

Research Article

Automatic Detection of Concrete Spalling Using Piecewise Linear Stochastic Gradient Descent Logistic Regression and Image Texture Analysis

Nhat-Duc Hoang ¹, Quoc-Lam Nguyen,² and Xuan-Linh Tran ¹

¹Faculty of Civil Engineering, Institute of Research and Development, Duy Tan University, P809 - 03 Quang Trung, Danang, Vietnam

²Faculty of Civil Engineering, Duy Tan University, P202 - 03 Quang Trung, Danang 550000, Vietnam

Correspondence should be addressed to Xuan-Linh Tran; tranxuanlinh@dtu.edu.vn

Received 18 April 2019; Revised 10 June 2019; Accepted 26 June 2019; Published 16 July 2019

Guest Editor: Murari Andrea

Copyright © 2019 Nhat-Duc Hoang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Recognition of spalling on surface of concrete wall is crucial in building condition survey. Early detection of this form of defect can help to develop cost-effective rehabilitation methods for maintenance agencies. This study develops a method for automatic detection of spalled areas. The proposed approach includes image texture computation for image feature extraction and a piecewise linear stochastic gradient descent logistic regression (PL-SGD LR) used for pattern recognition. Image texture obtained from statistical properties of color channels, gray-level cooccurrence matrix, and gray-level run lengths is used as features to characterize surface condition of concrete wall. Based on these extracted features, PL-SGD LR is employed to categorize image samples into two classes of “nonspall” (negative class) and “spall” (positive class). Notably, PL-SGD LR is an extension of the standard logistic regression within which a linear decision surface is replaced by a piecewise linear one. This improvement can enhance the capability of logistic regression in dealing with spall detection as a complex pattern classification problem. Experiments with 1240 collected image samples show that PL-SGD LR can help to deliver a good detection accuracy (classification accuracy rate = 90.24%). To ease the model implementation, the PL-SGD LR program has been developed and compiled in MATLAB and Visual C#.NET. Thus, the proposed PL-SGD LR can be an effective tool for maintenance agencies during periodic survey of buildings.

1. Introduction

In the maintenance process of high-rise buildings, it is important to identify surface defects to ensure the serviceability of structures. The reason is that surface quality strongly affects the safety as well as esthetics of buildings. After buildings are delivered to clients, their conditions quickly deteriorate due to the combined influences of inclement weather conditions, occupants' activities, and structural aging [1]. The deterioration of buildings is usually reflected in forms of cracks and spalls. These forms of damage do not only bring about inconvenience to occupants but also degrade the structural integrity [2]. If periodic maintenance processes cannot detect and handle this damage timely, building's owners may suffer from financial losses due to degradation of asset value. Hence, correct detection of surface damage is a crucial task in building condition assessment.

Spalling (see Figure 1) happens when fragments of materials (e.g., concrete, mortar) are ejected from the surface structure because of impact or internal stress. Spalling occurs in concrete because of moisture incursion into structural elements. Spalls are commonly observed in various structural elements in buildings such as wall, beam, column, ceiling, and floor. Particularly for reinforced concrete structures, spalls are indicators of oxidation or corrosion of reinforcing steel [3–5]. Therefore, if this form of defect is unidentified and untreated during the building maintenance process, the problem of corroded reinforcements may quickly expand and significantly worsen the structural durability.

In Vietnam, periodic building maintenance is usually performed by human technicians. This practice is also common in other countries due to the fact that human inspection can help to obtain a high level of accuracy and directly

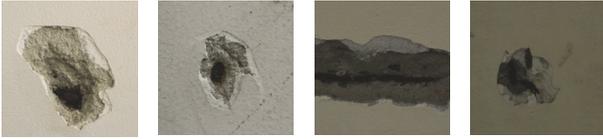


FIGURE 1: Spalls on concrete surface.

point out the problems underlying the detected defects [6]. Nevertheless, manual survey is notoriously known to be time consuming and labor intensive [3]. Moreover, the quality of building assessment is also strongly dependent on the skill, experience, and subjective judgment of human inspectors. This fact may lead to variability of building assessment results. In addition, the processes of data collection by means of measurement, data processing, and report are extremely time consuming for high-rise buildings having a huge number of structural elements needed to be surveyed periodically. Therefore, it is beneficial for maintenance agencies to be equipped with a more productive method of collecting and processing building condition data.

Among automated methods for building condition assessment, image processing based methods have been extensively used because of a quick advancement in the field and affordable cost of digital cameras [7–9]. Using image process techniques, regions of image samples suffering from spalling can be distinguished and isolated from healthy regions based on the features extracted from these regions. Essentially, spalling belongs to the category of area based defect. This means that using information of one pixel cannot be sufficient for spall detection. Hence, characteristics of an image region need to be extracted and analyzed for recognizing spall defect.

Nevertheless, recognition of area based defect on surface of structures is a challenging task. It is due to various difficulties including diversified textures of concrete surface, uneven illumination, irregular textures caused by stains, etc. Therefore, in recent years, a considerable number of research works have been dedicated to automatic detection of area based distress including spalling. Suwwanakarn et al. [10] proposed the employment of three circular filters to detect air pockets on the surfaces of concrete. Koch and Brilakis [11] put forward a pothole detection method which employs image thresholding, morphological thinning, elliptic regression, and texture extraction.

German et al. [4] extracted major properties of spalled regions on concrete columns for postearthquake safety assessment of buildings; this study involves the technique of image thresholding by means of a local entropy-based algorithm and a global adaptive thresholding approach. Subsequently, template matching and morphological operations can be performed to identify damaged regions [4]. A multispectral image analysis approach was presented by Valença et al. [12] to evaluate concrete damage and delineate the deteriorated zones in an automatic manner.

Kim et al. [13] established a framework to assess the dimensional and surface quality of precast concrete elements on the basis of BIM and 3D laser scanning. Paal

et al. [14] present a computer vision based method for detecting building's columns and retrieving their properties used for damage recognition. A technique for localizing and quantifying spalling defects on concrete surfaces has been put forward by the employment of a terrestrial laser scanner [5]. Li et al. [15] investigate the feasibility of an integrated framework for the detection and measurement of potholes on the basis of 2D images and Ground Penetrating Radar data. Konishi et al. [16] detect void in subway tunnel lining using thermal image photographs and signal analyses. Dawood et al. [3] develop an integrated model based on image processing techniques and machine learning algorithms for spalling detection used in condition survey of subway networks; the image processing technique includes various methods of image smoothing, thresholding, histogram equalization, and filtering. Hoang [2] relied on steerable filters and machine learning for recognizing defects appearing on wall surface. Oliveira Santos et al. [17] and Santos et al. [18] put forward hyperspectral image processing models to detect cracking patterns both on clean and on concrete surface with biological stains. Recent research works [19–23] have pointed out an increasing trend of applying computer vision in structural health inspection.

Since the spalled areas and healthy ones have distinctive texture properties, texture of image samples can be computed and employed for spall recognition. Image texture expresses the spatial arrangement of color or intensities in an image sample [24]. Therefore, image texture computation methods such as statistical measurements of color channels (e.g., mean, standard deviation, skewness, etc.) [25], gray-level cooccurrence matrix [26], and gray-level run lengths [27] can be potentially applied for spall detection. Based on the image texture based features, machine learning approaches can be employed for classifying data instances into categories of spall (positive class) and nonspall (negative class). Nevertheless, few studies have investigated the efficiency of the aforementioned image texture computation in spall recognition. Moreover, it is evident that the combination of image processing and machine learning can potentially lead to effective solutions for structure health monitoring [28–34]. However, models that hybridize the strengths of image processing and machine learning based classifiers have rarely been employed for spall detection. Therefore, the current study is an attempt to fill these gaps in the literature.

Furthermore, the problem of spall detection can be formulated as a two-class pattern recognition; it is able to model the target output as a binary response variable with “nonspall” = 0 and “spall” = 1. Hence, logistic regression (LR), which is one of the models for binary data [35], can be employed for pattern classification. Logistic regression is a simple linear classifier yet effective machine learning model which is capable of delivering probabilistic prediction outcomes. The key procedure of a LR model is to define a linear classifier (in the form of a hyperplane) and an objective function (in the form of a log likelihood function); accordingly, gradient descent algorithm can be applied to adapt the model parameters [36]. The implementation of LR

is straightforward and its successful applications have been reported in various studies [37–39].

Nevertheless, one observable limitation of LR is that its employment of a linear classifier is expressed in the form of a hyperplane. To improve the capability of LR in dealing with nonlinear data, this study investigates the feasibility of replacing the conventional linear classifier used in LR by a piecewise linear model. This modification can lead to a higher degree of flexibility of model structure and potentially bring about a better predictive accuracy. In this study, a piecewise linear LR model, named PLLR, is developed for detecting spalled regions on surface of concrete wall structures. Additionally, a sequential algorithm described in the previous work of Hoang [40] and the stochastic gradient descent algorithm [41] are used to train the PLLR model. Data set including 1240 image samples has been collected to construct and verify the proposed method. The statistical descriptions of color channels, properties of the gray-level cooccurrence matrix, and properties of the gray-level run lengths are employed to compute the texture of image samples. In addition, based on the set of image texture based features, principal component analysis is employed for dimension reduction. The performance of the proposed PLLR is benchmarked with those of stochastic gradient descent LR and backpropagation artificial neural network.

The subsequent parts of the paper are organized as follows. The second section reviews the research methodology, followed by the third section that describes the collected image data set; the fourth section describes the structure of the proposed model used for automatic recognition of concrete wall spalling. The fifth section will report experimental results; several concluding remarks of this study are provided in the final section.

2. Research Methodology

2.1. Image Texture Computation. Due to the typical texture of concrete walls, two pixels having the same color/gray level can belong to both spalled and nonspalled areas. Thus, it is infeasible to detect spall in a pixel level because information of a single pixel is not sufficient for spall recognition. As stated earlier, since spalled and nonspalled concrete wall surface have distinctive features regarding color and roughness. Hence, information of texture of an image region can be helpful for identifying spalled wall sections. Accordingly, a large surveying image can be separated into a number of nonoverlapped image samples with a fixed size (e.g. 100x100 pixels). This division can also help to expedite the texture computation process. Based on such small image samples, image textures regarding statistical measurements of color channels [25], gray-level cooccurrence matrix [26], and gray-level run lengths [27] can be computed and used for data classification.

2.1.1. Statistical Properties of Color Channels. Since concrete surface background may contain irregular objects such as paints or stains caused by corroded steel reinforcements, information regarding the color of image samples can be

helpful for the task of spall recognition. Let $P(I)$ represent the first-order histogram of an image sample S ; $P(I)$ is computed as follows [25]:

$$P_c(I) = \frac{N_{I,c}}{W \times H} \quad (1)$$

where c is a color channel (either red or green or blue). $N_{I,c}$ denotes the number of pixels having color value of the channel $c = I$. H and W represent the two parameters of the image height and width.

Accordingly, the average (μ_c) and the standard deviation (σ_c) of color value can be computed in the following manner:

$$\mu_c = \sum_{i=0}^{NL-1} I_{i,c} \times P_c(I) \quad (2)$$

$$\sigma_c = \sqrt{\sum_{i=0}^{NL-1} (I_{i,c} - \mu_c)^2 \times P_c(I)} \quad (3)$$

where $I_{i,c} = 0, 1, 2, \dots, 255$. $NL = 256$ is the number of discrete color values.

Moreover, the skewness (δ_c) and kurtosis (η_c) of discrete color values are calculated as follows:

$$\delta_c = \frac{\sum_{i=0}^{NL-1} (I_{i,c} - \mu_c)^3 \times P_c(I)}{\sigma^3} \quad (4)$$

$$\eta_c = \frac{\sum_{i=0}^{NL-1} (I_{i,c} - \mu_c)^4 \times P_c(I)}{\sigma^4} \quad (5)$$

The entropy (ρ_c) and range (Δ_c) of color intensity can be computed and characterize distinctive features of image samples. These ρ_c and Δ_c are calculated as follows:

$$\rho_c = - \sum_{i=0}^{NL-1} P_c(I) \times \log_2(P_c(I)) \quad (6)$$

$$\Delta_c = \max(I_c) - \min(I_c) \quad (7)$$

2.1.2. Gray-Level Cooccurrence Matrix (GLCM). In computer vision field, gray-level cooccurrence matrix (GLCM) [26] is a commonly employed method for texture classification. A cooccurrence matrix provides information regarding the distribution of cooccurring pixel values at a given offset r [42]. After the cooccurrence matrix is computed, it is often normalized. Subsequently, a set of statistical measures can be computed from this normalized matrix. Furthermore, it is beneficial for texture classification to detect features of an image's region which are rotationally invariant. This is the reason why the cooccurrence matrix is usually computed at different regular angles (Δ) with certain value of offset r . The commonly used values of Δ are 0° , 45° , 90° , and 135° [42].

It is proper to note that a color image sample must be converted to a gray-scale image before the computation of its GLCM. Let $\delta = (r, \Delta)$ represent a relationship employed to compute a GLCM of an image. Thus, the joint probability of the pairs of color levels that occur at the two

locations dictated by the relationship δ can be calculated [42]. The information of this joint probability is provided in a cooccurrence matrix P_δ within which $P_\delta(i, j)$ denotes the probability of the two color levels of i and j occurring at the relationship δ [43].

The normalized cooccurrence matrix is computed as follows:

$$P_\delta^N(i, j) = \frac{P_\delta(i, j)}{S_p} \quad (8)$$

where P_δ^N represents the normalized GLCM and S_p is the total number of pixels.

With $r = 1$ and $\Delta = 0^\circ, 45^\circ, 90^\circ$, and 135° , we can establish four cooccurrence matrices. Based on these four matrices, the indices of angular second moment (AM), contrast (CO), correlation (CR), and entropy (ET) can be obtained and utilized for texture classification [44, 45]. These indices are calculated by the following equations [26]:

$$AM = \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} P_\delta^N(i, j)^2 \quad (9)$$

where $N_g = 256$ denotes the number of color level values. Consider

$$CO = \sum_{k=0}^{N_g-1} k^2 \sum_{\substack{i=1 \\ |i-j|=k}}^{N_g} \sum_{j=1}^{N_g} P_\delta^N(i, j) \quad (10)$$

$$CR = \frac{\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} i \times j \times P_\delta^N(i, j) - \mu_X \mu_Y}{\sigma_X \sigma_Y} \quad (11)$$

where μ_X, μ_Y, σ_X , and σ_Y represent the means and standard deviations of the marginal distribution associated with $P_\delta^N(i, j)$ [26].

2.1.3. Gray-Level Run Lengths (GLRL). First proposed by Galloway [27], texture analysis based on gray-level run lengths is an effective method in image processing. This method is based on the observation that relatively long gray-level runs occur more often in a coarse texture and a fine texture often includes short runs [46]. Based on previous experimental works, properties of gray-level run lengths can help construct useful features for texture classification tasks [47–50]. For an image sample and in a given direction, a run-length matrix $p(i, j)$ is defined as the number of times that the image contains a run length j of gray level i [27]. Based on $p(i, j)$, various texture features can be computed [46].

Let M and N denote the number of gray levels and the maximum run length, respectively. Moreover, let N_r be the total number of runs and let N_p be the number of pixels in the image. Given a set of directions $\{0^\circ, 45^\circ, 90^\circ, 135^\circ\}$, a run length matrix can be computed. Accordingly, based on each run length matrix, the Short Run Emphasis (SRE), Long Run Emphasis (LRE), Gray-Level Nonuniformity (GLN), Run

Length Nonuniformity (RLN), and Run Percentage (RP) are defined as follows [27, 46]:

$$SRE = \frac{1}{N_r} \sum_{i=1}^M \sum_{j=1}^N \frac{p(i, j)}{j^2} \quad (12)$$

$$LRE = \frac{1}{N_r} \sum_{i=1}^M \sum_{j=1}^N p(i, j) \times j^2 \quad (13)$$

$$GLN = \frac{1}{N_r} \sum_{i=1}^M \left(\sum_{j=1}^N p(i, j) \right)^2 \quad (14)$$

$$RLN = \frac{1}{N_r} \sum_{j=1}^N \left(\sum_{i=1}^M p(i, j) \right)^2 \quad (15)$$

$$RP = \frac{N_r}{N_p} \quad (16)$$

In addition to the above five properties, Chu et al. [51] proposed the Low Gray-Level Run Emphasis (LGRE) and High Gray-Level Run Emphasis (HGRE) as follows:

$$LGRE = \frac{1}{N_r} \sum_{j=1}^N \sum_{i=1}^M \frac{p(i, j)}{i^2} \quad (17)$$

$$HGRE = \frac{1}{N_r} \sum_{j=1}^N \sum_{i=1}^M p(i, j) \times i^2 \quad (18)$$

Furthermore, the Short Run Low Gray-Level Emphasis (SRLGE), Short Run High Gray-Level Emphasis (SRHGE), Long Run Low Gray-Level Emphasis (LRLGE), and Long Run High Gray-Level Emphasis (LRHGE) have been proposed by Dasarthy and Holder [48] as follows:

$$SRLGE = \frac{1}{N_r} \sum_{j=1}^N \sum_{i=1}^M \frac{p(i, j)}{i^2 \times j^2} \quad (19)$$

$$SRHGE = \frac{1}{N_r} \sum_{j=1}^N \sum_{i=1}^M \frac{p(i, j) \times i^2}{j^2} \quad (20)$$

$$LRLGE = \frac{1}{N_r} \sum_{j=1}^N \sum_{i=1}^M \frac{p(i, j) \times j^2}{i^2} \quad (21)$$

$$LRHGE = \frac{1}{N_r} \sum_{j=1}^N \sum_{i=1}^M p(i, j) \times i^2 \times j^2 \quad (22)$$

2.2. Stochastic Gradient Descent Logistic Regression. It is noted that the task at hand is to construct a decision boundary that divides data instances into two class labels of nonspall and spall. Thus, the logistic regression (LR) model, which is a powerful pattern classifier, can be employed [52]. LR is selected to establish the spall detection model in this study because its learning phase is straightforward. Moreover, the LR model structure is easy to interpret. This machine learning

```

Procedure SGD
Create a training dataset
Randomly create  $\theta$ 
Defining  $MaxEpoch$  // the maximum number of epochs
Defining  $\alpha$  // the learning rate parameter
For  $ep = 1$  to  $MaxEpoch$ 
    Shuffle samples in the training data set
    For  $i = 1$  to  $M$  //  $M =$  number of data samples
        For  $k = 0$  to  $D$ 
             $\theta_k = \theta_k + \alpha \frac{\partial l(\theta_k)}{\partial (\theta_k)}$ 
        End For
    End For
End For
Return  $\theta$ 

```

ALGORITHM 1: Pseudo code of the Stochastic Gradient Descent (SGD) algorithm.

method has also been successfully applied in various recent applications [53–56].

Let y be the outcome of the model. $y = 1$ (the positive class) when an image sample is subject to spall and $y = 0$ (the negative class) when an image sample is free from spall. Let x be a vector of input features which are extracted from an image sample. Herein, $x_i = x_{i1}, x_{i2}, \dots, x_{iD}$ where D denotes the number of the features used for classification. In addition, a vector of $\theta = \theta_0, \theta_1, \theta_2, \dots, \theta_D$ represents adaptable parameters of a LR model.

$h_\theta(x_i)$ is used to express the probability of the positive class output of spall which is calculated as follows [35]:

$$h_\theta(x_i) = h_\theta(x_{i1}, x_{i2}, \dots, x_{iD}) = \frac{1}{1 + \exp(-\eta_i)} \quad (23)$$

$$= \frac{1}{1 + \exp(-\theta^T x_i)}$$

where $\eta_i = \theta_0 + \theta_1 x_{i1} + \theta_2 x_{i2} + \dots + \theta_D x_{iD} = \theta^T x_i$.

Notably, $g(\eta_i) = 1/(1 + \exp(-\eta_i))$ is called the logistic function or the sigmoid function and its derivative is expressed in the following form [41]:

$$g'(\eta_i) = g(\eta_i) \times (1 - g(\eta_i)) \quad (24)$$

The probabilities of the positive and negative classes are given as follows:

$$P(y_i = 1 | x_i, \theta) = h_\theta(x_i) \quad (25)$$

$$P(y_i = 0 | x_i, \theta) = 1 - h_\theta(x_i) \quad (26)$$

Hence, the output probability can be computed in the following equation [41]:

$$P(y_i = 0 | x_i, \theta) = (h_\theta(x_i))^{y_i} (1 - h_\theta(x_i))^{1-y_i} \quad (27)$$

The likelihood of the LR model parameters can be expressed as follows [41]:

$$L(\theta) = \prod_{i=1}^M (h_\theta(x_i))^{y_i} (1 - h_\theta(x_i))^{1-y_i} \quad (28)$$

where M represents the number of data samples.

To identify the model parameter θ , the following log likelihood function is maximized:

$$l(\theta) = \log(L(\theta)) \quad (29)$$

$$= \sum_{i=1}^M y_i \log(h_\theta(x_i)) + (1 - y_i) (1 - \log(h_\theta(x_i)))$$

The stochastic gradient descent (SGD) [36] can be employed to construct the LR model by adapting its adaptable parameters θ . Before the model construction phase, the original collected data sample should be divided into two sets: a training set and a testing set. The first set is employed to adapt the model parameters; the latter set is reserved for confirming the model generalization capability. The procedure of the SGD algorithm is described in Algorithm 1.

Within the SGD algorithm, the quantity $\partial l(\theta_k)/\partial (\theta_k)$ can be computed as follows:

$$\frac{\partial l(\theta_k)}{\partial (\theta_k)} = (y_i - h_\theta(x_i)) x_{i,k} \quad (30)$$

Thus, the update rule employed to determine the LR model parameter θ is expressed as follows:

$$\theta_k = \theta_k + \alpha (y_i - h_\theta(x_i)) x_{i,k} \quad (31)$$

where $x_{i,0} = 1$ with all i .

2.3. Piecewise Linear Model. As stated earlier, one limitation of the standard LR is that its decision is given in the form of a linear model which is essentially a hyperplane. This study aims at extending the capability of LR by employing a piecewise linear decision surface. The underlying concept is illustrated in Figure 2. Herein, a model $y(x)$ is used to separate the input space into two regions characterizing two data categories. Instead of using a linear decision surface, a piecewise linear one is employed to fit a subset of the input data x . The transition location from a certain subset to another one is termed a breakpoint or a knot [57].

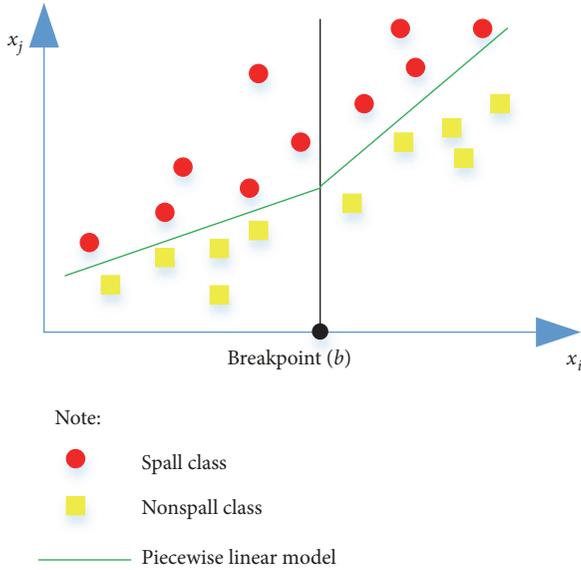


FIGURE 2: Piecewise linear model.

The breakpoints enhance the flexibility of the model by disintegrating the input space into subspaces in which each linear model can be used to fit the collected data [40].

Similar to the concept of a piecewise linear regression [58], the mathematical formulation of a piecewise linear model with one breakpoint is shown as follows:

$$y(x_i) = \begin{cases} \sum_{d=1}^{D+1} \theta_d x_{i,d} & \text{if } x_{i,d} \leq b \\ \sum_{d=1}^{D+1} \theta_d x_{i,d} & \text{if } x_{i,d} > b \end{cases} \quad (32)$$

where x_i denotes the vector of the i^{th} predicting variable consisting of D elements. b represents the knot value. y denotes the model output.

The model establishment of the piecewise linear classification model requires the selection of the knots (b) and the model parameter (θ). In this study, the knot position is identified via a sequential algorithm described in the previous work of Hoang [40]. In addition, based on the collected data samples and the selected knots, the aforementioned SGD algorithm is used to train piecewise LR and reveal the set of θ that brings about the best fit to the data set at hand.

3. The Image Data Set

Since the logistic regression is a supervised machine learning algorithm, a data set consisting of 1240 pavement image samples with the ground truth label has been collected to construct the logistic regression based classification model. Herein, the numbers of image samples in the two categories of nonspalling and spalling are both 620. The digital images have been collected during survey trip of several high-rise buildings in Danang city (Vietnam). The employed camera is the Cannon EOS M10. The camera is positioned at a distance of about 1.5 meter from the concrete surface. Image samples

of the two categories of nonspalling (label = 0) and spalling (label = 1) have been prepared for further analysis.

To expedite the speed of the feature extraction process, the size of image sample has been fixed to be 100x100 pixels. The collected image samples are demonstrated in Figure 3. It is worth noticing that the ground truth of image samples is assigned by human inspectors and the wall condition (either nonspalling or spalling) is determined at the image level. Moreover, an image is labeled as spalling if the spalling area occupies at least 50% of the entire image sample. To ensure the diversity of the image set, the class of nonspalling includes samples of intact concrete surface, cracks, and stains; the class of spalling also takes into account samples in which steel reinforcement is revealed.

4. The Proposed Piecewise Linear Stochastic Gradient Descent Logistic Regression Model for Wall Spall Detection

This section of the study describes the overall structure of the newly developed piecewise linear (PL) stochastic gradient descent logistic regression (SGDLR) model for wall spall detection. The proposed model, named PL-SGDLR, is a hybridization of image texture computation and data classification approach. The statistical measurements of color channels, GLCM, and GLRL are employed for computing the texture of each image sample. The PL-SGDLR uses the image texture as features for classifying data samples into the categories of nonspall and spall. The model structure is illustrated in Figure 4 which basically includes two modules: image texture based feature extraction and PL-SGDLR based data classification. The first module is developed in Visual C#.NET by the author; the second module is programmed in MATLAB. The graphical user interfaces of the two modules are demonstrated in Figure 5.

In the first module of feature extraction, image texture computing techniques including statistical analysis of color channels, GLCM, and GLRL are employed to extract features from image samples. This module has been developed by the author in Visual C# .NET (Framework 4.6.1). First, the group of features based on statistical properties of color images is computed. For each of the three color channels (red, green, and blue), six statistical indices of mean, standard deviation, skewness, kurtosis, entropy, and range are computed according to the aforementioned formulas. Thus, the total number of features extracted from these statistical measurements of an image sample is $6 \times 3 = 18$.

Second, the group of texture features extracted from four cooccurrence matrices corresponding to the directions of 0° , 45° , 90° , and 135° are obtained. Since each cooccurrence matrix yields four properties of the angular second moment, contrast, correlation, and entropy, the total number of features extracted from GLCMs is $4 \times 4 = 16$. Third, the feature group extracted from four GLRL matrices is calculated. The four GLRL matrices are constructed by considering the texture of pixels in the four directions of 0° , 45° , 90° , and 135° . For each GLRL matrix, the 11 properties of the Short Run Emphasis (SRE), Long Run Emphasis (LRE), Gray-Level

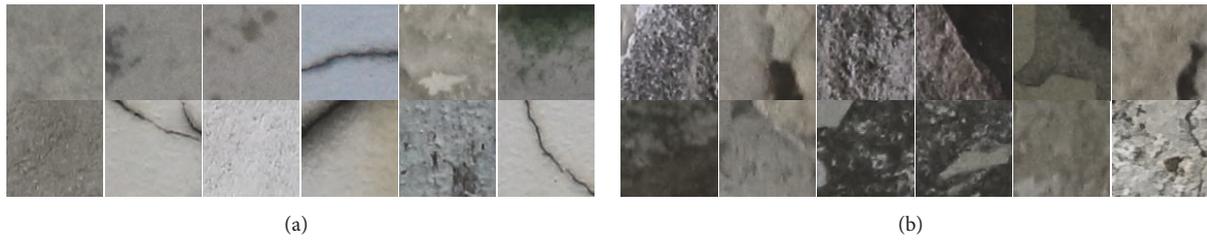


FIGURE 3: The collected image samples: (a) nonspalling class and (b) spalling class.

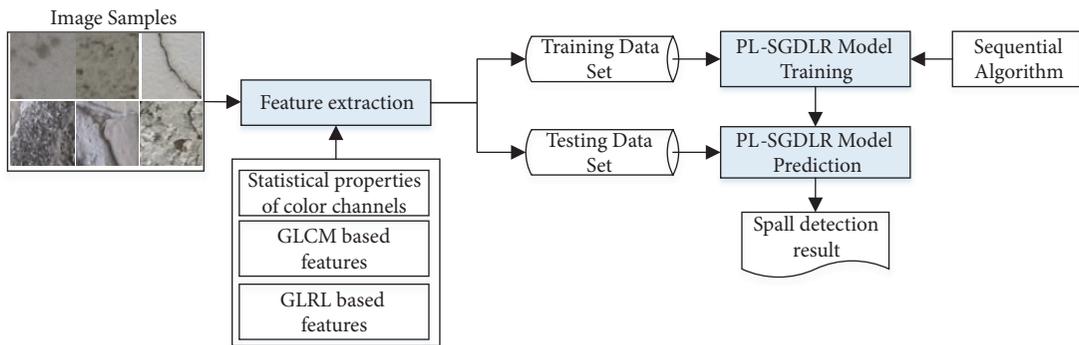
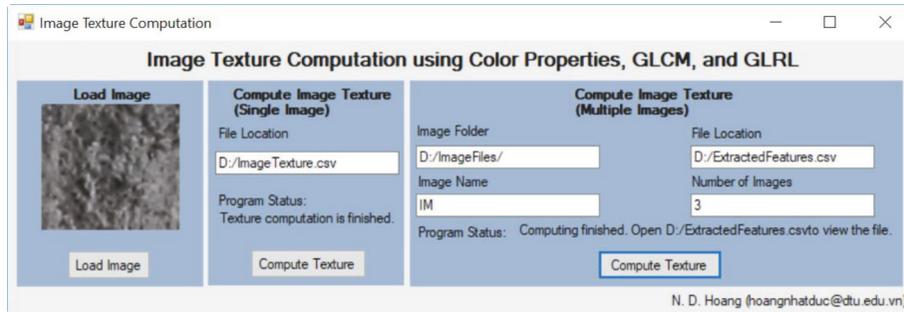
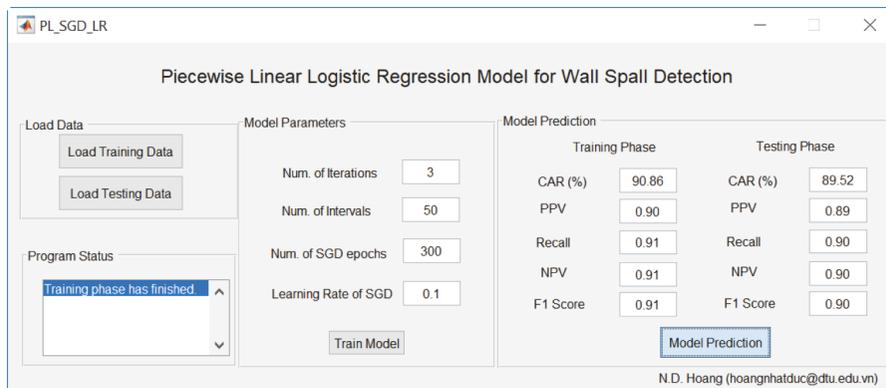


FIGURE 4: Structure of the proposed PL-SGD LR.



(a)



(b)

FIGURE 5: Graphical user interface (GUI) of the PL-SGD LR program. (a) Texture computation module and (b) data classification module.

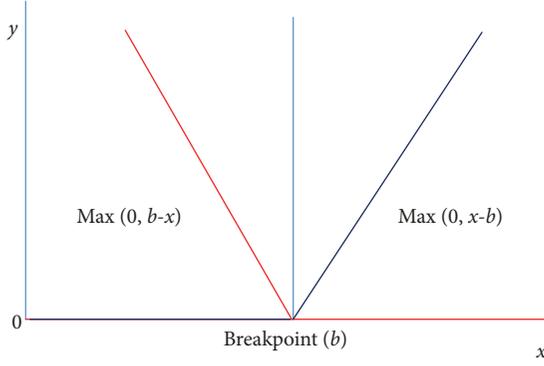


FIGURE 6: Concept of a hinge function.

Nonuniformity (GLN), Run Length Nonuniformity (RLN), Run Percentage (RP), Low Gray-Level Run Emphasis (LGRE), High Gray-Level Run Emphasis (HGRE), the Short Run Low Gray-Level Emphasis (SRLGE), Short Run High Gray-Level Emphasis (SRHGE), Long Run Low Gray-Level Emphasis (LRLGE), and Long Run High Gray-Level Emphasis (LRHGE) are calculated to represent an image texture. Hence, the total number of features extracted from GLRL matrices is $4 \times 11 = 44$.

Accordingly, each image sample is presented by a feature vector consisting of $18 + 16 + 44 = 78$ elements. When the feature extraction module finishes, a numerical data set consisting of 1240 data samples and 78 input features is prepared for further analysis. This data set has two class outputs: 0 denoting nonspall (negative class) and 1 denoting spall (positive class). For standardizing the data ranges and facilitating the data modeling process, the extracted data set has been processed using Z-score data normalization [59]. Furthermore, the widely employed statistical procedure of principal component analysis (PCA) is employed for dimension reduction. PCA basically converts the input features of the original numerical data set into a set of linearly uncorrelated variables [60]. The processed data is then randomly separated into two sets: a training set (90%) and a testing set (10%). The first data set is used for model construction; the latter data set is reserved for model verification.

The training phase of a PL-SGDLR model relies on the concept of a hinge function [61] (see Figure 6). As can be seen from the figure, the output of a hinge function is zero for a certain part of its range. Therefore, this function is useful for dividing the data into separated regions; each of the regions can be satisfactorily fitted by a linear model. Using such a concept of hinge functions, a PL-SGDLR model having one predicting variable x and one breakpoint b is given as follows:

$$y = \beta_0 + \beta_{11} \max(0, \text{sign}(x - b)) + \beta_{12} \max(0, \text{sign}(b - x)) + \beta_{21} \max(0, x - b) + \beta_{22} \max(0, b - x) \quad (33)$$

Hence, the output $y(x)$ according to different values of the explanatory variable x can be written as follows:

$$(i) \text{ If } x > b \text{ then } y = \beta_{11} + \beta_{21} \max(0, x - b).$$

$$(ii) \text{ If } x < b \text{ then } y = \beta_{12} + \beta_{22} \max(0, b - x).$$

$$(iii) \text{ If } x = b \text{ then } y = \beta_0.$$

In essence, at two sides of a breakpoint b of the predicting variable x , two linear models are constructed. The terms β_0 , β_{11} , β_{12} , β_{21} , and β_{22} are parameters of these two linear models. Without much difficulty, the model with one predicting variable and one breakpoint can be generalized to a model with many predicting variables and multiple breakpoints in the following manner:

$$y = \sum_{d=1}^D \sum_{v=1}^{V_d} LF_{d,v}(x_d) \quad (34)$$

where d is the index of predicting variables; D denotes the number of predicting variables; v represents the index of the hinge function of the d^{th} predicting variable; V_d denotes the number of hinge functions of the d^{th} predicting variable.

To identify the appropriate breakpoints for the predicting variables, the range of each input feature x_d is partitioned into N_{BP} equally spaced subranges as follows: $\min(x_d) < b_{d,1} < b_{d,2} < \dots < \max(x_d)$. Accordingly, each variable x has $N_{BP} + 1$ candidates of knots. The model is then constructed sequentially by adding a suitable breakpoint for each input variable in each iteration. Procedure of the model construction phase is demonstrated in Algorithm 2.

In order to accept a breakpoint from a set of candidates, the following fitness function is proposed:

$$f_{Fitness} = \frac{PPV + NPV}{2} + \frac{\lambda}{SumBP + \omega} \quad (35)$$

where PPV and NPV are Positive Predictive Value and Negative Predictive Value, respectively. λ denotes a regularization parameter; $SumBP$ is the number of currently accepted breakpoints; $\omega = 1$ is a scalar simply used to ensure numerical stability.

These two quantities of PPV and NPV are computed as follows:

$$\text{Positive Predictive Value (PPV)} = \frac{TP}{TP + FP} \quad (36)$$

$$\text{Negative Predictive Value (NPV)} = \frac{TN}{TN + FN} \quad (37)$$

In (35), the first term $((PPV + NPV)/2)$ represents the model classification accuracy; the second term $(\lambda/(SumBP + \omega))$ is used to quantify the model complexity. It is reasonable to obtain a model featuring a high value of classification accuracy with moderated complexity. It is because a model having a high degree of complexity tends to be overfitted. Moreover, the model complexity can be expressed in terms of the $SumBP$. Therefore, it is desirable to obtain a model with high value of both predictive values (PPV and NPV) and a low $SumBP$. The breakpoint acceptance criterion (BAC) calculates the fitness value to examine the benefit of accepting a knot candidate. If a candidate can help to increase the model's fitness value, it is allowed to enter the model structure. It is noted that in order to compute the classification accuracy the overall LR model is fitted by the SGD described in the previous section of the study.

```

Define the breakpoint acceptance criterion (BAC)
Define the maximum number of iterations  $MaxIter$ 
Define the parameter  $N_{BP}$ 
For  $i = 1$  to  $MaxIter$ 
    Initialize Model_Structure = []
    For  $d = 1$  to  $D$  //  $D$  is the number of predicting variables
        Identify a breaking point for  $X_d$  based on BAC
        Update Model_Structure
        Identify Model_Parameter  $\beta$  using SGD algorithm
    End For
End For
Return Model_Structure and Model_Parameter

```

ALGORITHM 2: Pseudo code of the model training phase.

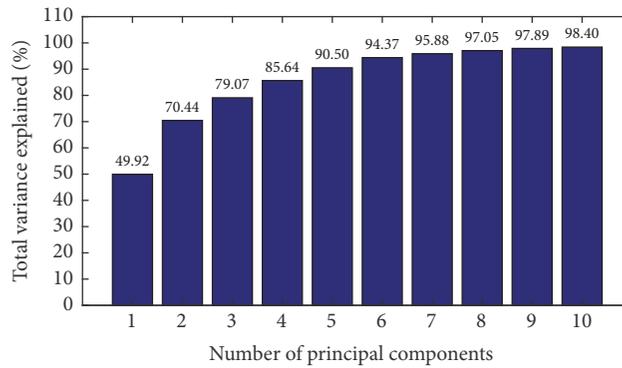


FIGURE 7: Total variance explained by principal components.

5. Experimental Result and Comparison

As mentioned in the previous section, the data set, which consists of 1240 samples and 78 input features, is employed to construct and validate the proposed PL-SGDLR approach. The original input data with the number of features = 78 has been preprocessed by PCA to eliminate linear correlation among its variables. The result of the PCA data transformation process is a new set of linearly uncorrelated variables; each new variable is a linear combination of the 78 original features representing texture of image samples.

Figure 7 reports the PCA result in the form of the total variance explained by principal components. Based on several trial runs, the threshold of total variance = 95% is used to select the suitable number of principal components. Accordingly, the number of principal components = 7 (corresponding to the total variance = 95.88%) is used. Additionally, the feature extraction process of the proposed PL-SGDLR model is demonstrated with an image sample of nonspall class (Figure 8(a)) and with an image sample of spall class (Figure 8(b)).

Based on the PCA result, the transformed data set including 7 input variables and the class label of either 0 (nonspall) or 1 (spall) has been divided into training and testing subsets. The former and the latter subsets consist of 90% and 10% of the collected data set, respectively. The first set is employed in the model construction phase; the

second set is reserved for evaluating the model generalization capability when predicting spalls in novel image samples. Furthermore, because one time of model training and testing cannot well reveal the model generalization capability due to the randomness in data selection, this study has performed a random subsampling of the original data set. This random subsampling process contains 20 runs. In each run, 10% of the data is randomly drawn to form the testing set; the rest of the data is used for model training purpose.

As can be seen from the training process of PL-SGDLR, it is required to select the parameters of the number of training iterations ($MaxIter$), the number of training epochs ($MaxEpoch$), and N_{BP} which determines the number of knot candidates, learning rate (α) used in the SGD algorithm, and regularization parameter (λ) used in the training phase of PL-SGDLR. The suitable values of these hyperparameters of the model have been experimentally found as follows: $MaxIter = 3$, $MaxEpoch = 300$, $N_{BP} = 50$, learning rate (α) = 0.1, $\lambda = 0.01$.

In addition, besides the aforementioned PPV and NPV, Classification Accuracy Rate (CAR), Recall, and F1 score can also be employed to express the model spall detection result. These performance measurement indices are calculated as follows [62]:

$$\begin{aligned}
 &\text{Classification Accuracy Rate: CAR} \\
 &= \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (38)
 \end{aligned}$$

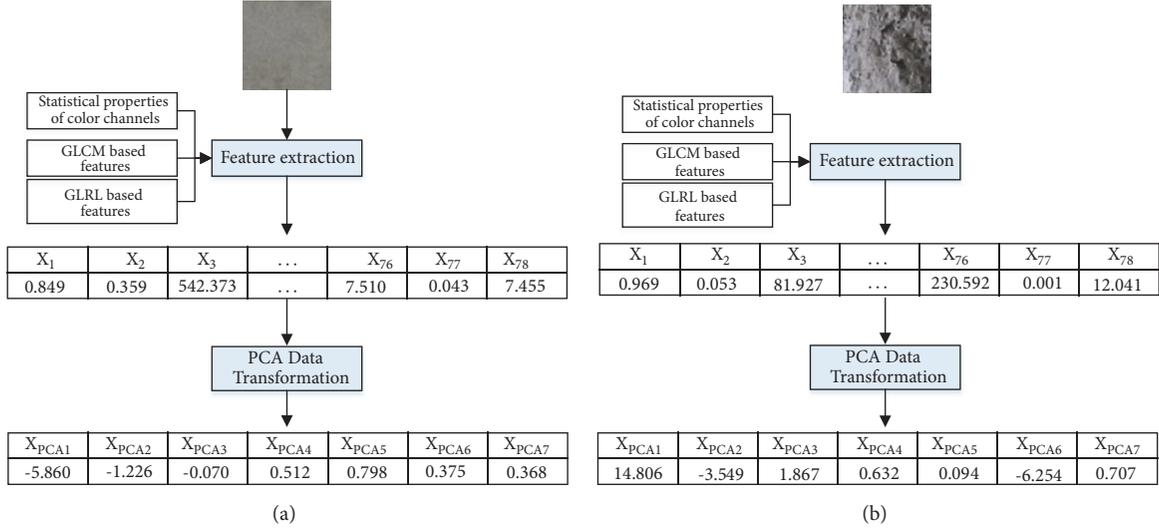


FIGURE 8: Illustration of the feature extraction process. (a) A nonspall sample and (b) a spall sample.

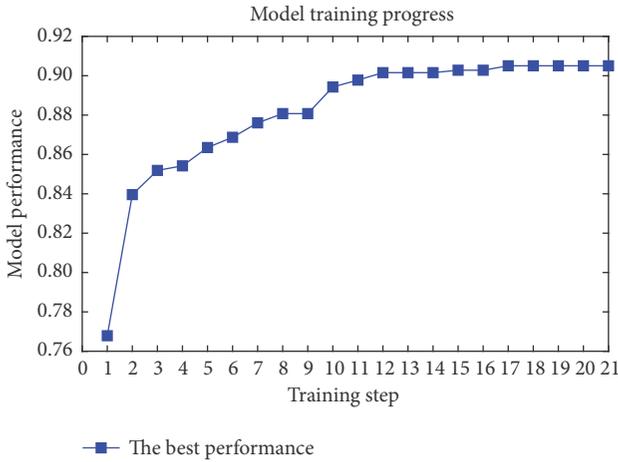


FIGURE 9: Illustration of the PL-SGDLR model's training progress.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (39)$$

$$\text{F1 Score} = \frac{2TP}{2TP + FP + FN} \quad (40)$$

After the repeated data sampling with 20 independent runs, the average performance of the PL-SGDLR model used for predicting testing samples is reported as follows: CAR = 90.24%, PPV = 0.90, Recall = 0.91, NPV = 0.91, and F1 score = 0.90. A typical training phase of the proposed PL-SGDLR model is demonstrated in Figure 9. In this figure, the horizontal axis denotes the training step which is equal to the number of $MaxIter$ multiplied by the number of input variables. Herein, $MaxIter$ is 3 and the number of input variables is 7. The vertical axis represents the fitness function value described in (35).

As mentioned earlier, to guarantee the diversity of the image set and to better cope with the real-world circumstance, anomalies such as cracks and stains have been included in the image data set (see Figure 10). The category of spalling also contains image samples in which steel reinforcement is revealed. Image samples in which spalling and anomalies (such as crack and stains) coexist are also included in the image samples to train and verify the prediction model. Based on experimental results, the prediction model can predict the correct labels of the image samples containing anomalies.

In addition, to demonstrate the capability of the proposed PL-SGDLR model, the SGD-LR and the Levenberg-Marquardt Backpropagation Artificial Neural Network (LM-ANN) [63] are utilized as benchmark approaches. These two machine learning models are selected due to their successful applications reported in previous studies [37, 39, 64–66]. SGD-LR is programmed in MATLAB by the author. In addition, the LM-ANN model is constructed by the MATLAB's Statistics and Machine Learning Toolbox [67]. The SGD-LR is also trained with 300 epochs and the learning rate = 0.1. In addition, via several trial-and-error runs, the appropriate configuration of the BPANN model is as follows: the number of neurons = 7, the learning rate = 0.01, and the number of training epochs = 1000.

The performances of spall detection models obtained from the repeated data sampling with 20 runs are summarized in Table 1. As can be observed from the experimental outcomes, the proposed PL-SGDLR has obtained the best predictive performance (CAR = 90.24%, PPV = 0.90, Recall = 0.91, NPV = 0.91, and F1 score = 0.90), followed by LM-ANN (CAR = 88.83%, PPV = 0.88, Recall = 0.90, NPV = 0.90, and F1 score = 0.89) and SGD-LR (CAR = 84.40%, PPV = 0.84, Recall = 0.85, NPV = 0.85, and F1 score = 0.84). The box plots of spalling detection performance of the proposed PL-SGDLR as well as the benchmark models of LM-ANN and SGD-LR are shown in Figure 11. As shown in this figure, the

TABLE 1: Prediction Results Comparison.

Phase	Metrics	SGD-LR		LM-ANN		PL-SGDLR	
		Mean	Std.	Mean	Std.	Mean	Std.
Training	CAR (%)	86.42	0.40	90.68	1.46	91.07	0.72
	PPV	0.86	0.02	0.90	0.02	0.90	0.02
	Recall	0.87	0.02	0.91	0.01	0.93	0.02
	NPV	0.87	0.01	0.91	0.01	0.93	0.02
	F1 Score	0.86	0.00	0.91	0.01	0.91	0.01
Testing	CAR (%)	84.40	3.56	88.83	3.71	90.24	2.48
	PPV	0.84	0.05	0.88	0.04	0.90	0.04
	Recall	0.85	0.05	0.90	0.05	0.91	0.02
	NPV	0.85	0.04	0.90	0.04	0.91	0.02
	F1 Score	0.84	0.04	0.89	0.04	0.90	0.02

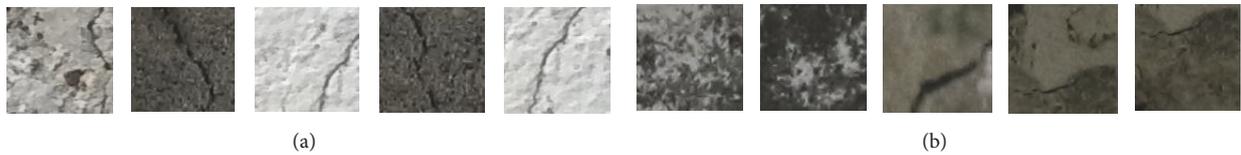


FIGURE 10: Illustrations of image samples containing anomalies: (a) cracks and (b) stains.

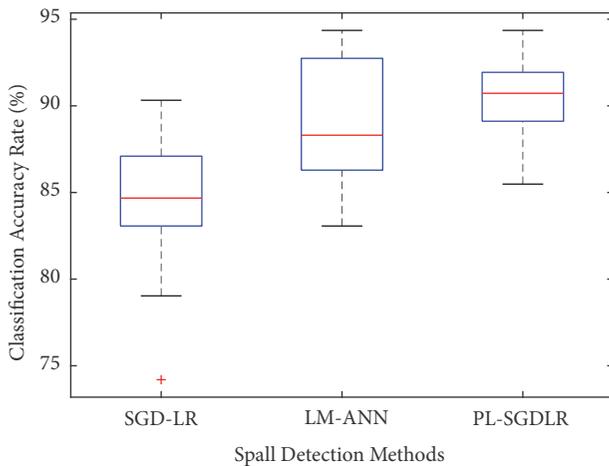


FIGURE 11: Box plots of classification performance.

median value of the CAR of the PL-SGDLR (90.72%) is also higher than those of the LM-ANN (84.68%) and SGD-LR (88.31%). Thus, it can be seen that the newly developed PL-SGDLR has outperformed the two benchmark methods in all of the employed performance measurement indices.

In addition, the Wilcoxon signed-rank test [68] is also employed in this section to investigate the statistical difference of each pair of spalling detection methods. Herein, the significance level of the test is set to be 0.05. By assessing the CAR values obtained from the repeated data sampling with 20 runs, the Wilcoxon signed-rank test shows that the spalling detection performance of the PL-SGDLR is statistically different from that of the SGD-LR with the p-values = 0.0001. Nevertheless, the statistical test regarding the performances of the PL-SGDLR and LM-ANN models has

the p-values = 0.1971. Based on this result, it can be seen that the performance of the PL-SGDLR is highly competitive to that of the LM-ANN. However, since the CAR, PPV, Recall, NPV, and F1 Score of the PL-SGDLR are higher than those of LM-ANN, it is able to confirm that PL-SGDLR is a capable tool for detecting concrete wall spalls.

6. Conclusion

Detecting spalled areas in concrete wall structures is an important task in structural health monitoring. This study proposes a computer vision based model to replace the time-consuming manual method commonly used for periodic building survey. The proposed model is a hybridization of image texture analysis and machine learning approaches. Image texture computed by the statistical measurements of color channels, GLCM, and GLRL is employed as features to characterize the condition of concrete wall surface. Based on such extracted features, the PL-SGDLR is employed to classify image samples into two categories of nonspall and spall. An image data set consisting of 1240 samples has been collected to train and verify the PL-SGDLR model.

This study also extends the modeling capability of the standard LR model by employing a piecewise linear decision surface. A sequential procedure is proposed to iteratively construct the piecewise linear LR model. Experimental results point out that the newly developed model can help to achieve good spall detection accuracy with CAR = 90.24%. This result is better than those of LM-ANN (CAR = 88.83%) and LR (CAR = 84.40%). Since the performance of PL-SGDLR is better than that of LR, it is able to confirm that the utilization of a piecewise linear decision surface can help the LR to extend its nonlinear modeling capability. Accordingly, the proposed PL-SGDLR can be a potential

tool to assist maintenance agencies in the task of periodic survey. Future extensions of the current work may include the investigation into the effect of different spatial resolutions on the spalling detection results and the utilization of other advanced machine learning methods to enhance the prediction accuracy. Furthermore, the effect of different percentages of the spalling area on the model prediction outcome is also worth investigating.

Data Availability

The supplementary file provides the data set used in this study. The data set and the developed programs can be accessed via https://github.com/NhatDucHoang/PL_LR_WSD.

Conflicts of Interest

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

Acknowledgments

This research is financially supported by Duy Tan University.

Supplementary Materials

The supplementary file provides the data set used in this study. The first 78 columns of the data are texture based features extracted from image samples. The last column is the ground truth label of the data instance with 0 = “nonspall” and 1 = “spall.” The data set and the developed programs can be accessed via https://github.com/NhatDucHoang/PL_LR_WSD. (*Supplementary Materials*)

References

- [1] W. Zhang, Z. Zhang, D. Qi, and Y. Liu, “Automatic crack detection and classification method for subway tunnel safety monitoring,” *Sensors*, vol. 14, no. 10, pp. 19307–19328, 2014.
- [2] N. Hoang, “Image processing-based recognition of wall defects using machine learning approaches and steerable filters,” *Computational Intelligence and Neuroscience*, vol. 2018, pp. 1–18, 2018.
- [3] T. Dawood, Z. Zhu, and T. Zayed, “Machine vision-based model for spalling detection and quantification in subway networks,” *Automation in Construction*, vol. 81, pp. 149–160, 2017.
- [4] S. German, I. Brilakis, and R. Desroches, “Rapid entropy-based detection and properties measurement of concrete spalling with machine vision for post-earthquake safety assessments,” *Advanced Engineering Informatics*, vol. 26, no. 4, pp. 846–858, 2012.
- [5] M. Kim, H. Sohn, and C. Chang, “Localization and quantification of concrete spalling defects using terrestrial laser scanning,” *Journal of Computing in Civil Engineering*, vol. 29, no. 6, Article ID 04014086, 2015.
- [6] Y. Cha, W. Choi, G. Suh, S. Mahmoudkhani, and O. Büyüköztürk, “Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 9, pp. 731–747, 2018.
- [7] A. Mohan and S. Poobal, “Crack detection using image processing: a critical review and analysis,” *Alexandria Engineering Journal*, 2017.
- [8] Y.-S. Yang, C.-I. Wu, T. T. C. Hsu, H.-C. Yang, H.-J. Lu, and C.-C. Chang, “Image analysis method for crack distribution and width estimation for reinforced concrete structures,” *Automation in Construction*, vol. 91, pp. 120–132, 2018.
- [9] H. K. Jung and G. Park, “Rapid and non-invasive surface crack detection for pressed-panel products based on online image processing,” *Structural Health and Monitoring*, 2019.
- [10] S. Suwwanakarn, Z. Zhu, and I. Brilakis, “Automated air pockets detection for architectural concrete inspection,” in *In Proceedings of the Construction Congress I, American Society of Civil Engineers*, Reston, VA, USA, 2007.
- [11] C. Koch and I. Brilakis, “Pothole detection in asphalt pavement images,” *Advanced Engineering Informatics*, vol. 25, no. 3, pp. 507–515, 2011.
- [12] J. Valença, L. Gonçalves, and E. Júlio, “Damage assessment on concrete surfaces using multi-spectral image analysis,” *Construction and Building Materials*, vol. 40, pp. 971–981, 2013.
- [13] M.-K. Kim, J. C. P. Cheng, H. Sohn, and C.-C. Chang, “A framework for dimensional and surface quality assessment of precast concrete elements using BIM and 3D laser scanning,” *Automation in Construction*, vol. 49, pp. 225–238, 2015.
- [14] S. G. Paal, J. Jeon, I. Brilakis, and R. DesRoches, “Automated damage index estimation of reinforced concrete columns for post-earthquake evaluations,” *Journal of Structural Engineering*, vol. 141, no. 9, Article ID 04014228, 2015.
- [15] S. Li, C. Yuan, D. Liu, and H. Cai, “Integrated processing of image and GPR data for automated pothole detection,” *Journal of Computing in Civil Engineering*, vol. 30, no. 6, Article ID 04016015, 2016.
- [16] S. Konishi, K. Kawakami, and M. Taguchi, “Inspection method with infrared thermometry for detect void in subway tunnel lining,” *Procedia Engineering*, vol. 165, pp. 474–483, 2016.
- [17] B. Oliveira Santos, J. Valença, and E. Júlio, “Automatic mapping of cracking patterns on concrete surfaces with biological stains using hyper-spectral images processing,” *Structural Control and Health Monitoring*, vol. 26, no. 3, Article ID e2320, 2019.
- [18] B. O. Santos, J. Valença, and E. Júlio, “Detection of cracks on concrete surfaces by hyperspectral image processing,” in *Proceedings of the Automated Visual Inspection and Machine Vision II*, vol. 10334, SPIE Optical Metrology, Germany.
- [19] C. Liu, S. Shirowzhan, S. M. E. Sepasgozar, and A. Kaboli, “Evaluation of classical operators and fuzzy logic algorithms for edge detection of panels at exterior cladding of buildings,” *Buildings*, vol. 9, no. 2, article 40, 2019.
- [20] B. Wang, Y. Li, W. Zhao, Z. Zhang, Y. Zhang, and Z. Wang, “Effective crack damage detection using multilayer sparse feature representation and incremental extreme learning machine,” *Applied Sciences*, vol. 9, no. 3, article 614, 2019.
- [21] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth, “A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure,” *Advanced Engineering Informatics*, vol. 29, no. 2, pp. 196–210, 2015.
- [22] X. W. Ye, C. Z. Dong, and T. Liu, “A review of machine vision-based structural health monitoring: methodologies and applications,” *Journal of Sensors*, vol. 2016, Article ID 7103039, 10 pages, 2016.

- [23] N.-D. Hoang, "Image processing based automatic recognition of asphalt pavement patch using a metaheuristic optimized machine learning approach," *Advanced Engineering Informatics*, vol. 40, pp. 110–120, 2019.
- [24] L. G. Shapiro and G. C. Stockman, *Computer Vision*, Prentice Hall, Upper Saddle River, 2001.
- [25] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, Academic Press, 2009.
- [26] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 3, no. 6, pp. 610–621, 1973.
- [27] M. M. Galloway, "Texture analysis using gray level run lengths," *Computer Graphics and Image Processing*, vol. 4, no. 2, pp. 172–179, 1975.
- [28] H. Kim, E. Ahn, M. Shin, and S. Sim, "Crack and noncrack classification from concrete surface images using machine learning," *Structural Health and Monitoring*, vol. 18, no. 3, pp. 725–738, 2018.
- [29] S. Dorafshan, R. J. Thomas, and M. Maguire, "Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete," *Construction and Building Materials*, vol. 186, pp. 1031–1045, 2018.
- [30] L. Li, L. Sun, G. Ning, and S. Tan, "Automatic pavement crack recognition based on bp neural network," *PROMET - Traffic & Transportation*, vol. 26, no. 1, pp. 11–22, 2014.
- [31] G. K. Choudhary and S. Dey, "Crack detection in concrete surfaces using image processing, fuzzy logic, and neural networks," in *Proceedings of the IEEE Fifth International Conference on Advanced Computational Intelligence (ICACI '12)*, pp. 404–411, Nanjing, China, 2012.
- [32] N. Hoang, Q. Nguyen, and D. Tien Bui, "Image processing-based classification of asphalt pavement cracks using support vector machine optimized by artificial bee colony," *Journal of Computing in Civil Engineering*, vol. 32, no. 5, Article ID 04018037, 2018.
- [33] H. Hasni, A. H. Alavi, P. Jiao, and N. Lajnef, "Detection of fatigue cracking in steel bridge girders: A support vector machine approach," *Archives of Civil and Mechanical Engineering*, vol. 17, no. 3, pp. 609–622, 2017.
- [34] S. Wang, S. Qiu, W. Wang, D. Xiao, and K. C. P. Wang, "Cracking classification using minimum rectangular cover-based support vector machine," *Journal of Computing in Civil Engineering*, vol. 31, no. 5, Article ID 04017027, 2017.
- [35] A. Agresti, *An Introduction to Categorical Data Analysis*, Wiley Series in Probability and Statistics, John Wiley & Sons, Inc., Hoboken, NJ, USA, 2019.
- [36] M. Gormley, "Lecture note, 10-701, introduction to machine learning," in *Logistic Regression*, Carnegie Mellon School of Computer Science, 2016, <https://www.cs.cmu.edu/~mgormley/courses/10701-f16/slides/lecture5.pdf>.
- [37] M. Chang, M. Maguire, and Y. Sun, "Stochastic modeling of bridge deterioration using classification tree and logistic regression," *Journal of Infrastructure Systems*, vol. 25, no. 1, Article ID 04018041, 2019.
- [38] L. Lombardo and P. M. Mai, "Presenting logistic regression-based landslide susceptibility results," *Engineering Geology*, vol. 244, pp. 14–24, 2018.
- [39] H. Kim, T. Hong, and J. Kim, "Automatic ventilation control algorithm considering the indoor environmental quality factors and occupant ventilation behavior using a logistic regression model," *Building and Environment*, vol. 153, pp. 46–59, 2019.
- [40] N.-D. Hoang, "Estimating punching shear capacity of steel fibre reinforced concrete slabs using sequential piecewise multiple linear regression and artificial neural network," *Measurement*, vol. 137, pp. 58–70, 2019.
- [41] A. Ng, "Lecture notes," in *CS229 Machine Learning*, Stanford University, 2018, <http://cs229.stanford.edu/notes/cs229-notes1.pdf>.
- [42] F. Tomita and S. Tsuji, *Computer Analysis of Visual Textures*, Springer Science+ Business Media, New York, NY, USA, 1990.
- [43] M. Sonka, V. Hlavac, and R. Boyle, *Image processing, Analysis, and Machine Vision*, Cengage Learning, 2013.
- [44] G. M. Hadjidemetriou, P. A. Vela, and S. E. Christodoulou, "Automated pavement patch detection and quantification using support vector machines," *Journal of Computing in Civil Engineering*, vol. 32, no. 1, Article ID 04017073, 2018.
- [45] A. Jindal, N. Aggarwal, and S. Gupta, "An obstacle detection method for visually impaired persons by ground plane removal using speeded-up robust features and gray level co-occurrence matrix," *Pattern Recognition and Image Analysis*, vol. 28, no. 2, pp. 288–300, 2018.
- [46] X. Tang, "Texture information in run-length matrices," *IEEE Transactions on Image Processing*, vol. 7, no. 11, pp. 1602–1609, 1998.
- [47] J. S. Weszka, C. R. Dyer, and A. Rosenfeld, "Comparative study of texture measures for terrain classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 6, no. 4, pp. 269–285, 1976.
- [48] B. V. Dasarathy and E. B. Holder, "Image characterizations based on joint gray level—run length distributions," *Pattern Recognition Letters*, vol. 12, no. 8, pp. 497–502, 1991.
- [49] B. Abraham and M. S. Nair, "Computer-aided classification of prostate cancer grade groups from MRI images using texture features and stacked sparse autoencoder," *Computerized Medical Imaging and Graphics*, vol. 69, pp. 60–68, 2018.
- [50] M. R. Mookiah, T. Baum, K. Mei et al., "Effect of radiation dose reduction on texture measures of trabecular bone microstructure: an in vitro study," *Journal of Bone and Mineral Metabolism*, vol. 36, no. 3, pp. 323–335, 2018.
- [51] A. Chu, C. M. Sehgal, and J. F. Greenleaf, "Use of gray value distribution of run lengths for texture analysis," *Pattern Recognition Letters*, vol. 11, no. 6, pp. 415–419, 1990.
- [52] W. W. Piegorsch, *Statistical Data Analytics: Foundations for Data Mining, Informatics, and Knowledge Discovery*, John Wiley & Sons, 2015.
- [53] K. Kim, J. Kim, T.-Y. Kwak, and C.-K. Chung, "Logistic regression model for sinkhole susceptibility due to damaged sewer pipes," *Natural Hazards*, vol. 93, no. 2, pp. 765–785, 2018.
- [54] T. K. Saha and S. Pal, "Exploring physical wetland vulnerability of Atreyee river basin in India and Bangladesh using logistic regression and fuzzy logic approaches," *Ecological Indicators*, vol. 98, pp. 251–265, 2019.
- [55] H. C. Chan, C. C. Chang, P. A. Chen, and J. T. Lee, "Using multinomial logistic regression for prediction of soil depth in an area of complex topography in Taiwan," *Catena*, vol. 176, pp. 419–429, 2019.
- [56] N. Hoang, "Automatic detection of asphalt pavement raveling using image texture based feature extraction and stochastic gradient descent logistic regression," *Automation in Construction*, vol. 105, Article ID 102843, 2019.
- [57] S. E. Ryan and L. S. Porth, "A tutorial on the piecewise regression approach applied to bedload transport data," *Gen*

Tech Rep RMRS-GTR-189 Fort Collins, CO: US Department of Agriculture, Forest Service, Rocky Mountain Research Station, 2007.

- [58] M. E. Greene, O. Rolfson, G. Garellick, M. Gordon, and S. Nemes, "Improved statistical analysis of pre- and post-treatment patient-reported outcome measures (PROMs): the applicability of piecewise linear regression splines," *Quality of Life Research*, vol. 24, no. 3, pp. 567–573, 2015.
- [59] V. Nhu, N. Hoang, V. Duong, H. Vu, and D. Tien Bui, "A hybrid computational intelligence approach for predicting soil shear strength for urban housing construction: a case study at Vinhomes Imperia project, Hai Phong city (Vietnam)," *Engineering with Computers*, 2019.
- [60] J. Shlens, "A Tutorial on Principal Component Analysis," <https://arxiv.org/abs/1404.1100v1>.
- [61] L. Breiman, "Hinging hyperplanes for regression, classification, and function approximation," *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 999–1013, 1993.
- [62] A. Tharwat, "Classification assessment methods," *Applied Computing and Informatics*, 2018, <https://doi.org/10.1016/j.aci.2018.08.003>.
- [63] M. Hagan and M. Menhaj, "Training feedforward networks with the marquardt algorithm," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 5, no. 6, pp. 989–993, 1994.
- [64] A. Nandi, A. Mandal, M. Wilson, and D. Smith, "Flood hazard mapping in Jamaica using principal component analysis and logistic regression," *Environmental Earth Sciences*, vol. 75, no. 6, article 465, 2016.
- [65] C. Polykretis and C. Chalkias, "Comparison and evaluation of landslide susceptibility maps obtained from weight of evidence, logistic regression, and artificial neural network models," *Natural Hazards*, vol. 93, no. 1, pp. 249–274, 2018.
- [66] P. Ngo, N. Hoang, B. Pradhan et al., "A novel hybrid swarm optimized multilayer neural network for spatial prediction of flash floods in tropical areas using sentinel-1 SAR imagery and geospatial data," *Sensors*, vol. 18, no. 11, article 3704, 2018.
- [67] Matwork, *Statistics and Machine Learning Toolbox User's Guide*, Matwork Inc., 2017.
- [68] S. Sidney, *Non-Parametric Statistics for the Behavioral Sciences*, McGraw-Hill, New York, NY, USA, 1988.

