.10 | 22.72 | 85.85 | 86.81 | 87.36 | 0.8421 | 0.8528 | 0.8591 |

Table 4: The comparison of overall accuracy and Kappa coefficient under different classifier algorithm of Hyperion data.

| Comparative classification method | Run time (seconds) | | | Overall Accuracy (%) | | | Kappa Coefficient | | |
|---|---|---|---|---|---|---|---|---|---|
| | T1 | T2 | T3 | T1 | T2 | T3 | T1 | T2 | T3 |
| Minimum distance classification algorithm | 12 | 15 | 14 | 89.26 | 89.83 | 90.95 | 0.7567 | 0.7693 | 0.7952 |
| Support-vector machine classification algorithm | 25 | 27 | 26 | 94.55 | 94.63 | 94.91 | 0.8744 | 0.8752 | 0.8829 |
| Classification and regression trees (CART) | 7.5 | 7.2 | 6.7 | 86.0806 | 93.3795 | 94.1106 | 0.6806 | 0.8468 | 0.8901 |
| This-algorithm classifies (5% of the labeled labels) | 6.04 | 6.35 | 6.97 | 97.22 | 98.26 | 98.83 | 0.9495 | 0.9684 | 0.9788 |
| This-algorithm classifies(10% of labeled labels) | 7.16 | 7.69 | 8.16 | 98.20 | 98.90 | 99.17 | 0.9711 | 0.9820 | 0.9904 |

## Conflicts of Interest

The authors declare no conflicts of interest.

## Acknowledgments

This research is supported by the Natural Science Foundation of Henan Province (182300410111), the Key Research Project Fund of Institution of Higher Education in Henan Province (18A420001), Henan Polytechnic University Doctoral Fund (B2016-13), and the Open Program of Collaborative Innovation Center of Geo-Information Technology for Smart Central Plains Henan Province (2016A002).

## References

[1] Q. Tong, B. Zhang, L. Zheng et al., *Hyperspectral Remote Sensing-Principle, Technology and Application*, Higher Education Press, Beijing, China, 2006.

[2] A. F. H. Goetz, "Three decades of hyperspectral remote sensing of the Earth: a personal view," *Remote Sensing of Environment*, vol. 113, supplement 1, pp. S5–S16, 2009.

[3] Q. Tong, Y. Xue, and L. Zhang, "Progress in hyperspectral remote sensing science and technology in China over the past three decades," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 1, pp. 70–91, 2014.

[4] G. H. Mitri and I. Z. Gitas, "Mapping post-fire forest regeneration and vegetation recovery using a combination of very high spatial resolution and hyperspectral satellite imagery," *International Journal of Applied Earth Observation and Geoinformation*, vol. 20, no. 1, pp. 60–66, 2012.

[5] W. Shuyu, Z. Yuwei, and Y. Zhenhua, "Classification of remote sensing images of honghe wetland based on stochastic forest," *Mapping and Spatial Geography*, pp. 83–85, 2014 (Chinese).

[6] G. P. Petropoulos, C. Kalaitzidis, and K. Prasad Vadrevu, "Support vector machines and object-based classification for obtaining land-use/cover cartography from Hyperion hyperspectral imagery," *Computers & Geosciences*, vol. 41, pp. 99–107, 2012.

[7] R. Piiroinen, J. Heiskanen, M. Mõttus, and P. Pellikka, "Classification of crops across heterogeneous agricultural landscape in Kenya using AisaEAGLE imaging spectroscopy data," *International Journal of Applied Earth Observation and Geoinformation*, vol. 39, pp. 1–8, 2015.

[8] R. R. Nidamanuri and B. Zbell, "Transferring spectral libraries of canopy reflectance for crop classification using hyperspectral remote sensing," *Biosystems Engineering*, vol. 110, no. 3, pp. 231–246, 2011.

[9] R. Casa, F. Castaldi, S. Pascucci, A. Palombo, and S. Pignatti, "A comparison of sensor resolution and calibration strategies for soil texture estimation from hyperspectral remote sensing," *Geoderma*, vol. 197-198, pp. 17–26, 2013.

[10] I. Mariotto, P. S. Thenkabail, A. Huete, E. T. Slonecker, and A. Platonov, "Hyperspectral versus multispectral crop-productivity modeling and type discrimination for the HyspIRI mission," *Remote Sensing of Environment*, vol. 139, pp. 291–305, 2013.

[11] K. Zhao, D. Valle, S. Popescu, X. Zhang, and B. Mallick, "Hyperspectral remote sensing of plant biochemistry using Bayesian model averaging with variable and band selection," *Remote Sensing of Environment*, vol. 132, pp. 102–119, 2013.

[12] C. Wang, Z. Xu, S. Wang, and H. Zhang, "Semi-supervised classification framework of hyperspectral images based on the fusion evidence entropy," *Multimedia Tools and Applications*, vol. 77, no. 9, pp. 10615–10633, 2018.

[13] A. Harris, R. Charnock, and R. M. Lucas, "Hyperspectral remote sensing of peatland floristic gradients," *Remote Sensing of Environment*, vol. 162, pp. 99–111, 2015.

[14] R. J. Murphy and S. T. Monteiro, "Mapping the distribution of ferric iron minerals on a vertical mine face using derivative analysis of hyperspectral imagery (430-970nm)," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 75, pp. 29–39, 2013.

[15] K. Tan, R. Zhou, Y. Wan et al., "High-spectral and high-resolution remote sensing monitoring method for underground coal seam combustion," *Journal of Infrared and Millimeter Wave*, vol. 26, pp. 349–358, 2007.

[16] K. Liu, X. Sun, Z. Zhao et al., "Hyperspectral imaging detection method for ground target camouflage features," *Journal of PLA University of Technology (Natural Science Edition)*, vol. 6, pp. 166–169, 2005.

[17] W. Lu, X. Yu, Y. Ma, and J. Liu, "Research on detection algorithm of sea warship target by hyperspectral remote sensing image," *Ocean Mapping*, vol. 4, pp. 8–12, 2005.

[18] Y. Chu, W. Feng et al., *Analysis and Application of Hyperspectral Imagery*, Science Press, Beijing, China, 2013.

[19] K. Tan, "Research on hyperspectral remote Sensing image classification based on support vector machine," *Mining*, 2010.

[20] B. Du, L. Zhang, L. Zhang, and W. Hu, "A method for discriminating manifold learning by dimensionality reduction in hyperspectral images," *Photonics newspaper*, vol. 03, pp. 320–325, 2013.

[21] A. Ghosh, A. Datta, and S. Ghosh, "Self-adaptive differential evolution for feature selection in hyperspectral image data," *Applied Soft Computing*, vol. 13, no. 4, pp. 1969–1977, 2013.

[22] D. Letexier and S. Bourennane, "Noise removal from hyperspectral images by multidimensional filtering," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, no. 7, pp. 2061–2069, 2008.

[23] B. Bigdeli, F. Samadzadegan, and P. Reinartz, "Fusion of hyperspectral and LIDAR data using decision template-based fuzzy multiple classifier system," *International Journal of Applied Earth Observation and Geoinformation*, vol. 38, pp. 309–320, 2015.

[24] X. Tang, K. Gao, H. Cheng, and G. Ni, "Hyperspectral unmixing based on ISOMAP and spatial information," *Optik - International Journal for Light and Electron Optics*, vol. 125, no. 16, pp. 4283–4287, 2014.

[25] Z. Bing, "Frontier of hyper-spectral image processing and information extraction," *Journal of remote Sensing*, pp. 1062–1090, 2016.

[26] F. D. van der Meer, H. M. A. van der Werff, F. J. A. van Ruitenbeek et al., "Multi- and hyper-spectral geologic remote sensing: A review," *International Journal of Applied Earth Observation and Geoinformation*, vol. 14, no. 1, pp. 112–128, 2012.

[27] Z. Liangpei and L. Jiaye, "Summarization and prospect of sparse information processing of hyper-spectral images," *Acta Sinica remote Sensing*, pp. 1091–1101, 2016.

[28] P. Dubath, L. Rimoldini, M. Süveges et al., "Random forest automated supervised classification of Hipparcos periodic variable stars," *Monthly Notices of the Royal Astronomical Society*, vol. 414, no. 3, pp. 2602–2617, 2011.

[29] X. Fan, I. Riaz, Y. Rehman, and H. Shin, "Vanishing point detection using random forest and patch-wise weighted soft voting," *IET Image Processing*, vol. 10, no. 11, pp. 900–907, 2016.

[30] C. González, J. Mira-McWilliams, and I. Juárez, "Important variable assessment and electricity price forecasting based on regression tree models: Classification and regression trees, Bagging and Random Forests," *IET Generation, Transmission & Distribution*, vol. 9, no. 11, pp. 1120–1128, 2015.

[31] Y. Liu, P. Du, H. Zheng et al., "Research on classification of domestic small satellite remote sensing image based on random forest science of surveying and mapping," *Science of Surveying and Mapping*, no. 04, pp. 194–196, 2012.

[32] D.-J. Yu, Y. Li, J. Hu, X. Yang, J.-Y. Yang, and H.-B. Shen, "Disulfide connectivity prediction based on modelled protein 3D structural information and random forest regression," *IEEE Transactions on Computational Biology and Bioinformatics*, vol. 12, no. 3, pp. 611–621, 2015.

[33] L. Shilei, *Research on stochastic forest algorithm based on Hadoop platform and realization of image classification system*, Xiamen University, 2014.

[34] M. Zhu, J. Xia, X. Jin et al., "Class weights random forest algorithm for processing class imbalanced medical data," *IEEE Access*, vol. 6, pp. 4641–4652, 2018.

[35] X. Han-Qiu, "Study on extracting water body information by using improved normalized difference water body index (MNDWI)," *Acta Sinica Sinica*, pp. 589–595, 2005.

[36] K. Tan, J. Hu, J. Li, and P. Du, "A novel semi-supervised hyperspectral image classification approach based on spatial neighborhood information and classifier combination," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 105, pp. 19–29, 2015.

[37] C. Yi, Z. Xiufang, S. Zhangli et al., "Overview of semi-supervised integrated learning," *Computer Science*, vol. 1, pp. 7–13, 2017.

[38] M. Belgiu and L. Drăgu, "Random forest in remote sensing: A review of applications and future directions," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 114, pp. 24–31, 2016.

[39] K. Were, D. T. Bui, Ø. B. Dick, and B. R. Singh, "A comparative assessment of support vector regression, artificial neural networks, and random forests for predicting and mapping soil organic carbon stocks across an Afromontane landscape," *Ecological Indicators*, vol. 52, pp. 394–403, 2015.

[40] R.-M. Yang, G.-L. Zhang, F. Liu et al., "Comparison of boosted regression tree and random forest models for mapping topsoil organic carbon concentration in an alpine ecosystem," *Ecological Indicators*, vol. 60, pp. 870–878, 2016.

[41] I. Nitze, B. Barrett, and F. Cawkwell, "Temporal optimisation of image acquisition for land cover classification with random forest and MODIS time-series," *International Journal of Applied Earth Observation and Geoinformation*, vol. 34, no. 1, pp. 136–146, 2015.

[42] V. F. Rodriguez-Galiano, B. Ghimire, J. Rogan, M. Chica-Olmo, and J. P. Rigol-Sanchez, "An assessment of the effectiveness of a random forest classifier for land-cover classification," *ISPRS*

*Journal of Photogrammetry and Remote Sensing*, vol. 67, no. 1, pp. 93–104, 2012.

[43] L. Li, C. Solana, F. Canters, and M. Kervyn, "Testing random forest classification for identifying lava flows and mapping age groups on a single Landsat 8 image," *Journal of Volcanology and Geothermal Research*, vol. 345, pp. 109–124, 2017.

[44] J. C.-W. Chan and D. Paelinckx, "Evaluation of Random Forest and Adaboost tree-based ensemble classification and spectral band selection for ecotope mapping using airborne hyperspectral imagery," *Remote Sensing of Environment*, vol. 112, no. 6, pp. 2999–3011, 2008.

[45] V. F. Rodriguez-Galiano, M. Chica-Olmo, F. Abarca-Hernandez, P. M. Atkinson, and C. Jeganathan, "Random Forest classification of Mediterranean land cover using multi-seasonal imagery and multi-seasonal texture," *Remote Sensing of Environment*, vol. 121, pp. 93–107, 2012.

[46] C. Pelletier, S. Valero, J. Inglada, N. Champion, and G. Dedieu, "Assessing the robustness of Random Forests to map land cover with high resolution satellite image time series over large areas," *Remote Sensing of Environment*, vol. 187, pp. 156–168, 2016.

[47] L. Zg, *Some studies on stochastic forest improvement*, Xiamen University, 2013.

[48] M. Śmieja, "Weighted approach to general entropy function," *IMA Journal of Mathematical Control and Information*, vol. 32, no. 2, pp. 329–341, 2015.

[49] J. Wang, W. Pang, L. Wang et al., "Synthetic evaluation of steady-state power quality based on combination weighting and principal component projection method," *CSEE Journal of Power and Energy Systems*, vol. 3, no. 2, pp. 160–166, 2017.

[50] X. Qu, H. Chen, and G. Peng, "Novel detection method for infrared small targets using weighted information entropy," *Journal of Systems Engineering and Electronics*, vol. 23, no. 6, pp. 838–842, 2012.

[51] S. Gholinejad, R. Shad, H. S. Yazdi, and M. Ghaemi, "Improving Signal Subspace Identification Using Weighted Graph Structure of Data," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 6, pp. 831–835, 2017.

[52] Y. Xu, Y. Wang, and X. Miu, "Multi-attribute decision making method for air target threat evaluation based on intuitionistic fuzzy sets," *Journal of Systems Engineering and Electronics*, vol. 23, no. 6, pp. 891–897, 2012.

[53] C. Lindner, P. A. Bromiley, M. C. Ionita, and T. F. Cootes, "Robust and Accurate Shape Model Matching Using Random Forest Regression-Voting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1862–1874, 2015.

[54] N. Segev, M. Harel, S. Mannor, K. Crammer, and R. El-Yaniv, "Learn on Source, Refine on Target: A Model Transfer Learning Framework with Random Forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 9, pp. 1811–1824, 2017.

[55] J. Chen, K. Li, Z. Tang et al., "A Parallel Random Forest Algorithm for Big Data in a Spark Cloud Computing Environment," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 4, pp. 919–933, 2017.

[56] F. Miao, Y.-P. Cai, Y.-X. Zhang, X.-M. Fan, and Y. Li, "Predictive modeling of hospital mortality for patients with heart failure by using an improved random survival forest," *IEEE Access*, vol. 6, pp. 7244–7253, 2018.

[57] X. Wang, "Ladle furnace temperature prediction model based on large-scale data with random forest," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 770–774, 2017.

[58] C. Wang, Z. Guo, S. Wang, L. Wang, and C. Ma, "Improving hyperspectral image classification method for fine land use assessment application using semisupervised machine learning," *Journal of Spectroscopy*, vol. 2015, Article ID 969185, 8 pages, 2015.

[59] W. Chunyang, G. Zengzhang, W. Shuangting et al., "A method of strip noise removal from hyper-spectral images by fusion of bilateral filtering and moment matching," *Journal of surveying and Mapping Science and Technology*, pp. 153–156, 2014.

[60] L. Dong, K. Jie, H. Gensheng et al., "Treatment of thick cloud and cloud shadow in remote sensing image based on support vector machine," *Acta Geodaetica Et Cartographica Sinica*, vol. 41, no. 2, pp. 225–616, 2012.

[61] N. Payet and S. Todorovic, "Hough forest random field for object recognition and segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 5, pp. 1066–1079, 2013.

[62] X. Wang and C. Chen, "Ship Detection for Complex Background SAR Images Based on a Multiscale Variance Weighted Image Entropy Method," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 2, pp. 184–187, 2017.

[63] J. Xia, W. Liao, J. Chanussot, P. Du, G. Song, and W. Philips, "Improving Random Forest With Ensemble of Features and Semisupervised Feature Extraction," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 7, pp. 1471–1475, 2015.

[64] P. Kontschieder, S. R. Bulò, M. Pelillo, and H. Bischof, "Structured Labels in Random Forests for Semantic Labelling and Object Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 10, pp. 2104–2116, 2014.

[65] J. L. Alba, L. Docio, and S. Ruibal, "Soft-competitive-growing classifier with unsupervised fine-tuning," in *Proceedings of the 1997 IEEE International Conference on Neural Networks, ICNN 1997*, vol. 3, pp. 1418–1423, June 1997.

[66] N. K. Anh, N. Van Linh, N. K. Toi, and N. T. Tarn, "Multi-labeled document classification using semi-supervived mixture model of Watson distributions on document manifold," in *Proceedings of the 2013 International Conference on Soft Computing and Pattern Recognition, SoCPaR 2013*, pp. 123–128, December 2013.

[67] O. Rajadell, P. García-Sevilla, V. C. Dinh, and R. P. W. Duin, "Improving hyperspectral pixel classification with unsupervised training data selection," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 3, pp. 656–660, 2014.

[68] F. Lin and W. W. Cohen, "Semi-supervised classification of network data using very few labels," in *Proceedings of the 2010 International Conference on Advances in Social Network Analysis and Mining, ASONAM 2010*, pp. 192–199, August 2010.

[69] M. Ristin, M. Guillaumin, J. Gall, and L. Van Gool, "Incremental Learning of Random Forests for Large-Scale Image Classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 3, pp. 490–503, 2016.

[70] H. Pang, S. L. George, K. Hui, and T. Tong, "Gene selection using iterative feature elimination random forests for survival outcomes," *IEEE Transactions on Computational Biology and Bioinformatics*, vol. 9, no. 5, pp. 1422–1431, 2012.

[71] N. Quadrianto and Z. Ghahramani, "A very simple safe-Bayesian random forest," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 6, pp. 1297–1303, 2015.

[72] V. E. Kosmidou, P. C. Petrantonakis, and L. J. Hadjileontiadis, "Enhanced Sign Language Recognition Using Weighted Intrinsic-Mode Entropy and Signer's Level of Deafness," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 41, no. 6, pp. 1531–1543, 2011.

[73] X. Ma, J. Guo, K. Xiao, and X. Sun, "PRBP: Prediction of RNA-Binding Proteins Using a Random Forest Algorithm Combined with an RNA-Binding Residue Predictor," *IEEE Transactions on Computational Biology and Bioinformatics*, vol. 12, no. 6, pp. 1385–1393, 2015.

[74] H. Phan, M. Maaß, R. Mazur, and A. Mertins, "Random regression forests for acoustic event detection and classification," *IEEE/ACM Transactions on Audio Speech and Language Processing*, vol. 23, no. 1, pp. 20–31, 2015.

[75] C. C. M. Chen, H. Schwender, J. Keith, R. Nunkesser, K. Mengersen, and P. MacRossan, "Methods for identifying SNP interactions: a review on variations of logic regression, random forest and Bayesian logistic regression," *IEEE Transactions on Computational Biology and Bioinformatics*, vol. 8, no. 6, pp. 1580–1591, 2011.

[76] A. Topîrceanu and G. Grosseck, "Decision tree learning used for the classification of student archetypes in online courses," *Procedia Computer Science*, vol. 112, pp. 51–60, 2017.

[77] J. Ham, Y. Chen, M. M. Crawford, and J. Ghosh, "Investigation of the random forest framework for classification of hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, no. 3, pp. 492–501, 2005.

[78] J. Wang, M. Li, Y.-T. Hu, and Y. Zhu, "Comparison of hospital charge prediction models for gastric cancer patients: neural network vs. decision tree models," *BMC Health Services Research*, vol. 9, no. 1, 2009.

[79] I. Chikalov, S. Hussain, and M. Moshkov, "Bi-criteria optimization of decision trees with applications to data analysis," *European Journal of Operational Research*, vol. 266, no. 2, pp. 689–701, 2018.

Hindawi
Advances in Multimedia
Volume 2018, Article ID 7481645, 6 pages
https://doi.org/10.1155/2018/7481645

*Research Article*

# Visual Tracking Based on Discriminative Compressed Features

**Wei Liu** [1] **and Hui Wang** [2]

[1] *Department of Modern Education Technology, Ludong University, Yantai, China*
[2] *Lab, CNCERT/CC, Yumin Road No. 3A, Beijing 100029, China*

Correspondence should be addressed to Wei Liu; ldulw@sina.com

Received 3 April 2018; Revised 13 June 2018; Accepted 11 July 2018; Published 1 August 2018

Academic Editor: Lei Zhang



Visual tracking is a challenging research topic in the field of computer vision with many potential applications. A large number of tracking methods have been proposed and achieved designed tracking performance. However, the current state-of-the-art tracking methods still can not meet the requirements of real-world applications. One of the main challenges is to design a good appearance model to describe the target's appearance. In this paper, we propose a novel visual tracking method, which uses compressed features to model target's appearances and then uses SVM to distinguish the target from its background. The compressed features were obtained by the zero-tree coding on multiscale wavelet coefficients extracted from an image, which have both the low dimensionality and discriminate ability and therefore ensure to achieve better tracking results. The experimental comparisons with several state-of-the-art methods demonstrate the superiority of the proposed method.

## 1. Introduction

Visual tracking aims at locating the target of interest from an image sequence, which is one of the most activated research topics in the field of computer vision with many potential applications such as video surveillance, human-computer interaction, navigation, and automatic driving. It has attracted increasing interest in the past few decades [1–16]. However, due to a variety of challenging factors such as illumination changes, pose deformation, and occlusion, the performance of visual tracking is still far away from requirements in practical applications. The main difficulty is that it is not easy to design a good appearance modeling method, which is not only good at distinguishing the target from its background but also being robust to the above-mentioned appearance changes. Finding a good appearance modeling is a challenging problem in many visual applications such as image classification [17–19] and video recognition [20–22].

In the literature, there are a variety of visual tracking methods with focus on developing effective appearance modeling methods. Most of these methods can be classified into two groups: generative methods and discriminative methods. The former learns generative features from samples that only contain the target, whose purpose is to represent the target as accurate as possible. The latter learns discriminative features from samples including both the target and its background, which usually involves solving an optimization function. To achieve better tracking performance, discriminative methods attracted more attention.

In this paper, to overcome the challenges caused by low contrast, illuminative changes, and scale changes, we propose a novel tracking method using discriminative compressed features, which is real-time and able to process multiple scales of the target. The main idea of the proposed method is that it combines compressive sensing and multiscale texture transformation to extract compressed texture features and then uses SVM to classify the target from its background. The compressed features have both the low dimensionality and discriminate ability and therefore ensure to achieve better tracking results. The experimental comparisons with several state-of-the-art methods demonstrate the superiority of the proposed method.

The rest of this paper is organized as follows. In Section 2, we review the work closely related to our proposed approach. Section 3 gives a detailed description of the proposed tracking method. Experimental results are reported and analyzed in Section 6. We conclude this paper in Section 6.

## 2. Related Work

In the past decades, there are many tracking methods that have been proposed, which can be roughly divided into generative methods and discriminative methods. The former focuses on modeling the appearance of the tracked target and then finds the candidate that is the most similar to the target template as the tracking result. The representative methods include those trackers based on sparse representation [23–29]. In [29], sparse coding is used to extract features from sampled patches. The local sparse features are then pooled into a global representation. In [28], an online learning sparse representation is proposed for visual tracking to handle occlusion. In [25], a joint sparse representation framework is used to combine multi-cue features for visual tracking. Since features from different cues describe the tracked target from different aspects, more robust tracking results can be obtained when multi-cue features are used. In [23], a biologically inspired appearance model is proposed to model target appearance, which is also based on features extracted using sparse coding.

The discriminative methods learn a binary classifier, which is then used to classify a candidate as the target or background [5, 8, 14, 16, 30–34]. In [30], Yakut and Kehtarnavaz proposed to track ice-hockey pucks by combining three pieces of information in ice-hockey video frames using an adaptive gray-level thresholding method. In [31], Topkaya et al. proposed a multiple object tracking method using tracklet clustering, which first obtains short yet reliable tracklets and then clusters the tracklets over time based on color and spatial and temporal attributes. In [32], Wang and Zhao proposed an adaptive appearance model called Principal Component-Canonical Correlation Analysis (P3CA) to extract discriminative features for object tracking. In [14], Qi et al. propose a CNN based tracking method, which uses correlation filters to construct six weak trackers on outputs of six CNN layers. These weak trackers are then adaptively combined by a Normal Hedge algorithm. In [34], a further improved method is proposed which uses a SNT to compute the loss of each weak tracker, which achieves better tracking performance.

## 3. Discriminative Compressed Features

*3.1. Multiscale Wavelet Transformation.* Multiscale wavelet is a kind of wavelet which consists of more than two scale functions. It preserves the local properties of time-frequency domains while overcoming the drawbacks of a single wavelet and therefore has more properties of different frequencies. In this paper, we choose the GHM multiscale wavelet [35], which can be obtained by recursively calculating as follows:

$$v_{j,k} = \sum_m G_{m-2k} v_{j-1,m} \tag{1}$$

$$w_{j,k} = \sum_m H_{m-2k} v_{j-1,m} \tag{2}$$

where $v_{j,k}$ and $w_{j,k}$ are low-frequency coefficients and high-frequency coefficients of the $j$th scale of the input signal, respectively. $v_{j-1,m}$ denotes the low-frequency coefficients of the $(j-1)$th scale; $k$ and $m$ are the indices of the current scales, which are dependent on the input image. The multiwavelet filters are defined as

$$G_0 = \begin{bmatrix} \frac{3}{5\sqrt{2}} & \frac{4}{5} \\ -\frac{1}{20} & -\frac{3}{10\sqrt{2}} \end{bmatrix}$$

$$G_1 = \begin{bmatrix} \frac{3}{5\sqrt{2}} & 0 \\ \frac{9}{20} & \frac{1}{\sqrt{2}} \end{bmatrix} \tag{3}$$

$$G_2 = \begin{bmatrix} 0 & 0 \\ \frac{9}{20} & -\frac{3}{10\sqrt{2}} \end{bmatrix}$$

$$G_3 = \begin{bmatrix} 0 & 0 \\ -\frac{1}{20} & 0 \end{bmatrix} \tag{4}$$

$$H_0 = \frac{1}{10} \begin{bmatrix} -\frac{1}{2} & -\frac{3}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 3 \end{bmatrix}$$

$$H_1 = \frac{1}{10} \begin{bmatrix} \frac{9}{2} & -\frac{10}{\sqrt{2}} \\ -\frac{9}{\sqrt{2}} & 0 \end{bmatrix} \tag{5}$$

$$H_2 = \frac{1}{10} \begin{bmatrix} \frac{9}{2} & -\frac{3}{\sqrt{2}} \\ \frac{9}{\sqrt{2}} & -3 \end{bmatrix}$$

$$H_3 = \frac{1}{10} \begin{bmatrix} -\frac{1}{2} & 0 \\ -\frac{1}{\sqrt{2}} & 0 \end{bmatrix} \tag{6}$$

*3.2. Compressed Multiscale Features.* It is easy to obtain low-frequency components and high-frequency components after the signals are filtered by wavelet transformation. In general, most energy of the signal is in the low-frequency components. In contrast, high-frequency components of the signal reflect the details of the input image. Therefore, the simplest way of compressing the input image is to set the high-frequency coefficients to be zero when reconstructing the input image using wavelet transformation. The other option is to set the high-frequency coefficients of some local regions to be zero or to set the high-frequency coefficients based on a threshold, which will cause severe loss of image details, blurred images after compression, or loss of image information.

Wavelet transformation is able to composite the input image at different scales. More importantly, the subimage at each resolution has different frequency properties and different orientation selections. Therefore, it can be used to

encode different information of the input image at different scales.

It is widely thought of the fact that the targets in a video sequence are redundant in both spatial and frequency domains. The former indicates the adjacent pixels have spatial correlation. The latter indicates that the adjacent frequencies of a pixel have some kinds of correlation. On the other hand, the statistical features of image signals indicate that large coefficients always exist in low-frequency regions and therefore small bits can be assigned to those small coefficients or they will not be transmitted at all. It will cause high compression rates and very small information loss.

The compression method based on multiscale wavelet transformation applies the zero-tree coding to compression of high spectral images. The principle behind this method is that it exploits the structure correlation of high spectral images to construct only one effective (shared) image and then further determine the positions of nonzeros of multiscale wavelet coefficients. The shared image is obtained by combining multiscale frequency coefficients and therefore removes spatial redundancy and frequency redundancy with the purpose of improving compression efficiency.

The one-dimensional wavelet transformation filters the input signal by low-pass filtering and high-pass filtering and then obtains low-frequency components and high-frequency components by downsampling. According to Mallat algorithm, two-dimensional wavelet transformation can be implemented by several one-dimensional wavelet transformation and obtain low-frequency and high-frequency components, respectively. Given an input image with m rows and n columns, the process of 2D wavelet transformation is that it first decomposes the input image along its each row using 1D wavelet transformation, which will obtain L and H two parts. The second step is to decompose the L and H parts along its column using 1D wavelet transformation. With these two steps, the input image will get four parts (LL, HL, LH, and HH). The second level, third level, or higher level's wavelet transformation can be obtained by using such a process on the former level. Therefore, the wavelet transformation is an iterative process.

To meet the real-time requirements, the dimensionality of appearance features should not be too high. To meet this requirement, in this paper, we adopt compressive sensing to reduce the dimensionality of high-dimensional appearance features. Let $u \in \mathbf{R}^D$ be the wavelet features and $\Gamma$ be a random matrix computed using the same method as in [26]. The compressed features $v \in \mathbf{R}^d$ can be computed as $v = \Gamma u$.

## 4. Discriminative SVM Tracking

SVM is for classic binary pattern classification since it was proposed by Vapnik in 1995. In this paper, we use SVM as our tracking model.

*4.1. SVM Tracking.* To classify the target from its background, our tracking method tries to find a hyperplane in the D-dimensional compressed feature space to distinguish the features of the target and its background.

To achieve this aim, the optimization objective is to maximize the classifier's margin in the feature space. In other words, we need to meet the following conditions:

$$x_i \cdot w + b \geq 0 \quad \text{if } y_i = +1 \tag{7}$$

$$x_i \cdot w + b \leq 0 \quad \text{if } y_i = -1 \tag{8}$$

where $y_i$ is the class label of the $i$th sample. For example, if the sample is target, $y_i = +1$. Otherwise, if the sample is background, $y_i = -1$.

Given training samples and their corresponding labels, we first extract compressed features from each sample using the method introduced in Section 3. The features with their labels can then be fed to SVM to train SVM's parameters. In the tracking stage, for each target candidate, we can also extract the compressed features using the same method as like in the training stage. Then we can feed the extracted features to SVM to predicate its label. If the features are classified as +1, it is considered as the potential target. Otherwise, it is not considered as the potential target. The final target is selected as the potential target candidate with the largest probability.

*4.2. Model Update.* To make the proposed tracker adapt to target appearance changes over time, the tracker needs to be updated online. To this aim, we update the model using the collected positive and negative samples. In particular, we collect a set of positive and negative samples at time $t$. Using the proposed appearance model, we can extract the compressed features for all positive and negative samples. Then the SVM model can be updated as

$$u_l^1 = \lambda u_l^1 + (1-\lambda)\, u_l^1 \tag{9}$$

$$u_l^0 = \lambda u_l^0 + (1-\lambda)\, u_l^0 \tag{10}$$

$$\delta_l^1 = \sqrt{\lambda \left(\delta_l^1\right)^2 + (1-\lambda)\left(\delta_l^1\right)^2 + \lambda(1-\lambda)\left(u_l^1 - u_1\right)^2} \tag{11}$$

$$\delta_l^0 = \sqrt{\lambda \left(\delta_l^0\right)^2 + (1-\lambda)\left(\delta_l^0\right)^2 + \lambda(1-\lambda)\left(u_l^0 - u_0\right)^2} \tag{12}$$

where $\lambda$ denotes the learning rate, which controls the speed of model updating.

$$u^1 = \frac{1}{\alpha}\sum_{k=1}^{\alpha} v_{1,l}^k \tag{13}$$

$$u^0 = \frac{1}{\alpha}\sum_{k=1}^{\alpha} v_{0,l}^k \tag{14}$$

$$\delta^1 = \sqrt{\frac{1}{\alpha}\sum_{k=1}^{\alpha}\left(v_{1,l}^{(k)} - u^1\right)^2} \tag{15}$$

$$\delta^0 = \sqrt{\frac{1}{\alpha}\sum_{k=1}^{\alpha}\left(v_{0,l}^{(k)} - u^0\right)^2} \tag{16}$$

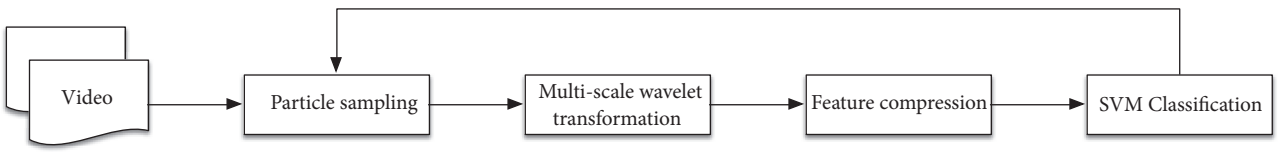



Figure 1: The flowchart of the proposed tracking method.

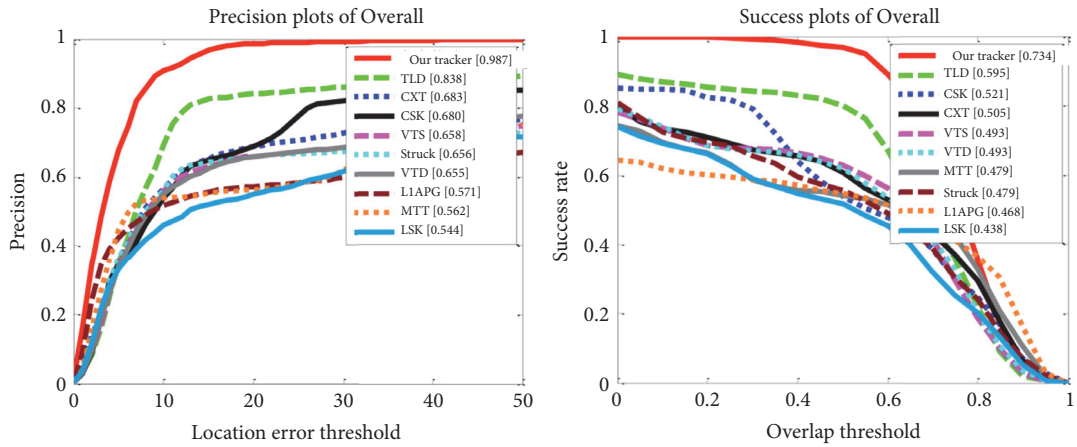



Figure 2: Overall precision plots and success plots on the test sequences.

## 5. Experiment Results

The target tracking is implemented in a particle filter framework. Several sequences from the OTB100 dataset have been chosen to evaluate the proposed tracking method. At the first frame, the target is initialized manually. Of course, the target can be initialized by a detector when the method is applied in real systems. After the target is initialized, a set of particles are sampled around the target. Whether each particle is considered as the target or not is based on the output of SVM scoring. In the next frame, the particles are sampled using the tracking result in the last frame as mean and a predefined covariance. The process is repeated frame by frame. The flowchart of the proposed tracking method is shown in Figure 1.

To test the performance of the proposed method, we compared the proposed method to several state-of-the-art trackers including TLD [36], CXT [1], Struck [37], L1APG [38], and MTT [39]. By quantitatively and qualitatively analyzing the experimental results, we demonstrate the outstanding performance of the proposed method.

Two frame based metrics widely used in tracking performance evaluation are (1) center location error, which is defined as the Euclidean distance between the central location of the tracked target and the manually labeled ground-truthed position; (2) bounding box overlap which is the ratio of the areas of the intersection and the union of the bounding box indicating the tracked subject and the ground-truthed bounding box. To measure the overall performance of a tracker on a test sequence, success rate and precision score are adopted. The former is computed as the percentage of image frames, which have a bounding box overlap larger than a given threshold. The latter is the percentage of image frames, which have a central position error less than a given threshold. In each case, when multiple thresholds are used, a curve is drawn to show how success rates or precision scores are affected by different thresholds. These curves are, namely, success plot and precious plot, respectively. In practical evaluations, we average the curves of a tracker over all the sequences, which have the same challenge and show a curve for each challenge item rather than a test sequence. In addition, we use the area under curve (AUC) of the success plot to quantitatively measure the overall performance of a tracker on a challenge item.

*5.1. Quantitative Comparison.* The overall precision plots and success plots are shown in Figure 2, from which we can see that the proposed method outperforms other methods in terms of the overall precision plots and success plots.

*5.2. Qualitative Comparison.* To further show the superiority of the proposed method, we show several examples of tracking results on Figures 3 and 4. As we can see from Figure 3, the proposed tracker outperforms other trackers on several representative frames on two sequences. More tracking results are shown in Figure 4, from which we can see that the proposed tracker also achieves the best tracking performance.

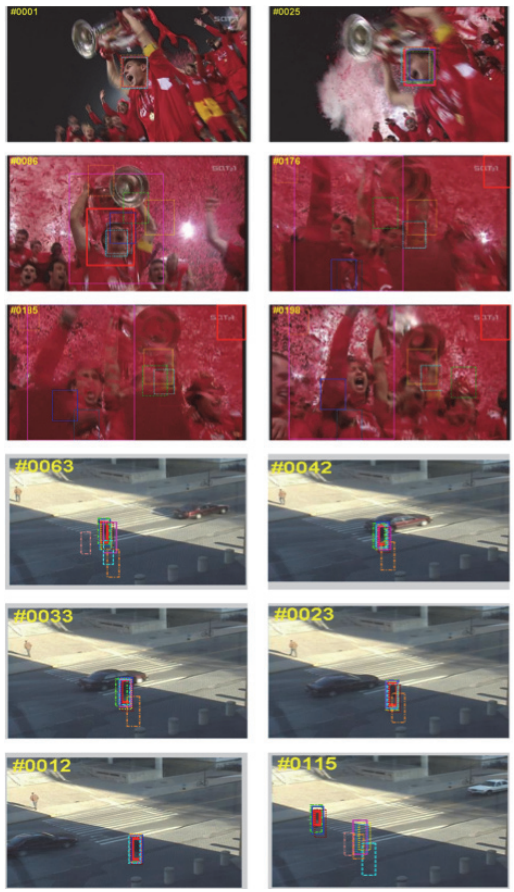



Figure 3: Examples of tracking results on representative frames of two sequences.


Figure 4: Examples of tracking results on representative frames of other four sequences.

## 6. Conclusion

In this paper, we propose to use compressed features to model the tracked target's appearance and then use SVM to perform tracking. The experimental results indicate the proposed method outperforms several state-of-the-art methods. The advantages of the proposed method are twofold: (1) It is good at handling scale changes of the target over time because the used features are obtained by multiscale wavelet transformation. (2) The speed of the proposed method can achieve real-time because the dimensionality of the used features was reduced by compressed sensing techniques.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

The research is supported by Project of Shandong Province Higher Educational Science and Technology Program (no. J14LN64).

## References

[1] T. B. Dinh, N. Vo, and G. Medioni, "Context tracker: exploring supporters and distracters in unconstrained environments," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 1177–1184, June 2011.

[2] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *Proceedings of the European Conference on Computer Vision*, pp. 702–715, 2012.

[3] J. Kwon and K. M. Lee, "Tracking by sampling trackers," in *Proceedings of the 2011 IEEE International Conference on Computer Vision (ICCV '11)*, pp. 1195–1202, November 2011.

[4] J. Han and P. H. N. De With, "Real-time multiple people tracking for automatic group-behavior evaluation in delivery simulation training," *Multimedia Tools and Applications*, vol. 51, no. 3, pp. 913–933, 2011.

[5] Z. Han, Q. Ye, and J. Jiao, "Combined feature evaluation for adaptive visual object tracking," *Computer Vision and Image Understanding*, vol. 115, no. 1, pp. 69–80, 2011.

[6] 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.

[7] J. Han, E. J. Pauwels, P. M. De Zeeuw, and P. H. N. De With, "Employing a RGB-D sensor for real-time tracking of humans across multiple re-entries in a smart environment," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 2, pp. 255–263, 2012.

[8] S. Gao, Z. Han, C. Li, Q. Ye, and J. Jiao, "Real-Time Multipedestrian Tracking in Traffic Scenes via an RGB-D-Based Layered Graph Model," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2814–2825, 2015.

[9] L. Zhang, W. Wu, T. Chen, N. Strobel, and D. Comaniciu, "Robust object tracking using semi-supervised appearance dictionary learning," *Pattern Recognition Letters*, vol. 62, pp. 17–23, 2015.

[10] S. Zhang, H. Zhou, H. Yao, Y. Zhang, K. Wang, and J. Zhang, "Adaptive NormalHedge for robust visual tracking," *Signal Processing*, vol. 110, pp. 132–142, 2015.

[11] S. Zhang, S. Kasiviswanathan, P. C. Yuen, and M. Harandi, "Online dictionary learning on symmetric positive definite manifolds with vision applications," in *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 3165–3173, January 2015.

[12] Z. He, X. Li, X. You, D. Tao, and Y. Y. Tang, "Connected component model for multi-object tracking," *IEEE Transactions on Image Processing*, vol. 25, no. 8, pp. 3698–3711, 2016.

[13] X. Li, Q. Liu, Z. He, H. Wang, C. Zhang, and W.-S. Chen, "A multi-view model for visual tracking via correlation filters," *Knowledge-Based Systems*, vol. 113, pp. 88–99, 2016.

[14] Y. Qi, S. Zhang, L. Qin et al., "Hedged deep tracking," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 4303–4311, July 2016.

[15] Z. He, S. Yi, Y.-M. Cheung, X. You, and Y. Y. Tang, "Robust Object Tracking via Key Patch Sparse Representation," *IEEE Transactions on Cybernetics*, vol. 47, no. 2, pp. 354–364, 2017.

[16] R. Shi, J. Zhang, Z. Xie, J. Gao, and X. Zheng, "Robust tracking with per-exemplar support vector machine," *IET Computer Vision*, vol. 9, no. 5, pp. 699–710, 2015.

[17] P. Wilf, S. Zhang, S. Chikkerur, S. A. Little, S. L. Wing, and T. Serre, "Computer vision cracks the leaf code," *Proceedings of the National Acadamy of Sciences of the United States of America*, vol. 113, no. 12, pp. 3305–3310, 2016.

[18] L. Liu, Z. Lin, L. Shao, F. Shen, G. Ding, and J. Han, "Sequential discrete hashing for scalable cross-modality similarity retrieval," *IEEE Transactions on Image Processing*, vol. 26, no. 1, pp. 107–118, 2017.

[19] Y. Guo, G. Ding, L. Liu, J. Han, and L. Shao, "Learning to hash with optimized anchor embedding for scalable retrieval," *IEEE Transactions on Image Processing*, vol. 26, no. 3, pp. 1344–1354, 2017.

[20] S. Zhang, H. Yao, X. Sun et al., "Action recognition based on overcomplete independent components analysis," *Information Sciences*, vol. 281, pp. 635–647, 2014.

[21] F. Jiang, S. Zhang, S. Wu, Y. Gao, and D. Zhao, "Multi-layered gesture recognition with Kinect," *Journal of Machine Learning Research (JMLR)*, vol. 16, pp. 227–254, 2015.

[22] K. Chen, G. Ding, and J. Han, "Attribute-based supervised deep learning model for action recognition," *Frontiers of Computer Science*, vol. 11, no. 2, pp. 219–229, 2017.

[23] S. Zhang, X. Lan, H. Yao, H. Zhou, D. Tao, and X. Li, "A biologically inspired appearance model for robust visual tracking," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2357–2370, 2017.

[24] S. Zhang, X. Lan, Y. Qi, and P. C. Yuen, "Robust Visual Tracking via Basis Matching," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 3, pp. 421–430, 2017.

[25] X. Lan, S. Zhang, and P. C. Yuen, "Robust joint discriminative feature learning for visual tracking," in *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, pp. 3403–3410, July 2016.

[26] S. Zhang, H. Zhou, F. Jiang, and X. Li, "Robust visual tracking using structurally random projection and weighted least squares," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 11, pp. 1749–1760, 2015.

[27] S. Zhang, H. Yao, X. Sun, and X. Lu, "Sparse coding based visual tracking: review and experimental comparison," *Pattern Recognition*, vol. 46, no. 7, pp. 1772–1788, 2013.

[28] S. H. Zhang, H. Yao, H. Zhou, X. Sun, and S. H. Liu, "Robust visual tracking based on online learning sparse representation," *Neurocomputing*, vol. 100, pp. 31–40, 2013.

[29] S. Zhang, H. Yao, X. Sun, and S. Liu, "Robust visual tracking using an effective appearance model based on sparse coding," *ACM Transactions on Intelligent Systems and Technology*, vol. 3, no. 3, pp. 43:1–43:18, 2012.

[30] M. Yakut and N. Kehtarnavaz, "Ice-hockey puck detection and tracking for video highlighting," *Signal, Image and Video Processing*, vol. 10, no. 3, pp. 527–533, 2016.

[31] I. S. Topkaya, H. Erdogan, and F. Porikli, "Tracklet clustering for robust multiple object tracking using distance dependent Chinese restaurant processes," *Signal, Image and Video Processing*, vol. 10, no. 5, pp. 795–802, 2016.

[32] Y. Wang and Q. Zhao, "Robust object tracking via online Principal Component–Canonical Correlation Analysis (P3CA)," *Signal, Image and Video Processing*, vol. 9, no. 1, pp. 159–174, 2015.

[33] D. Shan and C. Zhang, "Visual tracking using IPCA and sparse representation," *Signal, Image and Video Processing*, vol. 9, no. 4, pp. 913–921, 2015.

[34] Y. Qi, S. Zhang, L. Lei Qin et al., "Hedging Deep Features for Visual Tracking," in *Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence (IEEE T-PAMI)*, 2018.

[35] J. Sembiring, A. S. Sabzevary, and K. Akizuki, "Stochastic process on multiwavelet," *IFAC Proceedings Volumes*, vol. 35, no. 1, pp. 211–215, 2002.

[36] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: bootstrapping binary classifiers by structural constraints," in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 49–56, June 2010.

[37] S. Hare, A. Saffari, and P. H. S. Torr, "Struck: structured output tracking with kernels," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '11)*, pp. 263–270, IEEE, Barcelona, Spain, November 2011.

[38] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real Time Robust L1 Tracker Using Accelerated Proximal Gradient Approach," in *Proceedings of the IIEEE Conference on Computer Vision and Pattern Recognition*, pp. 1830–1837, June 2012.

[39] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja, "Robust visual tracking via multi-task sparse learning," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

Hindawi
Advances in Multimedia
Volume 2018, Article ID 3202495, 11 pages
https://doi.org/10.1155/2018/3202495

*Research Article*

# Impostor Resilient Multimodal Metric Learning for Person Reidentification

**Muhamamd Adnan Syed, Zhenjun Han, Zhaoju Li, and Jianbin Jiao**

*University of Chinese Academy of Sciences, Beijing, China*

Correspondence should be addressed to Zhenjun Han; hanzhj@ucas.ac.cn

Received 15 January 2018; Accepted 22 March 2018; Published 3 May 2018

Academic Editor: Deepu Rajan



In person reidentification distance metric learning suffers a great challenge from impostor persons. Mostly, distance metrics are learned by maximizing the similarity between positive pair against impostors that lie on different transform modals. In addition, these impostors are obtained from *Gallery* view for query sample only, while the Gallery sample is totally ignored. In real world, a given pair of query and Gallery experience different changes in pose, viewpoint, and lighting. Thus, impostors only from *Gallery* view can not optimally maximize their similarity. Therefore, to resolve these issues we have proposed an impostor resilient multimodal metric (IRM3). IRM3 is learned for each modal transform in the image space and uses impostors from both *Probe* and *Gallery* views to effectively restrict large number of impostors. Learned IRM3 is then evaluated on three benchmark datasets, VIPeR, CUHK01, and CUHK03, and shows significant improvement in performance compared to many previous approaches.

## 1. Introduction

Person reidentification (Re-ID) matches a given person across a large network of nonoverlapping cameras [1], and is fundamentally used for person tracking in camera networks. Despite years of research, reidentification is still a challenging problem as the data space in Re-ID is multimodal (modal in our work is defined as the space which is formed by the joint combination of different changes a given pair images of the same person undergo in different camera views) and the observed images in different views undergo various different changes in poses [2], viewpoints [3], lighting [4], background clutter, and also experience occlusion.

Most approaches in Re-ID can mainly be divided into two categories: robust features extraction [5–13] for representation and globally learning distance metric for matching [14, 15]. These global metrics [16–19] project features into low dimension subspace where they tend to maximize the discrimination among different persons; however, these metrics still suffer a great challenge from impostor (an impostor is a person that belongs to the other person and, however, possess higher similarity with the given query than the right Gallery sample) samples [20, 21]. Though, in past some attempts are made to eliminate impostors [14, 20–22], however, all these attempts have not given due consideration of different transform modals on which the reidentification images lie [23]. This situation is illustrated in Figure 1, where we have shown three transform modals $M_1$, $M_2$, and $M_3$ in the image space. $M_1$ contains a positive pair (query and Gallery) enclosed in green rectangles for which a metric is learned, while there are two more pairs lying in modals $M_2$ and $M_3$, respectively. View b images (enclosed in red rectangles) in $M_2$ and $M_3$ are similar to query in $M_1$ and, thus, are impostors for query sample. In conventional approaches [14, 20–22], the metric between query and Gallery samples in $M_1$ is learned using the impostor sample from $M_2$ (Metric $DM_1$) or $M_3$ (Metric $DM_2$) as a constraint. Therefore, when the similarity for positive pair is learned under the constraint of an impostor person lying on a different transform modal other than the positive pair, then the learned similarity metric would not be the optimal matching function, which can be proved from poor retrieval results in *Ranklist 1* and *Ranklist 2* in Figure 1.

Further, in Figure 1 previous approaches [14, 20–22] have used impostor samples for query sample only from the *Gallery view*, while totally ignoring the Gallery sample. Therefore, to resolve the above shortcomings in [14, 20–22] we have proposed an impostor resilient multimodal metric,

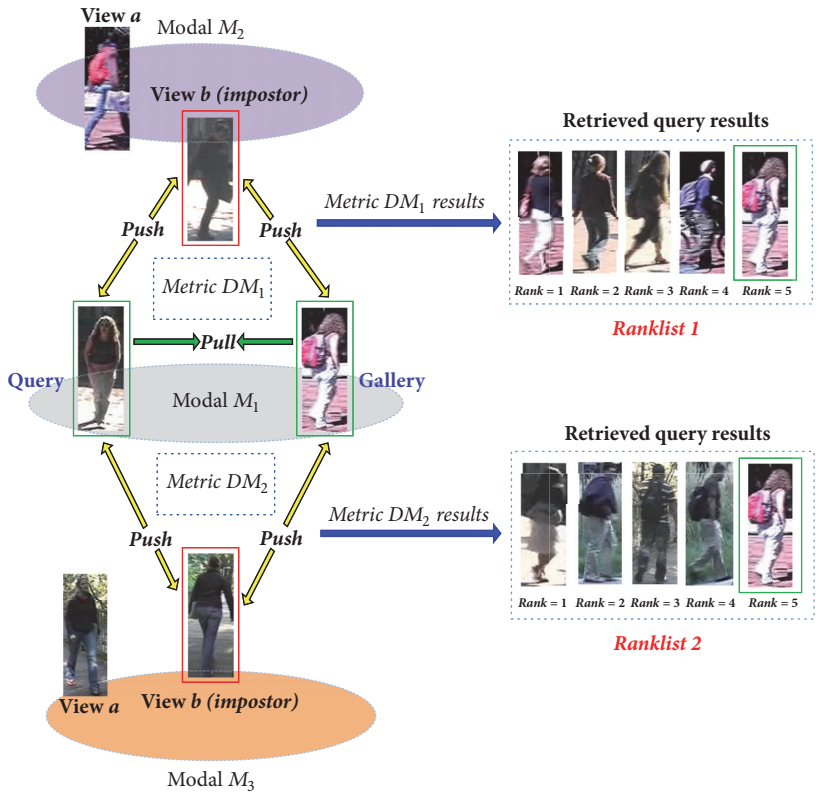



FIGURE 1: Three Modals $M_1$, $M_2$, and $M_3$ in Image Space. Query and Gallery lie in modal $M_1$, while one impostor for query lies in modal $M_2$ and the other in modal $M_3$. Metric $DM_1$ is learned using the impostor from modal $M_2$, and $DM_2$ is learned using the impostor from modal $M_3$. Then, the obtained retrieval results of $DM_1$ and $DM_2$ are shown in *Ranklist 1* and *Ranklist 2*, respectively. Correct Match is in green rectangle.

referred as IRM3, which eliminates the impostors largely and attains an optimal matching between positive pair. The objective of IRM3 is to maximize the matching of a positive pair against both the negative gallery samples (NGS) (samples which are not impostors and belong to different persons), as well as against impostors by taking into account the modal a given pair, its negative gallery samples, and its impostors reside. Further, in contrast to [14, 20–22], it also takes into consideration impostor samples for both the query and its respective Gallery sample. This pair of impostors are referred to as *Cross views* impostors (CVI) which are obtained for query and Gallery samples from their opposite views and help in further maximizing the similarity between given query and Gallery samples. The contributions of our impostor resilient multimodal metric IMR3 are as follows:

(i) Improving impostors resistance by jointly exploiting the transform modals [23], as well as impostor samples from both *Probe* and *Gallery* views;

(ii) With our IMR3 approach a significant gain in performance is obtained in Multikernel Local Fisher Discriminant Analysis (MK-LFDA) [44].

## 2. Methodology

Figure 2 shows the framework of our IRM3. In Figure 2, first color and texture features are extracted from each training sample; then, different modals are discovered in the image space. These modals are discovered by using sum of squares clustering which is explained in Section 2.2. Finally, for each modal cross views impostors (CVI) (explained in Section 2.3) and negative gallery samples (NGS) (explained in Section 2.4) are generated to train the modal metric $M_k$ for each transform modal $k$. In our work, the modal metric $M_k$ is learned using MK-LFDA [44], and the learning procedure is explained in Section 2.6. Finally, in Section 2.7 we have explained how we have performed matching between test query and Gallery.

*2.1. Feature Extraction.* RGB, HSV, LAB, YCbCr, and SCNCD histograms are extracted according to similar settings in [45] using 32 bins per channel, and settings in [12], respectively. Then, all five features are concatenated together. Similarly, DenseSIFT, SILTP, and HOG are extracted according to the settings in [46], [11], and [47], respectively, and are concatenated together. Dimension of color and texture features after concatenation become large, and since Re-ID data is multiview we have used CCA [48] to reduce dimension. However, to keep the local discriminative information of each type of feature we have applied CCA to color and texture features individually. By cross validation on VIPeR and CUHK03 we obtained optimal dimension for color feature to be 900, and texture feature to be 700. Finally, the

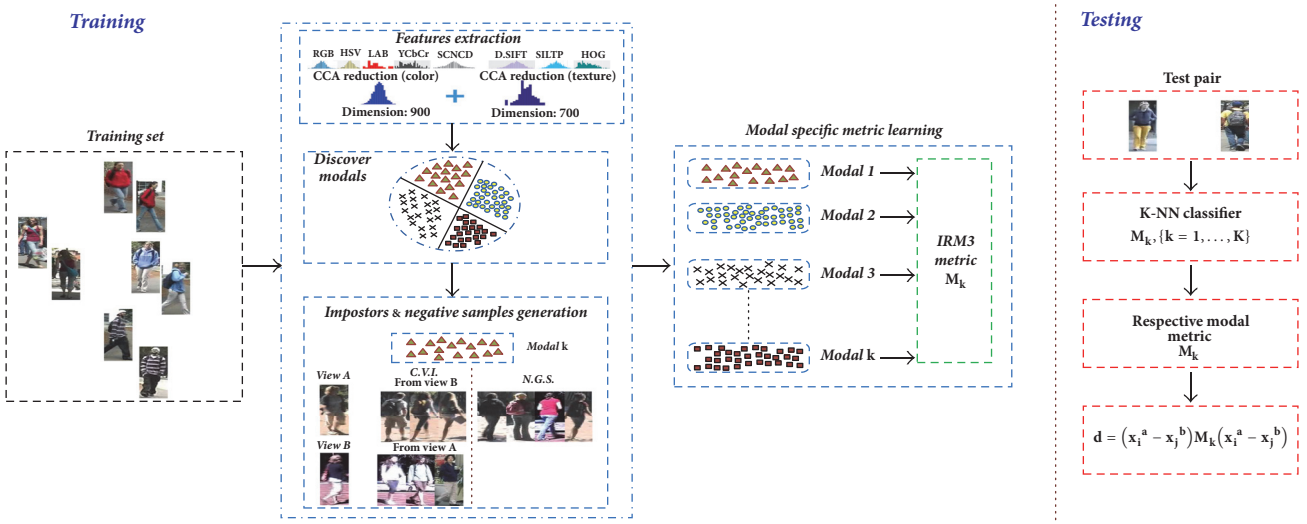



FIGURE 2: Impostor resilient multimodal metric learning (IRM3) for person reidentification.

reduced color and texture features are concatenated to form a feature vector $F$ of size 1600.

*2.2. Partition Image Space.* Let $X$ be the image space of a camera view; then $X$ is

$$X = \{x_i\}_{i=1,\ldots,n}, \tag{1}$$

where $x_i$ is the feature representation $F_i$ of person $i$ and $n$ are the number of persons in $X$. Since images in $X$ lie on different transform modals, therefore, there exist distinct clusters of different modals in $X$. Each of these modal clusters has its own unique transformation and visual patterns; thus, all the persons belonging to a modal $k$ can be obtained using sum of squares clustering as

$$S_w = \frac{1}{n}\sum_{k=1}^{K}\sum_{i=1}^{n} z_{k,i}\,(x_i - m_k)\,(x_i - m_k)^T, \tag{2}$$

where $K$ are the number of modals in $X$, $S_w$ is scatter matrix of within transform modals, $z_{k,i}$ is the association of $x_i$ with transform modal $k$, and $m_k$ is the center of the $k$th transform modal.

In (2), each modal center $m_k$ is critical in discovering distinct, stable, and nonempty modals in $X$. Thus, choosing any sample $(x_i^a, x_i^b)$ as center $m_k$ of any given modal $k$, it is necessary to make sure it is a right choice. In order to make sure a chosen modal center is right it has to fulfill two conditions: First, (i) if the chosen sample $(x_i^a, x_i^b)$ is a center of modal cluster $k$, then, all the persons in modal $k$ will be its neighbors, and it has the highest number of nearest neighbors. Second, (ii) center $m_k$ and all its nearest neighbors lie on the same modal; therefore, these neighbors will share similar patterns with the center $m_k$ in both *Probe* and *Gallery* views.

Now, we compute the number of nearest neighbors for each person in training set by taking into consideration the above two conditions. For this purpose, we have used both Probe $(x_i^a)$ and Gallery $(x_i^b)$ samples of each person to obtain four lists of neighbors, which are computed from both camera views. To acquire most reliable neighbors we then select only top@40 (here, the reason to choose top@40 neighbors is to maintain maximum reliability with minimum time and memory cost in large datasets. For instance, when we have $k = 16$ modals in CUHK03, then, in each modal there will be at least 78 training persons. Now, to obtain a center sample $x_i$ of any modal it must have at least 51% neighbors in that modal, and thus, we take top@40 neighbors which is in actual 52% proportion of the training persons in a modal to find out whether $x_i$ is a center or not) (top@20 for VIPeR) neighbors from each list and then perform an intersection operation among all the four lists to obtain the cardinality value, as well as IDs of the neighbors which are common in both *Probe* and *Gallery* views of a given person. This cardinality value and the IDs of the obtained neighbors are then stored in a matrix. Further, this procedure is repeated for the rest of the remaining $n - 1$ persons in the training set, and then their cardinality values, as well as IDs of the neighbors, are also stored in the same matrix.

Using this matrix we will now obtain our $K$ initial centers for $K$ modal transforms. These $K$ centers are chosen as the $K$ top persons with highest number of neighbors. However, it could be possible that two or more persons can have the same cardinality value, as well as share the same nearest neighbors IDs. In that condition simply choosing top $K$ persons will not be the best solution; instead, we chose only those top $K$ persons that do not have any person IDs common in their neighbors lists. In addition, for situations where more than two persons have the same cardinality and share same neighbors IDs, we randomly chose any one person from

them to represent that modal center. Finally, getting the $K$ modal centers the optimal partitioning of the image space $X$ is obtained by minimizing the trace of within transform modals scatter matrix as

$$\arg\min \operatorname{tr}\left(S_w\right). \tag{3}$$

Though the image space is partitioned into $K$ modals, however, to ensure the obtained modals are distinct and stable (in our work a stable modal is formed when it contains at least 15% training persons) we have updated the modal centers and repartitioned the space for further $t = 3$ times. The modal centers are updated as

$$m_k = \frac{1}{N'_k}\sum_{i=1}^{n} z_{k,i} * x_i, \tag{4}$$

where $N'_k$ is the number of persons in modal $k$ and given as

$$N'_k = \sum_{i=1}^{n} z_{k,i}. \tag{5}$$

Computing the initial modal centers is computationally tedious in our work, however, it has still moderate computational burden. For the training size of $n$ persons the complexity is about $\mathcal{O}(t \times K \times n)$, where $t$ is the number of iterations, and $K$ is the number of modals.

*2.3. Cross Views Impostors (CVI).* After getting the distinct modals in the image space $X$, we can now obtain the set of CVI for each positive pair $(x_i^a, x_i^b)$ lying in modal $k$ from both of its *Probe* and *Gallery* views. We believe in real world situation (open set) where a positive pair has always limited or few samples; these CVI can be exploited to deliver subtle and differentiating information in metric learning that can differentiate a given pair more efficiently against large number of diverse real world impostors, as well as negative gallery samples. These impostors are obtained by comparing the similarity value of a given person pair against the other persons in *Gallery* and *Probe* views. First, the similarity values for a Probe sample $x_i^a$ are computed with the whole *Gallery* view using metric $M_{\text{ini}}$ and CCA reduced feature $F$ as

$$S^i_{\text{probe}} = \left(x_i^a - x_j^b\right)^T M_{\text{ini}} \left(x_i^a - x_j^b\right), \tag{6}$$

where $x_i$ and $x_j$ are CCA reduced feature $F$ of person $i$ and $j$, while $M_{\text{ini}}$ is a globally learned metric with feature $F$ using K-LFDA [45]. We have used *linear* kernel to save memory and computational time. Similarly, the similarity values for Gallery person $x_i^b$ are obtained with the whole *Probe* view as

$$S^i_{\text{gallery}} = \left(x_i^b - x_j^a\right)^T M_{\text{ini}} \left(x_i^b - x_j^a\right). \tag{7}$$

These obtained values $S^i_{\text{probe}}$ and $S^i_{\text{gallery}}$ for person $(x_i^a, x_i^b)$ in modal $k$ are then stored into two sets as

$$\begin{aligned} \text{Sim}_{x_i^a} &= \left[S^i_{\text{probe},i'}\right]_{i'=1,\ldots,N'_k}, \\ \text{Sim}_{x_i^b} &= \left[S^i_{\text{gallery},i'}\right]_{i'=1,\ldots,N'_k}, \end{aligned} \tag{8}$$

where $N'_k$ refers to the number of persons in a modal $k$. Now, we compare each similarity value in these sets with the reference similarity value $S^{\text{ref}}_{(x_i^a, x_i^b)}$ of a given pair $(x_i^a, x_i^b)$ to obtain its CVI set $\text{Set}^{\text{C.V.I.}}_{(x_i^a, x_i^b)}$ as

$$\begin{aligned} &\text{Set}^{\text{CVI}}_{(x_i^a, x_i^b)} = \left[x_p\right], \\ &\text{here } x_p = S^i_{\text{probe},p} < S^{\text{ref}}_{(x_i^a, x_i^b)}, \text{ or } x_p = S^i_{\text{gallery},p} < S^{\text{ref}}_{(x_i^a, x_i^b)}, \end{aligned} \tag{9}$$

and $p$ is the index of impostor person, and $S^{\text{ref}}_{(x_i^a, x_i^b)}$ is computed as

$$S^{\text{ref}}_{(x_i^a, x_i^b)} = \left(x_i^a - x_i^b\right)^T M_{\text{ini}} \left(x_i^a - x_i^b\right). \tag{10}$$

Further, using (6)–(10), CVI set $\text{Set}^{\text{C.V.I.}}_{(x_i^a, x_i^b)}$ for all the $N'_k$ persons in the modal $k$ are computed. The computational cost of generating cross views impostors for a modal $k$ is about $\mathcal{O}(3 \times N'_k)$, where $N'_k \ll n$.

*2.4. Negative Gallery Samples (NGS).* We have also used negative gallery samples (NGS) to learn metric $M_k$. Set of NGS, denoted as $\text{Set}^{\text{Ng}}_{(x_i^a, x_i^b)}$, for person pair $(x_i^a, x_i^b)$ are obtained from Gallery view only as

$$\begin{aligned} &\text{Set}^{\text{Ng}}_{(x_i^a, x_i^b)} = \left[x_q^b\right], \\ &\qquad \text{where } q \neq p \text{ in } \text{Set}^{\text{CVI}}_{(x_i^a, x_i^b)}, \ q \neq i \text{ for probe } i, \end{aligned} \tag{11}$$

where $q$ is the index of NGS Further, the set of NGS $\text{Set}^{\text{Ng}}_{(x_i^a, x_i^b)}$ for all $N'_k$ persons in modal $k$ is then obtained using (11).

*2.5. Triplet Formation.* Getting the set of CVI $\text{Set}^{\text{C.V.I.}}_{(x_i^a, x_i^b)}$ and NGS $\text{Set}^{\text{Ng}}_{(x_i^a, x_i^b)}$ for all $N'_k$ persons in modal $k$ we will now generate triplet samples to learn metric $M_k$. Since the positive samples for each person $x_i$ are too scarce compared to the number of negative samples, therefore, following the protocol of data augmentation in [49] we augment each person pair five times. Similarly, following the protocol in [39] we generate 20 triplets for each positive pair. Now, the triplet samples $T_i^{\text{imp}}$ and $T_i^{\text{Ng}}$ for person $x_i$ using impostor $p$ and negative Gallery $q$ are given as

$$\begin{aligned} T_i^{\text{imp}} &= \left[\left\langle x_i^a, x_i^b, p\right\rangle\right], \\ T_i^{\text{Ng}} &= \left[\left\langle x_i^a, x_i^b, q\right\rangle\right], \end{aligned} \tag{12}$$

where $p$ and $q$ are taken from respective sets $\text{Set}^{\text{C.V.I.}}_{(x_i^a, x_i^b)}$ and $\text{Set}^{\text{Ng}}_{(x_i^a, x_i^b)}$ of person $x_i$.

*2.6. Impostor Resilient Multimodal Metric (IRM3).* Taking triplets from $T_i^{\text{imp}}$ and $T_i^{\text{Ng}}$, metric IRM3 for modal $k$ is learned using MK-LFDA [44]; however, to save both the computational time and memory requirements we adapted

[44] and use three RBF kernels and one $\chi^2$ kernel. The weights for these kernels are learned globally for once for each dataset in our work using the similar method in [44]. The reason to learn weights globally is to save both time and computational burden. Further, there is considerably minor effect on kernel weights; even the weights are learned globally. This is due to the fact that the global space is comprised of all the existing modals, and thus, all the modals contribute in learning the global weights. For learning weights of kernels all the extracted features are used individually, and the dimensions of these features are also individually reduced to 450 by CCA before learning weights. In all our experiments the obtained weights for VIPeR are 0.3, 0.22, and 0.22 for RBF kernels, while weight for $\chi^2$ kernel is 0.26. For CUHK01 and CUHK03 the obtained weights for RBF kernels are 0.28, 0.24, and 0.24, while weight for $\chi^2$ kernel is 0.24. The $\sigma$ values in all the datasets for the three RBF kernels are set to the mean value of modal $k$, as well as (mean value + mean/2) and (mean value − mean/2). These values for $\sigma$ are chosen to model all the different variations in the modal $k$, while the $\sigma$ value for $\chi^2$ kernel is also set to mean value of modal $k$. The mean value in our work is the similarity value between Probe and Gallery samples of center $m_k$. Finally, the metric $M_k$ is learned as

$$\max_{M_k} \operatorname{tr}\left(\frac{M_k^T S_B M_k}{M_k^T S_W M_k}\right), \tag{13}$$

where matrices $S_B$ and $S_W$ are obtained with similar method in [44]. Now, (13) is then solved using generalized eigenvalue problem [50] in (14) to obtain first $r' = 300$ eigenvectors corresponding to eigenvalues with largest magnitude as

$$S_B\varphi = \lambda S_W\varphi. \tag{14}$$

*2.7. Reidentification.* From Figure 2, reidentification between test pair $(x_i^a, x_j^b)$ is performed by first determining the transform modal the test pair belongs to using $K$-NN classifier. In $K$-NN classifier, the parameter $K$ is set to the number of modals in the image space; that is, in VIPeR the value of $K$ is set to the number of modals $k = 7$. Then, the features of $(x_i^a, x_j^b)$ are projected into the weighted multikernel space of the respective modal, followed by the respective modal metric $M_k$ to perform matching as

$$d_{(x_i^a, x_j^b)} = \left(x_i^a - x_j^b\right)^T M_k \left(x_i^a - x_j^b\right)^T. \tag{15}$$

## 3. Experiments

Our IRM3 metric is evaluated on three benchmark datasets: VIPeR, CUHK01, and CUHK03. We follow the evaluation protocol of [33] for test/train split for VIPeR, CUHK01, and CUHK03 datasets. However, in our work we have tested CUHK01 for $P = 486$ only, while CUHK03 is tested for both *Labelled* and *Detected* settings. All the experiments are conducted in *single-shot* mode, and all the reported Cumulative Matching Curves (CMC) are obtained by averaging the results over 20 trials.

*3.1. Experiment Protocols.* To thoroughly analyze the performance of IRM3 we have devised three evaluation strategies. These strategies evaluate IRM3 performance with different number of discovered modals $K$ in $X$, with *Gallery view* impostors (GVI) (GVI are the impostors from Gallery view only and are obtained in similar way as in previous conventional metrics [14, 20–22]), as well as *Cross views* impostors (CVI).

(i) IRM3 only: it is basic multimodal metric, learned with only *Negative Gallery Samples* (NGS).

(ii) IRM3 + GVI ($p'$): IRM3 is learned with impostors from *Gallery view* (GVI), as well as with NGS Here $p'$ refers to the number of impostors taken from *Gallery* view to form triplet samples and have values $p' = 5$, 10, and 15, while the remaining triplets are formed using NGS

(iii) IRM3 + CVI ($p'$): IRM3 is learned with CVI, as well as with NGS Here $p'$ refers to number of CVI samples used to form triplets and have values $p' = 5$, 10, and 15, while the remaining triplets are formed using NGS

All the samples from NGS, GVI, and CVI contain most difficult instances for a person and are randomly sampled offline, before training metric. In all the three strategies above, we have partitioned image space into $k = 3$, 5, and 7 for VIPeR, while, for CUHK01 we have used $k = 6$, 7, and 10 partitions, and for CUHK03 $k = 13$, 14, and 16 partitions are used, respectively.

*3.2. Results on VIPeR*

*Comparison with State-of-the-Art Features.* Results of IRM3 metric are compared with three state-of-the-art features LOMO [11], GoG [25], and $\text{mom}_f^{\text{LE}}$ [24] in Table 1. All the results in Table 1 are obtained for $K = 7$ modals, and our IRM3 + CVI ($p' = 15$) has attained rank@1 **52.81%** and has outperformed all the three features of reidentification, providing evidence that if the metric can address multimodal transform variations well as well as have strong resistance against impostors then the matching accuracy can be improved. Our learned IRM3 + CVI ($p' = 15$) considers optimizing all the rank orders simultaneously and, thus, has large improvement at rank@5 and rank@10.

*Comparison with Metric Learning.* We also compared metric IRM3 with 7 metrics. From Table 1 IRM3 + CVI ($p' = 15$) has outperformed both multimodal metric LAFT [23] and impostor resistance metric LISTEN [21]. The prime difference between IRM3 and [21, 23] is its capability of addressing both the person modal transform, as well as capability of further maximizing the matching against joint constraint of cross views impostors. All these are the causes of great challenge in matching pedestrians. In Table 1 only SS-SVM [16] is a metric that tries to model the transform modal for each individual person; however, it never paid attention to acquire resistance against impostors and thus has **19.21%** lower rank@1 accuracy than IRM3 + CVI ($p' = 15$). Though IRM3 has successful results, still it has **1.36%** lower rank@1 than SCSP [38].

Table 1: Top matching comparison on VIPeR.

| | | *Single-shot, P = 316* | | |
|---|---|---|---|---|
| | *Method* | $r = 1$ | $r = 5$ | $r = 10$ |
| F | LOMO [11] | 40.0 | - | 80.51 |
| | $\text{moM}_f^{\text{LE}}$ [24] | 48.0 | 76.8 | 85.4 |
| | GoG [25] | 49.7 | 79.7 | 88.7 |
| DF | SIR-CIR [26] | 35.76 | 68.38 | 82.9 |
| DMN | GS-CNN [27] | 37.8 | 66.9 | 77.4 |
| | DGD [28] | 38.6 | - | - |
| | LSTM [29] | 42.4 | 68.7 | 79.4 |
| | MuDeep [30] | 43.03 | 74.36 | 85.76 |
| | E2E-CAN [31] | 47.2 | 79.2 | 89.2 |
| | DLPA [32] | 48.7 | 74.7 | 85.1 |
| | Quadruplet-net [33] | 49.05 | 73.10 | 81.96 |
| | JLML [34] | 50.2 | 74.2 | 84.3 |
| M | ITL [35] | 15.2 | 34.2 | 45.9 |
| | LAFT [23] | 29.6 | - | 69.3 |
| | WARCA [36] | 37.47 | 70.78 | - |
| | LISTEN [21] | 39.62 | 69.97 | 82.9 |
| | L-1 graph [37] | 41.5 | - | - |
| | SS-SVM [16] | 42.66 | 70.1 | 84.27 |
| | SCSP [38] | **53.54** | 82.59 | 91.49 |
| | **IRM3 only** | 45.92 | 82.90 | 91.63 |
| | **IRM3 + GVI** ($p' = 15$) | 50.39 | **85.79** | **95.73** |
| | **IRM3 + CVI** ($p' = 15$) | 52.81 | **87.95** | **97.29** |

Obviously, VIPeR has large pose, misalignment, and body parts displacement issues which are specifically not addressed in our work and, thus, is necessarily needed to improve the matching and results largely.

*Comparison with Deep Methods*. Though, deep features (DF) and deep matching networks (DMN) have no match with conventional metric learning methods, however, from the results in Table 1 it is clearly evident if two major issues of reidentification (i.e., multimodal transforms, and strong rejection capability against impostors) can be well handled simultaneously, then comparable or even higher performance than deep methods can be attained. Our IRM3 + CVI ($p'$ = 15) has **7.1%** and **4.94%** higher rank@1 than Quadruplet-Net [33] and JLML [34], respectively. These obtained results demonstrate the fact that for smaller dataset like VIPeR deep matching networks have insufficient training samples to learn a discriminative network.

At last, Figure 3 shows the comparison of retrieval results of two queries from VIPeR dataset for XQDA [11] and our IRM3 + CVI ($p' = 15$) when $K = 7$ modals are used. Retrieval results of *Query 1* for XQDA find the correct match at *rank = 4* enclosed in green rectangle (b), while IMR3 finds the match at *rank = 2* enclosed in green rectangle (e). Similarly, for *Query 2* our IMR3 finds the match at *rank = 1* enclosed in green rectangle (j); in contrast, XQDA finds the correct match at *rank = 3* enclosed in green rectangle (h). Thus, our IRM3 approach improves matching, and consequently rank gets higher.

### 3.3. Results on CUHK01

*Comparison with State-of-the-Art Features*. Table 2 summarizes results of IRM3 for $K = 10$ modals and compares the obtained results with LOMO [11], GoG [25], and $\text{mom}_f^{\text{LE}}$ [24]. Though the three features are discriminative, however, our IRM3 approach is better than the three features in solving the two big challenges of Re-ID, that is, multimodal pedestrians matching and impostors resistance. Since CUHK01 has larger training set than VIPeR, thus, modal transforms can be well learned, and therefore, IRM3 + CVI ($p' = 15$) attains larger discrimination than $\text{mom}_f^{\text{LE}}$ [24]. Our IRM3 + CVI ($p' = 15$) has **15.15%** higher rank@1 accuracy than $\text{mom}_f^{\text{LE}}$ due to inherent virtue of handling different modals, person specific variations, and rejecting large number of impostors, all simultaneously.

*Comparison with Metric Learning*. In Table 2 three most recently proposed metrics CVAML [40], WARCA [36], and L-1 Graph [37] are compared with our IRM3 approach. All the three metrics have assumption of unimodal intercamera transform, rather than multimodal image space. Though WARCA [36] employed hard negative samples as learning constraint, however, ignoring other negative samples from

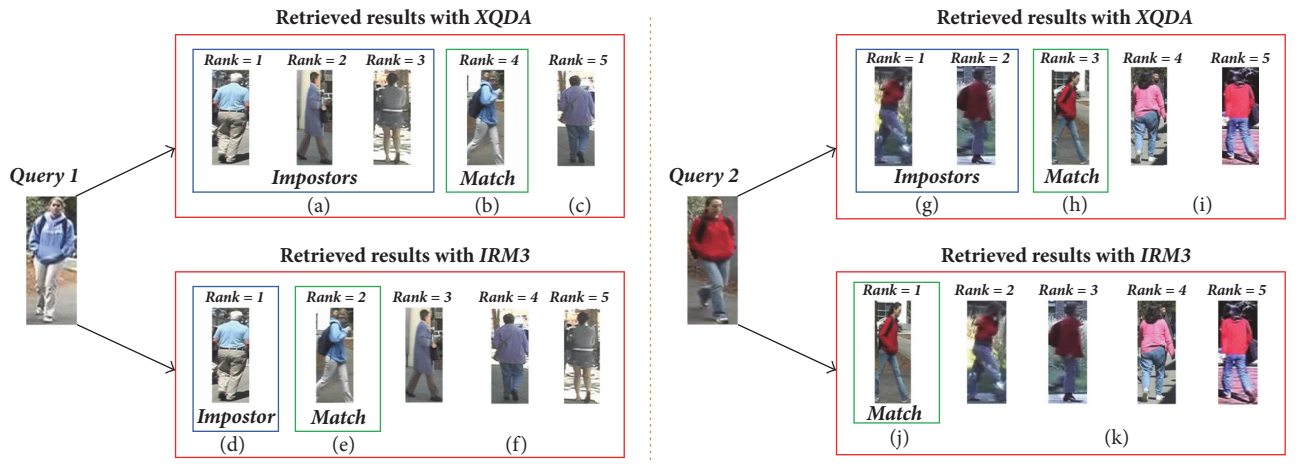



Figure 3: Two queries are shown, *Query 1* and *Query 2*, and their retrieval results using XQDA [11] and our IRM3. Correct match is shown in green rectangle, while blue rectangle shows impostors.

Table 2: Top matching comparison on CUHK01.

| | | *Single-shot, P = 486* | | |
|---|---|---|---|---|
| | *Method* | $r = 1$ | $r = 5$ | $r = 10$ |
| F | LOMO [11] | 49.2 | 75.7 | 84.2 |
| | $\text{moM}_f^{\text{LE}}$ [24] | 64.6 | 84.9 | 90.6 |
| | GoG [25] | 57.8 | 79.1 | 86.2 |
| DMN | MCP-CNN [39] | 53.7 | 84.3 | 91.0 |
| | DGD [28] | 66.6 | - | - |
| | E2E-CAN [31] | 67.2 | 87.3 | 92.5 |
| | DLPA [32] | 75.0 | 93.5 | 95.7 |
| | Quadruplet-net [33] | 62.55 | 83.44 | 89.71 |
| | JLML [34] | 69.8 | 88.4 | 93.3 |
| M | CVAML [40] | 57.3 | 81.2 | 86.5 |
| | WARCA [36] | 58.34 | 79.76 | - |
| | L-1 graph [37] | 50.1 | - | - |
| | **IRM3 only** | 68.24 | 88.71 | 94.31 |
| | **IRM3 + GVI** ($p' = 15$) | 72.91 | **94.61** | **97.6** |
| | **IRM3 + CVI** ($p' = 15$) | **76.14** | **96.90** | **98.35** |

*Gallery view* and not taking into consideration a person modal during learning have made it suffer greatly to attain higher accuracy. On the other hand, IRM3 + CVI ($p' = 15$) has capability to deal all these challenges and, thus, has attained **76.14%** rank@1 accuracy.

*Comparison with Deep Methods*. In Table 2, we can see several deep matching networks (DMN) have performed much well than conventional metrics on CUHK01. Only K-LFDA when trained with $\text{mom}_f^{\text{LE}}$ [24] feature attains comparable performance than DMN. However, motivated to resolve the challenges for reidentification in real world (i.e., multimodal image space, and diverse impostors) IRM3 + CVI ($p' = 15$) has much better results than MCP-CNN [39], E2E-CAN [31], Quadruplet-Net [33], and JLML [34], while our IRM3 + CVI ($p' = 15$) has **1.49%** higher rank@1 than DLPA [32]. DLPA extracts deep features by semantically aligning body parts, as well as rectifying pose variations. We believe if sematic body parts alignment and rectification of poses variations are included in our IRM3 then the results can be further improved.

### 3.4. Results on CUHK03

*Comparison with State-of-the-Art Features*. Table 3 compares LOMO [11] and GoG [25] features with our IRM3 metric in both *Labelled* and *Detected* settings. All the results in Table 3 are obtained for $K = 16$ modals. In Table 3, obtained results are much higher than the two features. The primary reason of gain in performance for IRM3 against the features [11, 25] is mainly due to the difference in their approaches. In [11, 25] a universal feature representation is proposed for

Table 3: Top matching comparison on CUHK03.

| | *Labelled, P = 100* | | | |
|---|---|---|---|---|
| | *Method* | $r = 1$ | $r = 5$ | $r = 10$ |
| F | LOMO [11] | 52.2 | 82.23 | 92.14 |
| | GoG [25] | 67.3 | 91.0 | 96.0 |
| DF | DCAF [41] | 74.21 | 94.33 | 97.54 |
| DFM | DGD [28] | 75.3 | - | - |
| | Quadruplet-net [33] | 75.53 | 95.15 | 99.16 |
| | MuDeep [30] | 76.87 | 96.12 | 98.41 |
| | E2E-CAN [31] | 77.6 | 95.2 | 99.3 |
| | JLML [34] | 83.2 | 98.0 | 99.4 |
| | DLPA [32] | 85.4 | 97.6 | 99.4 |
| M | SS-SVM [16] | 57.0 | 85.7 | 94.3 |
| | Null Sp. [42] | 62.55 | 90.05 | 94.80 |
| | SSM [43] | 76.6 | 94.6 | 98.0 |
| | WARCA [36] | 78.38 | 94.55 | - |
| | **IRM3 only** | 78.83 | 95.97 | 98.37 |
| | **IRM3 + GVI** ($p' = 15$) | 83.32 | **98.70** | **99.54** |
| | **IRM3 + CVI** ($p' = 15$) | **86.17** | **99.02** | **99.68** |
| | *Detected, P = 100* | | | |
| | *Method* | $r = 1$ | $r = 5$ | $r = 10$ |
| F | LOMO [11] | 46.25 | 78.9 | 88.55 |
| | GoG [25] | 65.5 | 88.4 | 93.7 |
| DF | SIR-CIR [26] | 52.17 | 83.7 | 90.4 |
| | DCAF [41] | 67.99 | 91.04 | 95.36 |
| DFM | LSTM [29] | 57.3 | 80.10 | 88.3 |
| | GS-CNN [27] | 68.1 | 88.1 | 94.6 |
| | E2E-CAN [31] | 69.2 | 88.5 | 94.1 |
| | MuDeep [30] | 75.64 | 94.36 | 97.46 |
| | JLML [34] | 80.6 | 96.9 | 98.7 |
| | DLPA [32] | **81.6** | **97.3** | **98.4** |
| M | L-1 graph [37] | 39.0 | - | - |
| | SS-SVM [16] | 51.2 | 80.8 | 89.6 |
| | Null Sp. [42] | 54.70 | 84.75 | 94.80 |
| | SSM [43] | 72.7 | 92.4 | 96.1 |
| | **IRM3 only** | 72.98 | 91.7 | 93.02 |
| | **IRM3 + GVI** ($p' = 15$) | 78.68 | 95.60 | 98.09 |
| | **IRM3 + CVI** ($p' = 15$) | 80.77 | 96.94 | 98.67 |

all the different persons, which may not be optimal for all the persons at the same time residing on different modals; in contrast, our motivation is based on discovering distinct modals in the image space and then addressing each modal specifically with empowerment of large number of impostors rejection. Therefore, our IRM3 + CVI ($p' = 15$) (in Labelled setting) has rank@1 accuracy of about **86.17%**.

*Comparison with Metric Learning.* In Table 3, recently proposed WARCA [36] and SSM [43] are compared with our IRM3 approach. WARCA [36] differs with our IRM3 approach in a way that it only addresses hard negative samples, while SSM [43] differs in a way that it has no measure to account for different modal transforms, as well as having no resistance against impostors. Our IRM3 + CVI ($p' = 15$) (in Labelled setting) has surpassed [36] and [43] and has attained **9.04%** and **11.1%** rise at rank@1 accuracy, respectively.

*Comparison with Deep Methods.* Interestingly, in Table 3 all the deep methods in *Labelled* and *Detected* settings have very high performance on CUHK03. These high results demonstrate the fact that CUHK03 is the largest dataset among all and, thus, can help in learning a more discriminative DMN. Even though both JLML [34] and DLPA [32] learn deep body features with global and local body parts alignment, as well as, pose alignment, however, our IMR3 approach benefitted with transform specific metrics empowered with impostors rejection still maintained to attain better results. Our IRM3

Table 4: Effect of multimodal transforms + impostor resistance (VIPeR, $P = 316$).

| Method | $r = 1$ | $r = 5$ | $r = 10$ |
|---|---|---|---|
| $k = 5$, $p' = 0$ | 45.27 | 82.16 | 91.1 |
| $k = 7$, $p' = 0$ | **45.92** | 82.90 | 91.63 |
| $k = 5$, GVI ($p' = 15$) | 48.88 | 85.33 | 95.37 |
| $k = 7$, GVI ($p' = 15$) | **50.39** | 85.79 | 95.73 |
| $k = 5$, CVI ($p' = 15$) | 52.10 | 87.14 | 96.87 |
| $k = 7$, CVI ($p' = 15$) | **52.81** | 87.95 | 97.29 |

considers optimizing all the rank orders simultaneously and, thus, have large gain at rank@5 and rank@10 in *Labelled* setting.

*3.5. Analysis.* In Table 4, we analyzed the effect of number of modals $K$ in testing for VIPeR. Initially, we have partitioned image space into $K = 5$ and then tested it without using any impostor sample ($K = 5$, $p' = 0$) to obtain rank@1 results of about 45.27%. As the more modals are discovered in the image space, such as $K = 7$, then the results get further improved even without using any impostor sample ($K = 7$, $p' = 0$), and rank@1 becomes 45.92%. The main reason behind this increment is the fact that now we could match more test samples correctly by using their actual modal transforms which were lost when the modals are less discovered in $K = 5$.

In addition, we could also see a positive increment in results when impostors from *Gallery* view are also added in learning metric. Both ($K = 5$, GVI ($p' = 15$)) and ($K = 7$, GVI ($p' = 15$)) have attained more higher differentiating capability than [14, 20–22], as now, they can restrict impostors by taking into care transform modals a positive pair and impostors undergo.

Interestingly, in our work this impostor resistance can be further enhanced. This is done by using *Cross views impostors* (CVI). From Table 4, it is clear that even for same number of modals say $K = 7$, when (CVI) are used then the differentiation capability of ($K = 7$, CVI ($p' = 15$)) gets further enhanced than ($K = 7$, GVI ($p' = 15$)) and rank@1 becomes 52.81%. This increment in rank@1 provides a strong evidence that CVI have ability to maximize the similarity of positive pair more than GVI by taking into care both the transform modal, as well as various different changes a given query and Gallery samples undergo in different views.

At last, in Figure 4 we have provided a performance comparison at rank@1 when the modal centers are chosen randomly, as well as when the centers are obtained using our method in Section 2.2. Obtained rank@1 accuracy for random centers is poor, because these random centers are obtained just by simply choosing the top-$K$ persons without taking into care their reliability, stability, and IDs.

*3.6. Efficiency.* We computed the run time of our IRM3 approach using MK-LFDA [44], XQDA [11], and K-LFDA [45] (with $\chi^2$ kernel) on CUHK03. There are 1260 training persons and 100 testing identities. All the algorithms are implemented in MATLAB and run on server machine having 6 CPUs (Xeon(r)e5-2620) with each CPU having 6 cores and

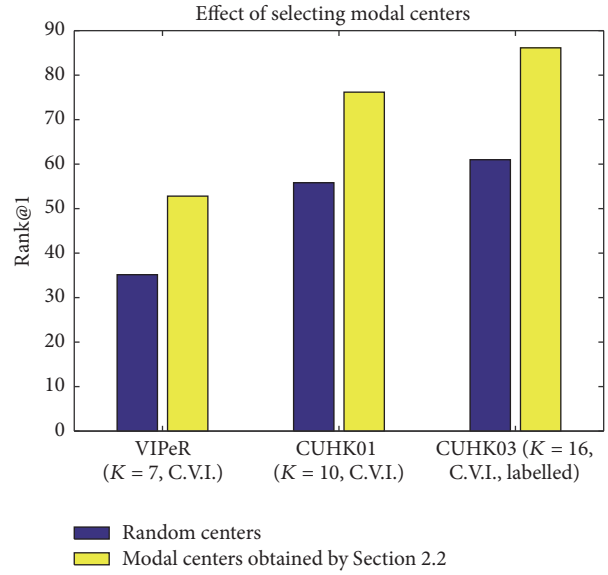



Figure 4: Performance at rank@1 when centers $m_k$ are selected randomly, and when centers are selected with our approach provided in Section 2.2.

Table 5: Run time comparison on CUHK03 (in seconds).

| Method | MK-LFDA [44] | XQDA [11] | K-LFDA [45] |
|---|---|---|---|
| Training | 171.26 | 174.7 | 160.03 |
| Testing | 28.8 | 30.23 | 32.45 |

total memory size of 256 GB. In Table 5, training time of MK-LFDA [44] is faster than XQDA [11] but lower than K-LFDA [45]. However, in testing when the weights of kernels are not learned MK-LFDA [44] is faster than both XQDA and K-LFDA. These timing results support the fact that our proposed method is well applicable in real time applications and in public spaces.

## 4. Conclusion

This paper presents a metric learning approach that exploits both multimodal transforms and Cross views impostors to improve the capability of metric to differentiate among different persons, as well as enhance rejection capability to decline large number of real world diverse impostors. In real world mostly pedestrian images are multimodal, and in public

spaces several persons share similar clothing; therefore, our IRM3 is learned to tackle such issues of reidentification and person tracking in public spaces. Extensive experiments on three challenging datasets (VIPeR, CUHK01, and CUHK03) demonstrate the effectiveness of our IRM3 metric which has outperformed many previous state-of-the-art metrics. In addition, we further intend to extend our approach for testing in real world scenario and intend to solve various other issues for real time implementation.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References


[1] W.-S. Zheng, S. Gong, and T. Xiang, "Associating groups of people," in *Proceedings of the 2009 20th British Machine Vision Conference, BMVC 2009*, UK, September 2009.

[2] R. Zhao, W. Oyang, and X. Wang, "Person Re-Identification by Saliency Learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 2, pp. 356–370, 2017.

[3] S. Bąk, S. Zaidenberg, B. Boulay, and F. Brémond, "Improving person re-identification by viewpoint cues," in *Proceedings of the 11th IEEE International Conference on Advanced Video and Signal-Based Surveillance, AVSS 2014*, pp. 175–180, Republic of Korea, August 2014.

[4] R. Rama Varior, G. Wang, J. Lu, and T. Liu, "Learning invariant color features for person reidentification," *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 3395–3410, 2016.

[5] C.-H. Kuo, S. Khamis, and V. Shet, "Person re-identification using semantic color names and RankBoost," in *Proceedings of the 2013 IEEE Workshop on Applications of Computer Vision, WACV 2013*, pp. 281–287, January 2013.

[6] Y. Hu, S. Liao, Z. Lei, D. Yi, and S. Z. Li, "Exploring structural information and fusing multiple features for person re-identification," in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPRW 2013*, pp. 794–799, June 2013.

[7] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 2360–2367, IEEE, San Francisco, Ca, USA, June 2010.

[8] A. Bhuiyan, A. Perina, and V. Murino, "Person re-identification by discriminatively selecting parts and features," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 8927, pp. 147–161, 2015.

[9] S. Khamis, C.-H. Kuo, V. K. Singh, V. D. Shet, and L. S. Davis, "Joint learning for attribute-consistent person re-identification," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 8927, pp. 134–146, 2015.

[10] J. Roth and X. Liu, "On the exploration of joint attribute learning for person re-identification," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 9003, pp. 673–688, 2015.

[11] S. Liao, Y. Hu, X. Zhu, and S. Z. Li, "Person re-identification by Local Maximal Occurrence representation and metric learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 2197–2206, June 2015.

[12] Y. Yang, J. Yang, J. Yan, S. Liao, D. Yi, and S. Z. Li, "Salient color names for person re-identification," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 8689, no. 1, pp. 536–551, 2014.

[13] Z. Mingyong, Z. Wu, C. Tian, Z. Lei, and H. Lei, "Efficient person re-identification by hybrid spatiogram and covariance descriptor," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPRW 2015*, pp. 48–56, June 2015.

[14] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *Journal of Machine Learning Research*, vol. 10, pp. 207–244, 2009.

[15] M. Kostinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof, "Large scale metric learning from equivalence constraints," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 2288–2295, IEEE, Providence, RI, USA, June 2012.

[16] Y. Zhang, B. Li, H. Lu, A. Irie, and X. Ruan, "Sample-specific SVM learning for person re-identification," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 1278–1287, July 2016.

[17] H. Shi, Y. Yang, X. Zhu et al., "Embedding deep metric for person Re-identification: A study against large variations," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 9905, pp. 732–748, 2016.

[18] Y.-C. Chen, W.-S. Zheng, J.-H. Lai, and P. C. Yuen, "An asymmetric distance model for cross-view feature mapping in person reidentification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 8, pp. 1661–1675, 2017.

[19] D. Chen, Z. Yuan, G. Hua, N. Zheng, and J. Wang, "Similarity learning on an explicit polynomial kernel feature map for person re-identification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pp. 1565–1573, June 2015.

[20] M. Hirzer, P. M. Roth, and H. Bischof, "Person re-identification by efficient impostor-based metric learning," in *Proceedings of the 2012 IEEE 9th International Conference on Advanced Video and Signal-Based Surveillance, AVSS 2012*, pp. 203–208, China, September 2012.

[21] X. Zhu, X.-Y. Jing, F. Wu et al., "Distance learning by treating negative samples differently and exploiting impostors with symmetric triplet constraint for person re-identification," in *Proceedings of the 2016 IEEE International Conference on Multimedia and Expo, ICME 2016*, July 2016.

[22] M. Dikmen, E. Akbas, T. S. Huang, and N. Ahuja, "Pedestrian recognition with a learned metric," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 6495, no. 4, pp. 501–512, 2011.

[23] W. Li and X. Wang, "Locally aligned feature transforms across views," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2013*, pp. 3594–3601, June 2013.

[24] M. Gou, O. Camps, and M. Sznaier, "moM: Mean of Moments Feature for Person Re-identification," in *Proceedings of the 2017*

*IEEE International Conference on Computer Vision Workshop (ICCVW)*, pp. 1294–1303, Venice, Italy, October 2017.

[25] T. Matsukawa, T. Okabe, E. Suzuki, and Y. Sato, "Hierarchical Gaussian descriptor for person re-identification," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 1363–1372, July 2016.

[26] F. Wang, W. Zuo, L. Lin, D. Zhang, and L. Zhang, "Joint learning of single-image and cross-image representations for person re-identification," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 1288–1296, July 2016.

[27] R. R. Varior, M. Haloi, and G. Wang, "Gated siamese convolutional neural network architecture for human re-identification," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 9912, pp. 791–808, 2016.

[28] T. Xiao, H. Li, W. Ouyang, and X. Wang, "Learning deep feature representations with Domain Guided Dropout for person re-identification," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 1249–1258, July 2016.

[29] R. R. Varior, B. Shuai, J. Lu, D. Xu, and G. Wang, "A siamese long short-term memory architecture for human re-identification," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 9911, pp. 135–153, 2016.

[30] X. Qian, Y. Fu, Y. Jiang, T. Xiang, and X. Xue, "Multi-scale Deep Learning Architectures for Person Re-identification," in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5409–5418, Venice, October 2017.

[31] H. Liu, J. Feng, M. Qi, J. Jiang, and S. Yan, "End-to-end comparative attention networks for person re-identification," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3492–3506, 2017.

[32] L. Zhao, X. Li, Y. Zhuang, and J. Wang, "Deeply-Learned Part-Aligned Representations for Person Re-identification," in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 3239–3248, Venice, October 2017.

[33] W. Chen, X. Chen, J. Zhang, and K. Huang, "Beyond Triplet Loss: A Deep Quadruplet Network for Person Re-identification," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1320–1329, Honolulu, HI, July 2017.

[34] W. Li, X. Zhu, and S. Gong, "Person Re-Identification by Deep Joint Learning of Multi-Loss Classification," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, pp. 2194–2200, Melbourne, Australia, August 2017.

[35] W. Liao, M. Y. Yang, N. Zhan, and B. Rosenhahn, "Triplet-Based Deep Similarity Learning for Person Re-Identification," in *Proceedings of the 2017 IEEE International Conference on Computer Vision Workshop (ICCVW)*, pp. 385–393, Venice, October 2017.

[36] C. Jose and F. Fleuret, "Scalable metric learning via weighted approximate rank component analysis," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 9909, pp. 875–890, 2016.

[37] E. Kodirov, T. Xiang, Z. Fu, and S. Gong, "Person re-identification by unsupervised $\ell 1$ graph learning," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 9905, pp. 178–195, 2016.

[38] D. Chen, Z. Yuan, B. Chen, and N. Zheng, "Similarity learning with spatial constraints for person re-identification," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 1268–1277, July 2016.

[39] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng, "Person re-identification by multi-channel parts-based CNN with improved triplet loss function," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 1335–1344, July 2016.

[40] H. Yu, A. Wu, and W. Zheng, "Cross-View Asymmetric Metric Learning for Unsupervised Person Re-Identification," in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 994–1002, Venice, October 2017.

[41] D. Li, X. Chen, Z. Zhang, and K. Huang, "Learning Deep Context-Aware Features over Body and Latent Parts for Person Re-identification," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7398–7407, Honolulu, HI, July 2017.

[42] L. Zhang, T. Xiang, and S. Gong, "Learning a discriminative null space for person re-identification," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016*, pp. 1239–1248, July 2016.

[43] S. Bai, X. Bai, and Q. Tian, "Scalable Person Re-identification on Supervised Smoothed Manifold," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3356–3365, Honolulu, HI, July 2017.

[44] M. A. Syed and J. Jiao, "Multi-kernel metric learning for person re-identification," in *Proceedings of the 23rd IEEE International Conference on Image Processing, ICIP 2016*, pp. 784–788, September 2016.

[45] F. Xiong, M. Gou, O. Camps, and M. Sznaier, "Person re-identification using kernel-based metric learning methods," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Preface*, vol. 8695, no. 7, pp. 1–16, 2014.

[46] R. Zhao, W. Ouyang, and X. Wang, "Unsupervised salience learning for person re-identification," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '13)*, pp. 3586–3593, IEEE, Portland, Ore, USA, June 2013.

[47] G. Lisanti, I. Masi, and A. Del Bimbo, "Matching people across camera views using kernel canonical correlation analysis," in *Proceedings of the 8th ACM/IEEE International Conference on Distributed Smart Cameras, ICDSC 2014*, Italy, November 2014.

[48] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical correlation analysis: an overview with application to learning methods," *Neural Computation*, vol. 16, no. 12, pp. 2639–2664, 2004.

[49] W. Li, R. Zhao, T. Xiao, and X. Wang, "DeepReID: Deep filter pairing neural network for person re-identification," in *Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014*, pp. 152–159, June 2014.

[50] Y. Ying and P. Li, "Distance metric learning with eigenvalue optimization," *Journal of Machine Learning Research*, vol. 13, pp. 1–26, 2012.