

## Research Article

# Automatic Recognition of Asphalt Pavement Cracks Based on Image Processing and Machine Learning Approaches: A Comparative Study on Classifier Performance

Nhat-Duc Hoang <sup>1</sup> and Quoc-Lam Nguyen<sup>2</sup>

<sup>1</sup>Faculty of Civil Engineering, Institute of Research and Development, Duy Tan University, Vietnam

<sup>2</sup>Faculty of Civil Engineering, Duy Tan University, Da Nang, Vietnam

Correspondence should be addressed to Nhat-Duc Hoang; [hoangnhatduc@dtu.edu.vn](mailto:hoangnhatduc@dtu.edu.vn)

Received 21 May 2018; Accepted 28 June 2018; Published 11 July 2018

Academic Editor: Muddasser Alam

Copyright © 2018 Nhat-Duc Hoang and Quoc-Lam Nguyen. 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.

Periodic surveys of asphalt pavement condition are very crucial in road maintenance. This work carries out a comparative study on the performance of machine learning approaches used for automatic pavement crack recognition. Six machine learning approaches, Naïve Bayesian Classifier (NBC), Classification Tree (CT), Backpropagation Artificial Neural Network (BPANN), Radial Basis Function Neural Network (RBFNN), Support Vector Machine (SVM), and Least Squares Support Vector Machine (LSSVM), have been employed. Additionally, Median Filter (MF), Steerable Filter (SF), and Projective Integral (PI) have been used to extract useful features from pavement images. In the feature extraction phase, performance comparison shows that the input pattern including the diagonal PIs enhances the classification performance significantly by creating more informative features. A simple moving average method is also employed to reduce the size of the feature set with positive effects on the model classification performance. Experimental results point out that LSSVM has achieved the highest classification accuracy rate. Therefore, this machine learning algorithm used with the feature extraction process proposed in this study can be a very promising tool to assist transportation agencies in the task of pavement condition survey.

## 1. Introduction

The acceptable level of road serviceability is very crucial to ensure the economic growth and the safety of passengers. Therefore, transportation agencies periodically survey and collect the pavement condition data. Accurate and timely recognition of pavement distress as well as pavement health monitoring measures has increasingly become an integral part of the regional road maintenance system [1]. It is because early detection of pavement distress can help to develop cost-effective rehabilitation methods and prevent the reduction in service life of pavement structures [2].

Cracks are widely considered to be an important indicator of road surface degradation. The causes of cracks in asphalt pavement can be vehicle overload, inclement climatic conditions, and aging of road structure [3]. Detection of cracks in pavement surface is highly useful for the task of

road maintenance. The reason is that if cracks are recognized timely and accurately, the maintenance cost of road can be saved significantly [4].

In developing countries, roads are usually surveyed manually by human inspectors. This traditional approach of road inspection is time-consuming and subjected to variation in assessment outcomes. Therefore, automatic pavement condition inspection and evaluation have become a common desire of transportation agencies. To construct automatic pavement assessment systems, researchers and practitioners extensively rely on image processing techniques within which 2-dimensional images are the input information. Various intelligent methods are then employed to enhance and transform the images to highlight the objects of interest which are pavement cracks [5]. Instead of analyzing the whole image, a set of useful features can be extracted from the image to detect the status of crack and to distinguish the type of cracks [2, 6].

During the last three decades, various research works have dedicated to establishing pavement crack detection models. Kaseko et al. [7] performed a comparative evaluation of neural network classifier and the traditional classifiers of the Bayes classifier and the k-nearest neighbor (k-NN); this work highlighted the potential of neural network in pavement crack classification. Lee and Kim [8] proposed a simple method for feature extraction used for crack class categorization; this method is based on the concept of Crack Type Index. Edge detection methods that employed Canny and Sobel algorithms have been used for crack detection [9]. Wang et al. [10] attempted to use wavelet transform approaches to recognize the existence of cracks and observed promising outcomes. Ying and Salari [11] applied the beamlet transform-based technique to extract linear features of crack objects; this technique has the advantage of crack feature extraction in the presence of noise.

Gavilán et al. [12] established an adaptive road crack detection system which employed Support Vector Machine (SVM) ensembles. Ouma and Hahn [4] constructed an automatic recognition approach of linear cracks based on the wavelet-morphology and circular Radon transform methods. Mokhtari et al. [13] utilized artificial neural network (ANN), decision tree, and k-nearest neighbors to classify pavement images into “Crack” and “No Crack” labels; the ANN was proved to be superior to the decision tree and k-nearest neighbors. Tiled fuzzy Hough transform was applied to detect near straight segments of cracks embedded in pavement textures [14]; this study confirms that the fuzzy Hough transform is effective in diminishing the contribution of texture and noise pixels. Li et al. [3] constructed an automatic method used for both detection and segmentation of pavement cracks using the steerable matched filtering and an active contour model. Fujita et al. [15] proposed a linear SVM based classification model that uses a set of hand-crafted features extracted from digital images.

Cubero-Fernandez et al. [6] extracted the characteristics of images using various techniques including logarithmic transformation, bilateral filter, Canny algorithm, a morphological filter, and Projective Integral method; a Classification Tree is then utilized and applied to categorize the images containing cracks. Zhang et al. [16] and Gopalakrishnan et al. [1] relied on the Convolutional Neural Network (CNN) to classify crack patterns; CNN is the approach that incorporates the feature extraction and the image classification process. However, CNN requires a significant amount of training samples to construct a robust classifier and therefore consumes a considerable computational cost. A SVM based method that takes into account the information of neighboring pixels has been recently introduced by Ai et al. [17]. Hoang and Nguyen [2] employed the image processing methods of Steerable Filters and Projective Integral for the feature extraction task as well as machine learning for classification task. Although the machine learning based method proposed in Hoang and Nguyen [2] has a good performance, this method cannot effectively recognize diagonal crack patterns.

Based on recent review works, the trend of applying automatic methods for pavement condition assessment is increasingly observed in the academic community due to

the affordable cost of image acquisition equipment and the rapid advancements of image processing techniques [18, 19]. Nevertheless, automatic crack detection and classification still face significant challenges including the complexity of the pavement texture, unexpected objects, nonuniform illumination, weak signals of crack patterns, the inhomogeneity of cracks, and the diversity of crack patterns [2, 5, 15]. Therefore, more studies should be dedicated to improving the effectiveness of pavement classification models. This improvement can be achieved either through the enhancement of the feature extraction phase or through the identification of more suitable machine learning approaches.

Based on such motivations, this study proposes an alternative tool for automatic pavement crack classification that employs image processing and machine learning methods. The current study extends the body of knowledge in the following aspects:

- (i) To deal with the complex and noisy texture of the pavement background, image processing techniques including Median Filter, Steerable Filter, and Projective Integral are used in the feature extraction phase.
- (ii) Six machine learning algorithms including Naïve Bayes Classifier, Classification Tree, Backpropagation ANN, radial basis function ANN, SVM, and Least Squares SVM are employed to categorize pavement images into five classes: alligator crack, diagonal crack, longitudinal crack, noncrack, and transverse crack. This study compares the performances of these classifiers to identify the most appropriate one.
- (iii) In addition, to specifically improve the accuracy of detecting diagonal cracks, a rotated Projective Integral method is employed.

The subsequent part of the article is organized as follows: The second section reviews the research methodology; the third section presents the processes of image acquisition and feature extraction followed by the experimental result and comparison; the last section summarizes the study with several remarks.

## 2. Research Methodology

### 2.1. Image Processing Techniques

**2.1.1. Median Filter (MF).** MF, a nonlinear image filtering technique, is an effective approach to noise removal. This image filtering technique is widely used in the field of image processing because it has the advantage of edge preservation. Basically, MF replaces each pixel in the image with the median of its neighboring pixels [20]. The number of the neighboring pixels is determined by the window size (e.g., 3x3 or 5x5 pixels). As demonstrated by Arias-Castro and Donoho [21], MF can be better than Gaussian blur at noise removal and edge preservation edges for a fixed window size. Rababaah [22] experimentally compared the performances of several image denoising techniques and found that MF is the most suitable method for processing asphalt pavement images. **Figure 1** illustrates the effects of MF on denoising a pavement image with different window sizes.

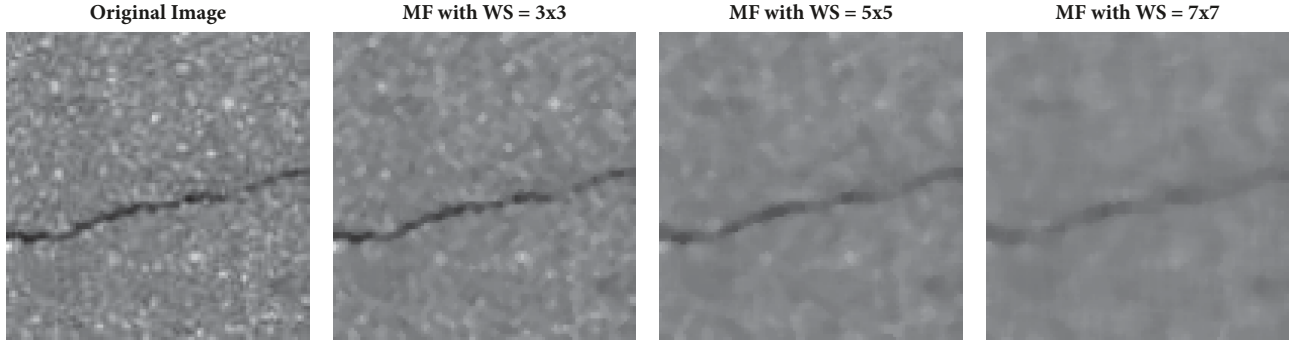


FIGURE 1: Median filtering of a pavement image with different window sizes.

**2.1.2. Steerable Filter (SF).** The Steerable Filter (SF) [23] is essentially an image enhancement technique that employs orientation-selective convolution kernels. As demonstrated in the previous works of Cubero-Fernandez et al. [6] and Hoang and Nguyen [2], this image enhancement technique is particularly useful for differentiating the crack patterns and the background texture of asphalt pavement. In addition to crack detection, SF has been successfully employed in other tasks of the computer vision field such as object tracking, text classification, and distress recognition [3, 24–26].

It is noted that, in the SF algorithm, a linear combination of Gaussian second derivatives is used as a basic filter. For an image  $I(x,y)$ , a 2D Gaussian distribution at a certain pixel coordination is expressed as follows:

$$G(x, y, r) = \frac{1}{\sqrt{2\pi}r} \exp\left(-\frac{(x^2 + y^2)}{2r^2}\right) \quad (1)$$

where  $r$  denotes a tunable parameter of the Gaussian function variance.

The SF formulation with the orientation of  $\theta$  is expressed as follows:

$$F(x, y, r, \theta) = G_{xx} \cos^2(\theta) + 2G_{xy} \cos(\theta) \sin(\theta) + G_{yy} \sin^2(\theta) \quad (2)$$

where  $G_{xx}$ ,  $G_{xy}$ , and  $G_{yy}$  represent the Gaussian second derivatives and their formulas are shown below:

$$G_{xx}(x, y, r) = \frac{(x^2 - r^2) \exp(-(x^2 + y^2)/2r^2)}{\sqrt{2\pi}r^5} \quad (3)$$

$$G_{yy}(x, y, r) = \frac{(y^2 - r^2) \exp(-(x^2 + y^2)/2r^2)}{\sqrt{2\pi}r^5} \quad (4)$$

$$G_{xy}(x, y, r) = G_{xy}(x, y, r) = \frac{xy \exp(-(x^2 + y^2)/2r^2)}{\sqrt{2\pi}r^5} \quad (5)$$

It is worth noticing that when the value of the Gaussian function variance ( $r$ ) is fixed, the final filter response is a combined result of SF with an orientation set  $\theta$ . The value of

$\theta$  is selected from a set of  $\Theta = [0 : 30 : 360]$ . The SF response of an asphalt pavement image containing cracks is illustrated in **Figure 2** with different value of the parameter  $r$ . The final SF response at the pixel location of  $(x, y)$  in the image  $I$  is computed as follows:

$$R(x, y) = F(x, y, \sigma, \theta) * I(x, y) \quad (6)$$

where “\*” is the convolution operator.

**2.1.3. Projective Integral (PI).** In the image processing field, PI is a simple yet effective method to characterize the shape as well as the texture within an image. This method has been widely utilized in the field of face or facial emotion recognition [27]. PI has recently demonstrated its important role in pavement distress classification and recognition [2, 3, 6].

With a digital image  $I(x,y)$ , the horizontal and vertical PIs are commonly employed. The formulas used to compute these two aforementioned PIs are shown as follows:

$$HP(y) = \sum_{i \in x_y} I(i, y) \quad (7)$$

$$VP(x) = \sum_{j \in y_x} I(x, j) \quad (8)$$

where HP and VP represent the horizontal and vertical PIs, respectively.  $x_y$  and  $y_x$  denote the set of horizontal pixels at the vertical pixel  $y$  and the set of vertical pixels at the horizontal pixel  $x$ , respectively.

As shown in the previous work of Hoang and Nguyen [2], HP and VP are highly useful for the task of recognizing the alligator crack, longitudinal crack, noncrack, and transverse crack. The reason is that an alligator crack case and a noncrack case are often characterized by relatively stable PIs in both horizontal and vertical axes; however, the average value of the first case is higher than the latter case. On the other hand, a longitudinal crack case and a transverse crack case ideally feature one peak of intensity in VP and HP, respectively.

However, these two PIs are not sufficient to identify the diagonal crack. An example of image analysis using PI is provided in **Figure 3**. It is clearly shown that HP and VP of an image with an alligator crack are very similar to those of

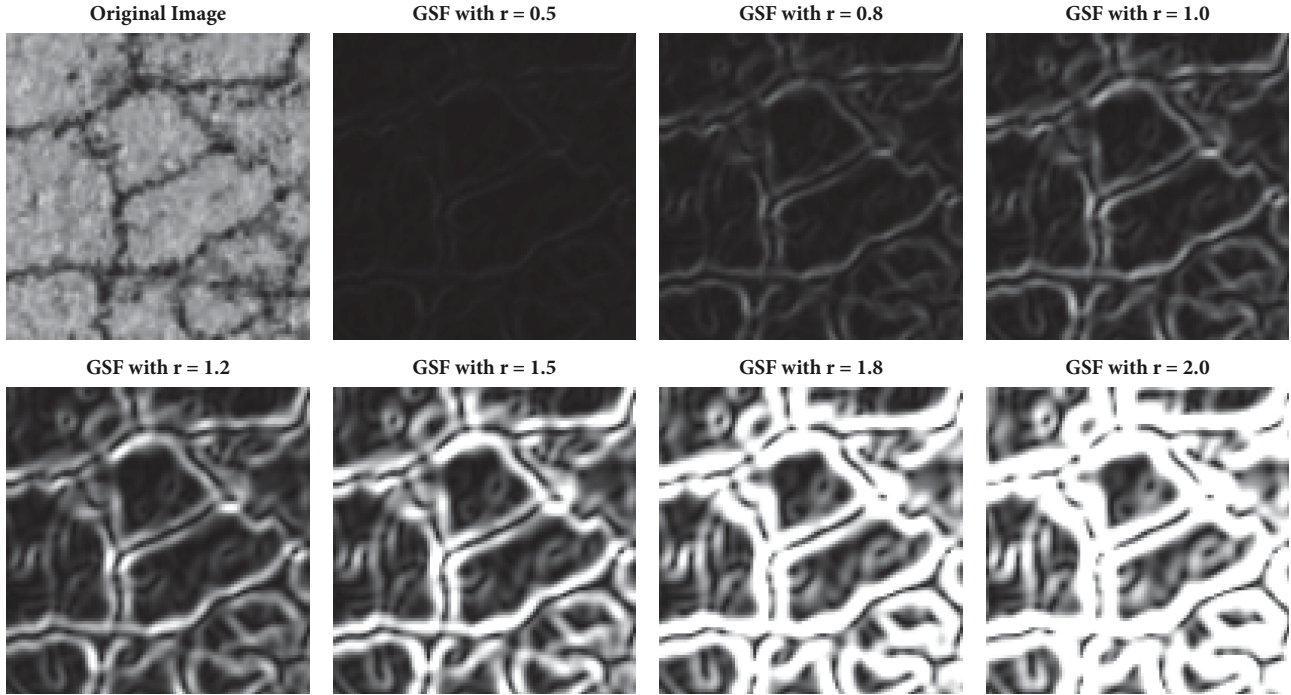


FIGURE 2: The SF responses of a pavement image with different values of Gaussian's standard deviation ( $r$ ).

an image with a diagonal crack. Therefore, to obtain a more discriminative PI-based feature, this study employs the PI in the two diagonal directions, denoted as diagonal projections (DPs) 1 and 2. As can be seen in **Figure 3(b)**, the PI of an image containing a diagonal crack has relatively stable HP and VP; however, one of its two DPs features a peak of intensity.

## 2.2. Machine Learning Approaches Used for Pavement Crack Classification

**2.2.1. Naïve Bayes Classifier (NBC).** NBC is a simple method used for pattern classification. For a two-class problem, this algorithm assigns the input pattern to one of two-class  $C_m$  ( $m = 1, 2$ ). The class label of the input pattern is computed in the following manner [28, 29]:

$$P(C_m | X) = \frac{P(X | C_m) \times P(C_m)}{P(X)} \quad (9)$$

where  $P(C_m | X)$  denotes the posterior probability of the class  $C_m$ .  $P(X | C_m)$  is the likelihood which is the class-conditional probability density function of the input pattern  $X$ .  $P(C_m)$  is the prior probability of the class  $C_m$ .  $P(X)$  denotes the evidence factor.

The evidence factor  $P(X)$  is basically a scale factor employed to guarantee that the posterior probabilities sum to one [30].  $P(X)$  is calculated as follows:

$$P(X) = \sum_{m=1}^M P(X | C_m) \times P(C_m) \quad (10)$$

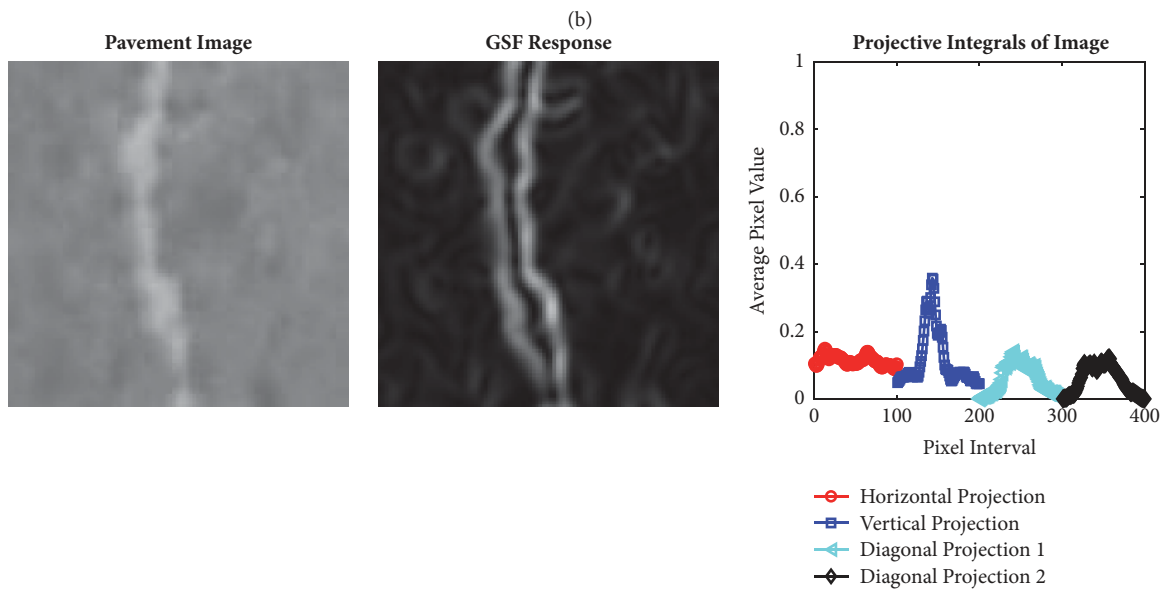
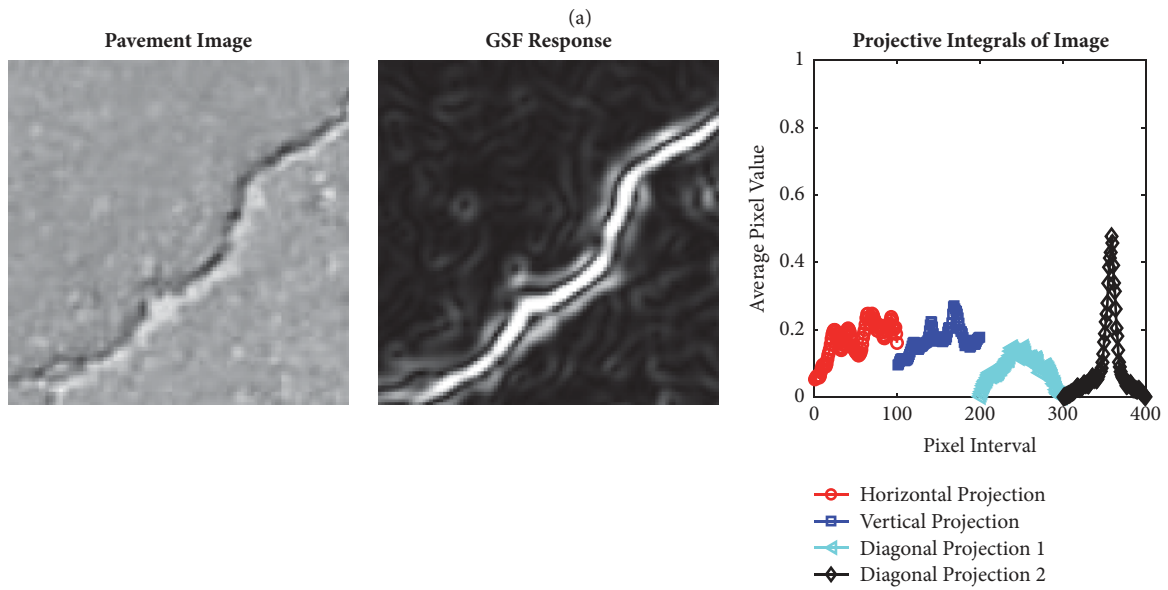
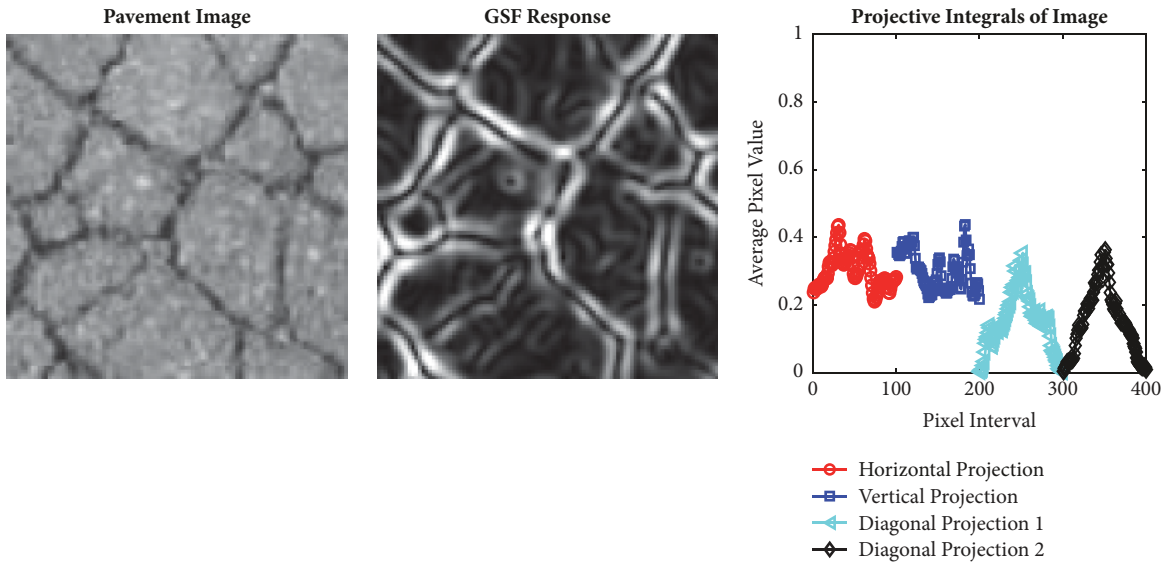
In addition, it is usually shown that the input pattern  $X$  is a  $D$ -dimensional vector. Thus, each element of  $X$  is denoted as  $X_j$  where  $j = 1, \dots, D$ . To compute  $P(X | C_m)$ , NBC relies on the assumption that the probability distributions of attributes  $X_j$ , within each class, are independent of each other [29]. Hence, the class-conditional density is expressed in the following manner:

$$P(X | C_m) = \prod_{j=1}^D P(X_j | C_m) \quad (11)$$

where  $P(X_j | C_m)$  is the probability distribution of the attributes  $X_j$  in a particular class  $C_m$ . Moreover, the density  $P(X_j | C_m)$  is assumed to be a Gaussian distribution.

**2.2.2. Classification Tree (CT).** CT, proposed in Breiman et al. [31], is a popular method for data classification [32, 33]. This algorithm is a method for discovering structural patterns in data and presenting the data in the form of a tree-like structure. In the training phase, a CT model is established by splitting subsets of the collected data set using all input variables to build two child nodes [34]. The most appropriate input variable is selected via an impurity function. The CT algorithm aims at creating data subsets which are as homogeneous as possible for each class label. The Gini impurity function is often employed to quantify the data homogeneous property; this function is expressed as follows [35]:

$$P = G_1 G_2 \quad (12)$$



(c)  
FIGURE 3: Continued.

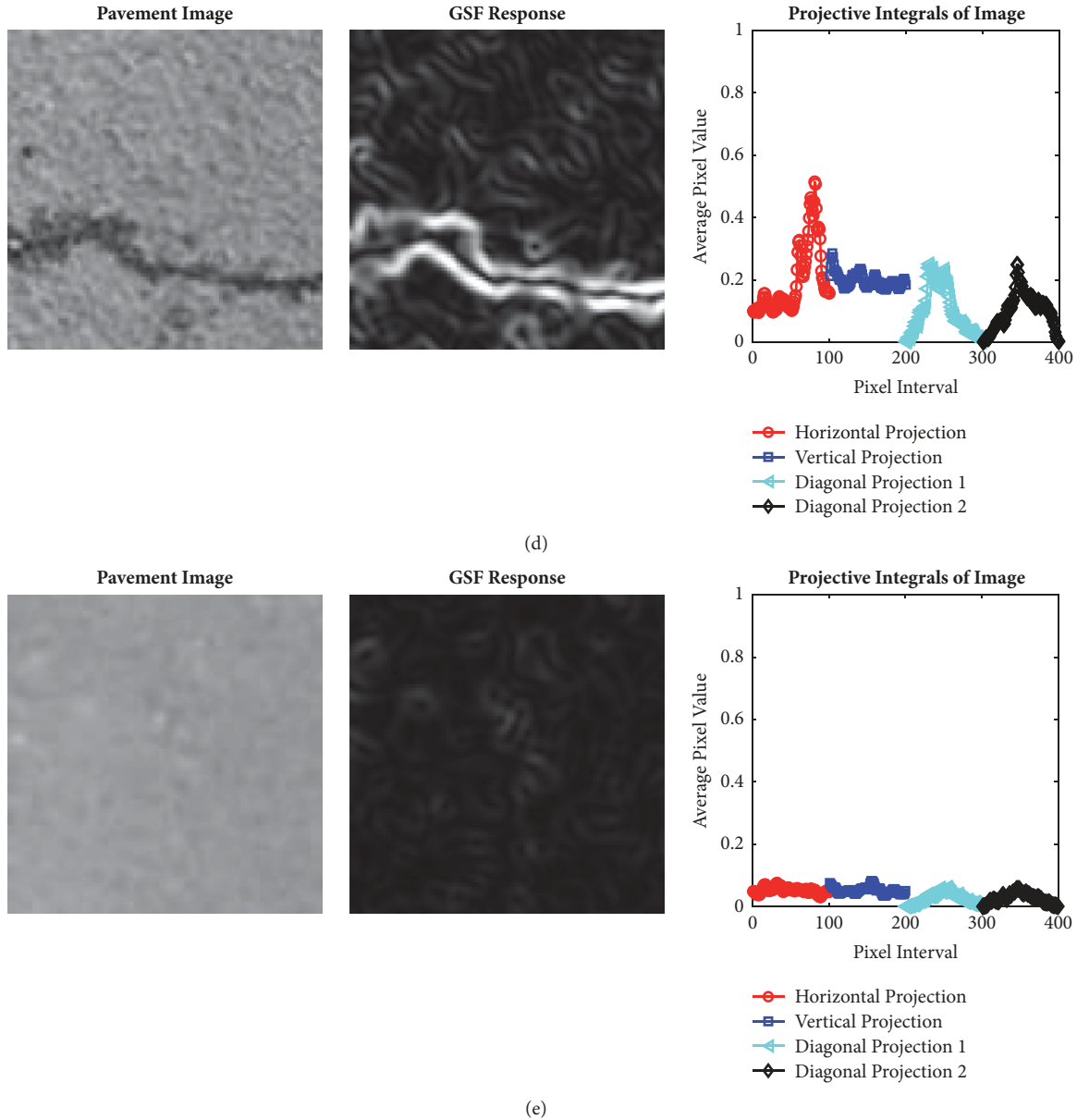


FIGURE 3: Projection integral of pavement images: (a) alligator crack, (b) diagonal crack, (c) longitudinal crack, (d) transverse crack, and (e) noncrack.

where a Gini impurity index of data subset  $k$  is calculated in the following manner [36]:

$$G_k = 1 - \sum_{i=1}^{n_{kc}} p_{ki}^2. \quad (13)$$

Here  $n_{kc}$  denotes the number of classes and  $p_{ki}$  represents the ratio of presence of class  $i$  in this set.

After being constructed, a CT model consists of a root node, a set of internal nodes, and a set of terminal nodes (also called leaves). Each node in the tree represents a binary decision that separates the input variable into either one of the two class labels. Therefore, the data classification process is performed in a top-down manner from the root node to the terminal node.

**2.2.3. Backpropagation Artificial Neural Network (BPANN).** BPANN is a machine learning based classifier inspired from biological neural networks. This algorithm simulates the knowledge acquisition and inference processes of the human brain [37]. Based on previous studies [38–41], BPANN is proved to be highly effective in dealing with complex nonlinear data modeling problems. A BPANN consists of the input, hidden, and output layers. The hidden layers contain a set of artificial neurons; the interconnected artificial neurons play the crucial role of identifying the structure in the data to compute the class labels of each data instance in the output layer.

Using BPANN, a data classification problem boils down to establishing a discrimination function  $f: X \in R^D \rightarrow Y \in$

$R^O$  where  $D$  is the number of input patterns and  $O$  denotes the number of class labels.

The BPANN model used for pattern classification is given as follows [42]:

$$f(X) = b_2 + W_2 \times (f_A(b_1 + W_1 \times X)) \quad (14)$$

where  $W_1$  and  $W_2$  denote weight matrices of the hidden layer and the output layer, respectively;  $N$  is the number of artificial neurons in the hidden layer;  $b_1 = [b_{11}, b_{12}, \dots, b_{1N}]$  and  $b_2$  denotes a bias vector of the hidden layer and of the output layer, respectively.  $f_A$  represents an activation function (e.g., log-sigmoid function).

The value of  $N$  should be selected appropriately to ensure the predictive capability of BPANN. As suggested by Heaton [43],  $N$  can be roughly selected to be  $2D/3 + O$ ; moreover, an BPANN model with  $N > 3D/2$  often brings about no predictive improvements. It is noted that model parameters of BPANN, stored  $W_1$ ,  $W_2$ ,  $b_1$ , and  $b_2$ , are adapted via the backpropagation process [44, 45].

**2.2.4. Radial Basis Function Neural Network (RBFNN).** A RBFNN [46] is a feedforward neural network model that employs radial basis functions as activation functions. Its structure is organized into an input layer, a hidden layer, and one output layer. A RBFNN carries out pattern classification by measuring the similarity between the query inputs with a set of prototypes stored in this model. In essence, each of the  $N_{NR}$  neurons in the hidden layer represents a prototype used for performing classification. A prototype is characterized by a centroid in the learning space.

The data similarity is dependent on the Euclidean distance between the two data points and computed via the radial basis function [47]. The radial basis function  $\phi(\cdot)$  is expressed as follows:

$$z_j(x) = \phi(\|x - c_j\|) \quad (15)$$

where  $c_j$  is the coordination of the  $j^{\text{th}}$  centroid,  $x$  denotes an input pattern, and  $\|x - c_j\|$  is the norm between the input data and the centroid.

In the final model, the RBFNN output is calculated through a sum product of the network's weight and the input vector [48]:

$$y = \sum_{j=1}^M w_j z_j(x) \quad (16)$$

where  $w_j$  is the network weight which can be adapted via the orthogonal least squares learning algorithm [49] used in the network training phase.

**2.2.5. Support Vector Machine (SVM).** SVM, proposed by Vapnik [50], is a binary classification method which originates from the statistical learning theory. Generally, the task at hand is to construct a decision boundary that separates the data into two classes. This decision boundary is generalized from a set of training data points  $\{x_k, y_k\}_{k=1}^N$  with input data  $x_k \in R^n$  and a set of class labels  $y_k \in \{-1, +1\}$ . The SVM first

employs a nonlinear mapping function and the kernel trick to increase the data dimension [51]. Subsequently, this method constructs a hyperplane that plays the role of a decision boundary.

The SVM training phase boils down to solving the following optimization problems:

$$\text{Minimize } J_p(w, e) = \frac{1}{2} w^T w + c \frac{1}{2} \sum_{k=1}^N e_k^2 \quad (17)$$

$$\text{Subjected to } y_k (w^T \phi(x_k) + b) \geq 1 - e_k, \quad (18)$$

$$k = 1, \dots, N, e_k \geq 0$$

where  $w \in R^n$  is a normal vector to the hyperplane and  $b \in R$  denotes the model bias;  $e_k > 0$  is a slack variable;  $c$  represents a penalty constant; and  $\phi(x)$  is a nonlinear mapping from the input space to the high-dimensional feature space.

Based on the kernel trick, it is not necessary to find an expression of  $\phi(x)$ . It is only required to compute the dot product of  $\phi(x)$  in the input space; this dot product is called a kernel function expressed as follows:

$$K(x_k, x_l) = \phi(x_k)^T \phi(x_l) \quad (19)$$

The widely employed kernel function is the radial basis function (RBF) [52, 53]. Find the optimal solution; the optimization problem in (17) is converted into its dual form which is essentially a quadratic programming problem [51]. Accordingly, the SVM model employed for classification problem is compactly shown as follows:

$$y(x_l) = \text{sign} \left( \sum_{k=1}^{SV} \alpha_k y_k K(x_k, x_l) + b \right) \quad (20)$$

where  $\alpha_k$  denotes the solution of the dual form of the optimization problem given in (17). SV is the number of support vectors which is actually the number of positive  $\alpha_k$ .

**2.2.6. Least Squares Support Vector Machine (LSSVM).** First described by Suykens and Vandewalle [54], LSSVM is a powerful approach for data classification. This machine learning method can be considered to be a least squares version of the standard SVM proposed by Vapnik [50]. A significant advantage of LSSVM is that its training phase is accomplished by solving a system of linear equations instead of the quadratic programming problem required by SVM. This fact considerably enhances the computational efficiency of LSSVM and superior performance of this machine learning algorithm has been reported in various applications [55, 56].

The LSSVM learning process can be described as follows:

$$\text{Minimize } J_p(w, e) = \frac{1}{2} w^T w + \gamma \frac{1}{2} \sum_{k=1}^N e_k^2 \quad (21)$$

$$\text{Subjected to } y_k (w^T \phi(x_k) + b) = 1 - e_k, \quad (22)$$

$$k = 1, \dots, N$$

where  $w \in R^n$  is the normal vector to the classification hyperplane, and  $b \in R$  denotes the bias;  $e_k \in R$  is error variable;  $\gamma > 0$  represents a regularization constant.

To solve the above constrained optimization, the Lagrangian is applied as follows:

$$L(w, b, e; \alpha) = J_p(w, e) - \sum_{k=1}^N \alpha_k \{y_k (w^T \phi(x_k) + b) - 1 + e_k\} \quad (23)$$

where  $\alpha_k$  denotes a Lagrange multiplier;  $\phi(x_k)$  is a nonlinear mapping function.

Using the KKT conditions for optimality, the previous optimization is equivalent to solving a linear system [57]. Accordingly, the LSSVM model for binary classification can be attained as follows:

$$y(x) = \text{sign} \left( \sum_{k=1}^N \alpha_k y_i K(x_k, x_i) + b \right) \quad (24)$$

where  $\alpha_k$  and  $b$  are the solution of the aforementioned constrained optimization problem.  $K(x_k, x_i)$  is the kernel function which is applied in a similar manner to the standard SVM.

### 3. Acquisition of Pavement Images and the Feature Extraction Process

**3.1. Pavement Image Acquisition.** Because all the six machine learning methods (NBC, DT, BPANN, RBFNN, SVM, and LSSVM) are supervised algorithms, a data set of asphalt pavement images with the corresponding ground truth labels must be prepared to construct and validate the machine learning based crack classification models. To establish the data set, images of asphalt pavement have been collected during field surveys in Da Nang city (Vietnam). Image samples are captured using digital camera at the distance of about 1.2m above the road surface.

To accelerate the computational process, the images are resized to be 100x100 pixels. There are five classes of pavement condition, namely, alligator crack (AC), diagonal crack (DC), longitudinal crack (LC), noncrack (NC), and transverse crack (TC). Each class label contains 300 image samples; therefore, the total number of data instances in the collected data set is 1500. In this study, medium and large cracks are the objects of interest. The image data sets are illustrated in **Figure 4**. For the purposes of model construction and verification, the data set has been divided into two subsets: the training set (80%) and the testing set (20%). The training set is employed to construct the six-machine learning model and the testing set plays the role of novel input patterns to exhibit the predictive performance of the crack classification model.

**3.2. Image Feature Extraction.** This step aims at creating a set of features used by the machine learning approaches in the task of pavement crack classification. The acquired input image is transferred through a series of processing steps

to enhance its representation; the whole feature extraction process is presented in **Figure 5**. The digital image is first processed by MF to remove the unwanted dot noise and partially diminish the background texture of asphalt pavements. The smoothed image is then enhanced by the SF algorithm which has the purpose of highlighting the crack patterns. The map created by the SF response is employed to construct four PIs, namely, HP, VP, and two DP (DP1 and DP2). The DP1 and DP2 are specifically used to deal with diagonal crack recognition. To compute these two DPs, the map of the SF response is rotated with the angles of +45 and -45 to create two rotated SF maps (demonstrated in **Figures 6** and **7**). The two DP1 and DP2 are obtained by computing the HPs of the two rotated SF maps. As can be shown in **Figure 6**, if the angle between the crack line and the horizontal axis is +45°, the intensity of DP2 has one peak value. On the contrary, the DP1 features one peak value of intensity if the angle between the crack line and the horizontal axis is -45° (demonstrated in **Figure 7**).

Since the image size is 100x100, the number of features created by the four PIs is 400. This number of features is relatively large and can impose certain difficulty for the six machine learning algorithms due to the curse of dimensionality [58]. Therefore, it can be beneficial to reduce the size of the feature set. To contract the features obtained from the PIs, a simple moving average technique is applied. More specifically, the average value of  $W$  consecutive values along the PIs is computed to create PIs with fewer data points (see **Figure 8**). For instance, if  $W = 10$  then the total number of features in the contracted PIs is reduced from 400 to 40. Observably, the contracted PIs still preserve important characteristics of the original PIs. Moreover, the moving average technique can be useful to diminish local fluctuations appearing in the original PIs. The reduced set of PI-based features then serves as input pattern to characterize the four types of cracks (AC, DC, LC, and TC) as well as the state of no detected cracks (NC).

### 4. Experimental Result and Classification Performance Comparison

As stated previously, the data set including 1500 image samples is used to create and verify the performance of the six machine learning models. The data set is divided into a training set (80%) and a testing set (20%). The first set is employed in the model construction phase; the second set is used to demonstrate the model generalization capability. Since a single run may not reflect the true performance of each machine learning approach due to the randomness in the data selection process, this study repetitively performs the training and testing processes 30 times. The model performance is then evaluated by averaging the outcomes obtained from 30 times of training and testing data sampling processes.

In the feature extraction phase, based on several trial-and-error runs, the most suitable window size of MF, the Gaussian function variance parameter ( $r$ ) used in SF, and the window size ( $W$ ) of PI-based feature dimension reduction are chosen to be [5, 5], 1.5, and 10, respectively.



Crack Patterns	Image samples				
AC					
DC					
LC					
NC					
TC					

FIGURE 4: Examples of the collected image data set.

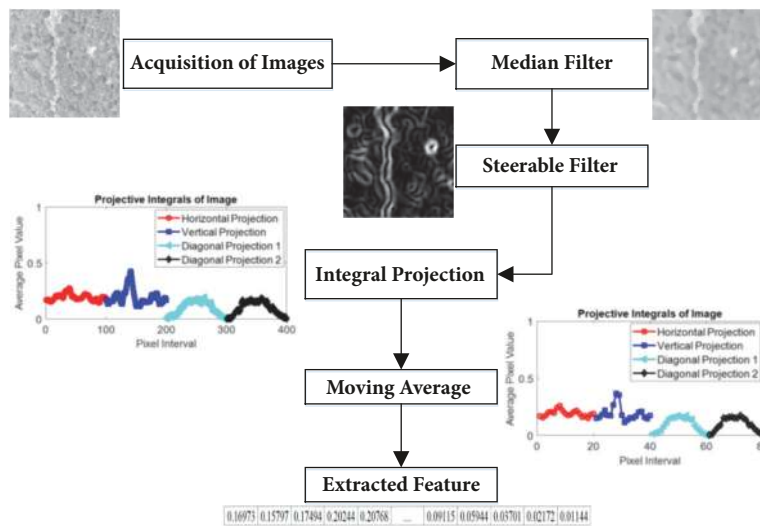


FIGURE 5: The proposed feature extraction process.

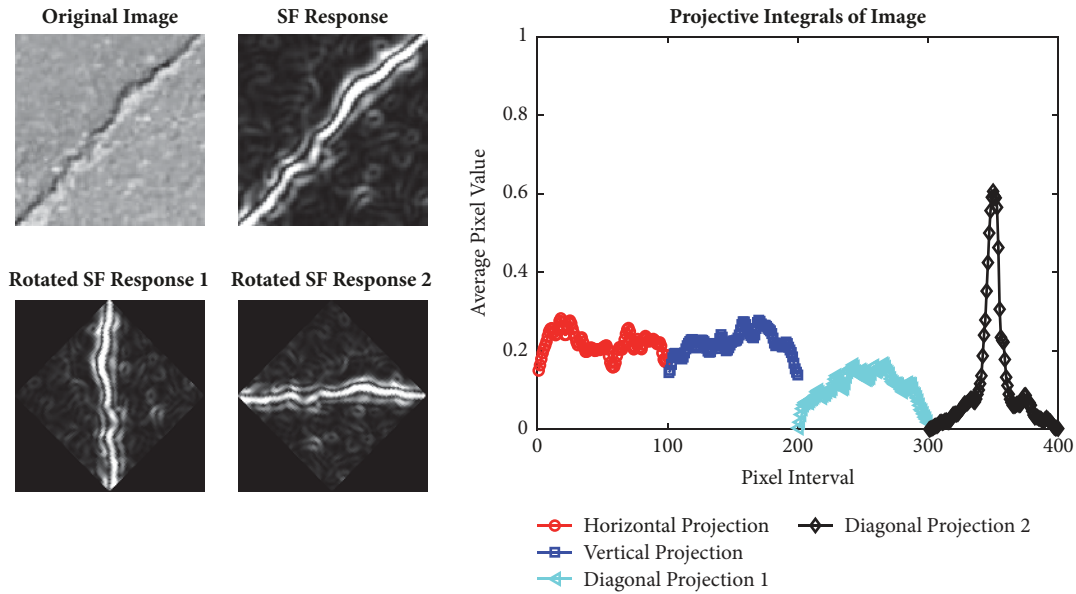


FIGURE 6: Rotated projection integrals: diagonal crack with angle =  $+45^\circ$ .

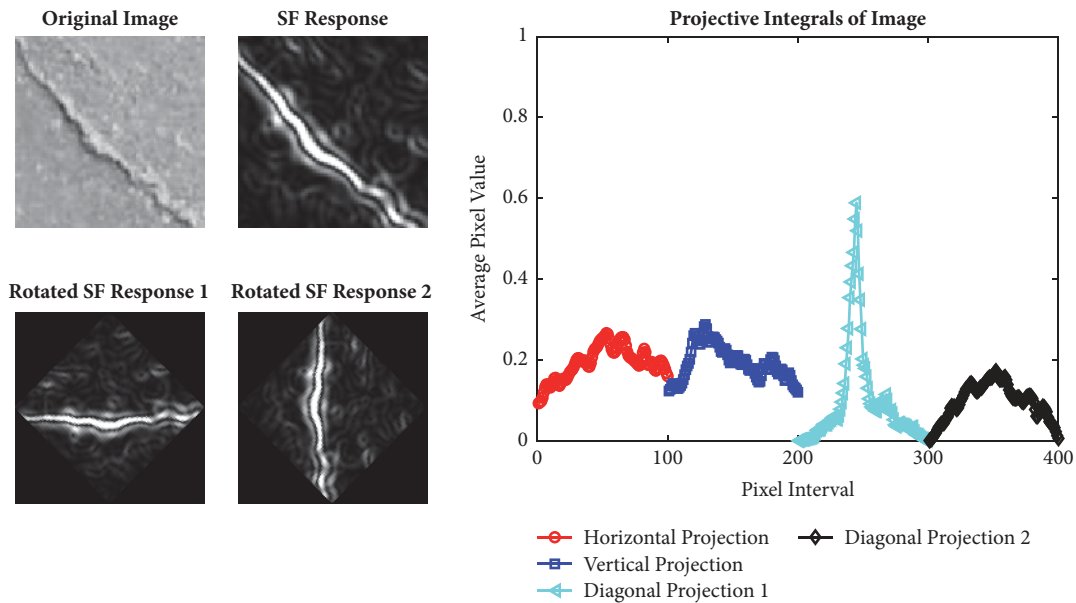
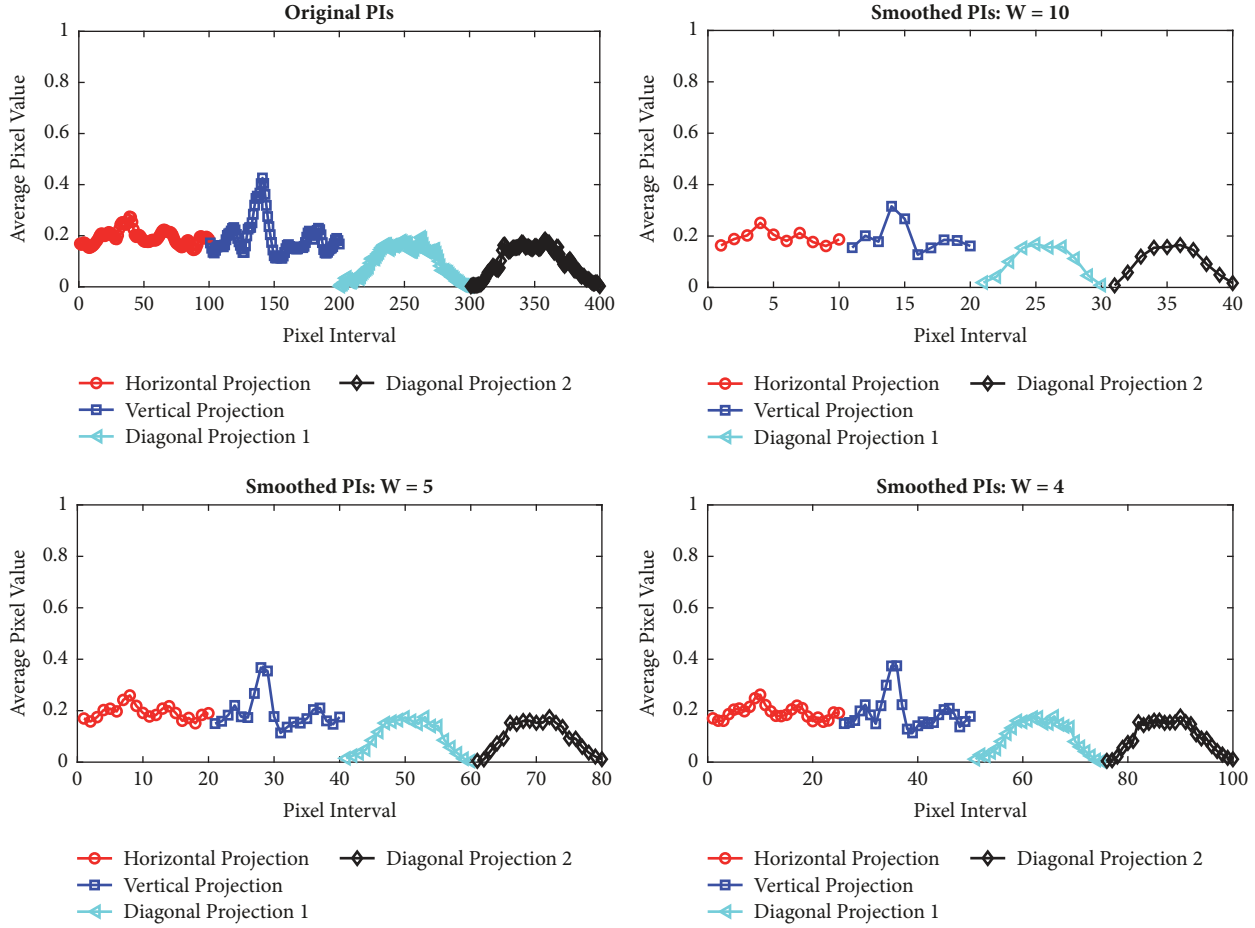


FIGURE 7: Rotated projection integrals: diagonal crack with angle =  $-45^\circ$ .

It is noted that six classification models (NBC, DT, BPANN, RBFNN, SVM, and LSSVM) are employed in the experiment. Besides the two models of BPANN and RBFNN which can be directly used for multiclass classification, the two-class classification versions of NBC, DT, SVM, and LSSVM are extended with one-versus-one (OvO) strategy [59] to deal with the multiclass nature of the pavement crack classification at hand. This strategy is selected due to its good performance; the OvO strategy also helps to avoid the problems of imbalanced data sets [58, 60].

The NBC, DT, BPANN, RBFANN, and SVM models are implemented in MATLAB environment via the Statistics and Machine Learning Toolbox [61]. The LSSVM performs its

training and predicting phases via the toolbox developed by [62]. To employ the DT, BPANN, RBFANN, SVM, and LSSVM models, it is necessary to select their tuning parameters. In this section, the tuning parameters that lead to the best testing performance of models are selected. For the DT model, the minimal number of observations per tree leaf is selected to be 1 as suggested by the MATLAB toolbox [61]. The crucial parameter of BPANN is  $Nr$  (the number of neurons in the hidden layer). In the experiment, as suggested by Heaton [43], this parameter of BPANN is allowed to vary from  $2/3D + O$  to  $1.5D$  (where  $D$  is the number of input variables and  $O$  is the number of the output classes). In addition, the Scaled Conjugate Gradient algorithm with the maximum number of


 FIGURE 8: Smoothed PIs with different values of window size ( $W$ ).

training epochs = 3000 is employed to construct the BPANN model. In the case of RBFANN, the number of neurons in the hidden layer ( $M$ ) and a spread parameter ( $SP$ ) must be specified [61]. For the data set in this study, the suitable value of  $M$  is searched in the ranges of [100, 300] with an interval of 50 and the possible values of  $SP$  are within the set of [0.1, 0.25, 0.5, 0.75, 1, 2.5, 5, 7.5, 10]. The regularization parameter and the kernel function parameter of SVM and LSSVM are set via the grid search process described in the previous work of Hoang and Bui [63].

Moreover, to quantify the predictive capability of the six machine learning models, the classification accuracy rate (CAR) for a particular class  $i$  is calculated as follows:

$$CAR_i = \frac{R_C^i}{R_A^i} \times 100 (\%) \quad (25)$$

where  $R_C^i$  and  $R_A^i$  represent the number of data samples in the class  $i$ th being correctly recognized and the total number of data samples in the class  $i$ th, respectively.

The overall classification accuracy rate (CAR) for all the five class labels is computed as follows:

$$CAR_{Overall} = \frac{\sum_{i=1}^5 CAR_i}{5} \quad (26)$$

The pavement crack classification results of the six machine learning algorithms obtained from the repeated sampling of data with 30 runs are summarized in **Table 1**. Focusing on the average values of CARs, LSSVM has obtained the highest overall CAR (92.62%), followed by SVM (91.91%), BPANN (84.79%), CT (76.84%), NBC (75.54%), and RBFNN (74.81%). The average CARs of LSSVM in predicting data in the classes of AC (95.33%), DC (94.00%), NC (91.17%), and TC (94.94%) are also the most desired outcomes. Only for the data in the LC class, SVM (88.06%) is better than LSSVM (87.67%). LSSVM (91.17%), SVM (91%), RBFNN (87.94%), NBC (87.94%), and BPANN (85.44%) show good performances in predicting the NC class. When predicting the data with the ground truth label of NC, only the CAR of CT is lower than 80%. In addition, **Figure 9** shows the box plots of prediction results of the six machine learning models.

Moreover, to better verify the statistical difference between each pair of machine learning models used in the task of multiclass pavement crack classification, the Wilcoxon signed-rank test (WSRT) is used. It is noted that WSRT is a nonparametric statistical hypothesis test that is commonly employed for confirming the statistical difference of models' predictive capability. Herein, the significance level of the test is set to be 0.05. Using WSRT, the  $p$  values are calculated based on the 30 runs of experiment with each model. If  $p$

TABLE 1: Prediction result comparison.

Statistical Measure	CAR (%)	Pavement Crack Classification Models					
		NBC	CT	BPANN	RBFNN	SVM	LSSVM
Average	AC	65.89	81.72	80.33	70.17	93.78	95.33
	DC	66.94	68.28	84.44	70.28	92.17	94.00
	LC	78.78	73.89	82.94	74.94	88.06	87.67
	NC	87.94	79.67	85.44	87.94	91.00	91.17
	TC	78.17	80.67	90.78	70.72	94.56	94.94
	Overall	75.54	76.84	84.79	74.81	91.91	92.62
Standard Deviation (Std.)	AC	7.18	4.09	5.94	8.07	4.17	3.46
	DC	3.48	6.38	5.88	5.03	3.31	3.02
	LC	4.75	6.21	4.08	5.28	4.78	4.48
	NC	4.24	5.24	5.57	4.37	3.65	3.56
	TC	4.53	4.96	3.27	6.60	3.36	2.08
	Overall	1.90	2.19	2.32	2.44	1.53	1.46

Note: AC: alligator crack; DC: diagonal crack; LC: longitudinal crack; NC: noncrack; TC: transverse crack.

TABLE 2:  $p$  values of pairwise comparison of machine learning models.

Model	Model					
	LSSVM	SVM	BPANN	CT	NBC	RBFNN
LSSVM	--	0.105	0.000	0.000	0.000	0.000
SVM	0.105	--	0.000	0.000	0.000	0.000
BPANN	0.000	0.000	--	0.000	0.000	0.000
CT	0.000	0.000	0.000	--	0.023	0.002
NBC	0.000	0.000	0.000	0.023	--	0.304
RBFNN	0.000	0.000	0.000	0.002	0.304	--

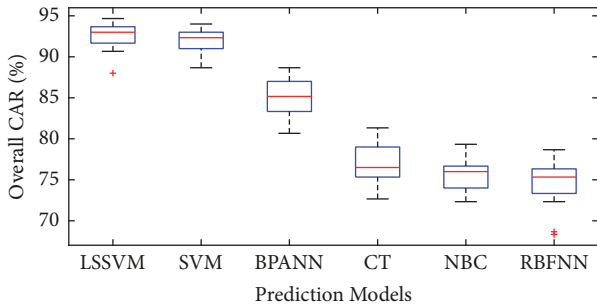


FIGURE 9: Box plots of prediction results of the machine learning models.

value of the test is smaller than 0.05, it is confident to state that the predictive performances of the two machine learning models are statistically different. The  $p$  values obtained from the hypothesis test are reported in Table 2. It is clearly shown that the performances of LSSVM and SVM are significantly better than BPANN, CT, NBC, and RBFNN. The result of BPANN is statistically better than those of CT, NBC, and RBFNN. In addition, the CT model is superior to NBC ( $p = 0.023$ ) and RBFNN ( $p = 0.002$ ). NBC and RBFNN have competitive performances ( $p = 0.304$ ) in modeling the data set at hand.

Moreover, this study also analyzes the effect of the window size parameter ( $W$ ), used in reducing the size of the PI-based feature set, on the performance of the machine learning model. Since LSSVM has the highest overall CAR, this algorithm is selected in this analysis. The values of  $W = 10, 5, 4, 2, 1$  result in the feature number (FN) = 40, 80, 100, 200, 400, respectively. It is noted that  $W = 1$  means that the data set contains all 400 original features obtained from the four PIs. The prediction results corresponding to different sizes of the feature set are reported in Table 3. It can be observed that the reduction of FN by means of moving average method has not deteriorated the prediction accuracy. In fact, as confirmed by WSRT reported in Table 4, the reduction in FN leads to improvements in the model performance. The model using a data set of 40 features has a better CAR (92.62%) than that obtained from the model employing the original 400 features (90.88%). There is only a small difference between the results obtained from LSSVM models that use 40 and 80 features ( $p = 0.923$ ). However, there are statistical differences between the aforementioned results with those obtained from models using 100, 200, and 400 features ( $p < 0.05$ ).

In addition, it is beneficial to investigate the effect of the DPs on the model performance. Herein, the feature set that contains DPs is compared to the feature set that has no DPs. The latter feature set, used in the previous works of Cubero-Fernandez et al. [6] and Hoang and Nguyen [2],

TABLE 3: Prediction results of LSSVM with different value of feature number.

Statistics	CAR (%)	Number of features				
		40	80	100	200	400
Average	AC	95.33	96.72	96.94	95.22	96.33
	DC	94.00	94.89	92.67	90.83	90.56
	LC	87.67	86.56	84.39	86.17	85.56
	NC	91.17	90.17	90.39	89.83	89.33
	TC	94.94	94.28	92.83	93.83	92.61
	Overall	92.62	92.52	91.44	91.18	90.88
Standard Deviation (Std.)	AC	3.46	2.29	2.15	2.43	1.88
	DC	3.02	3.58	3.38	3.78	3.64
	LC	4.48	4.79	5.00	4.80	4.88
	NC	3.56	3.29	3.57	3.37	4.21
	TC	2.08	3.78	3.79	2.44	3.71
	Overall	1.46	1.71	1.46	1.46	1.75

TABLE 4: *p* values of the LSSVM models with different feature number.

Feature Number	Feature Number				
	40	80	100	200	400
40	--	0.923	0.003	0.002	0.000
80	0.923	--	0.021	0.001	0.001
100	0.003	0.021	--	0.564	0.132
200	0.002	0.001	0.564	--	0.401
400	0.000	0.001	0.132	0.401	--

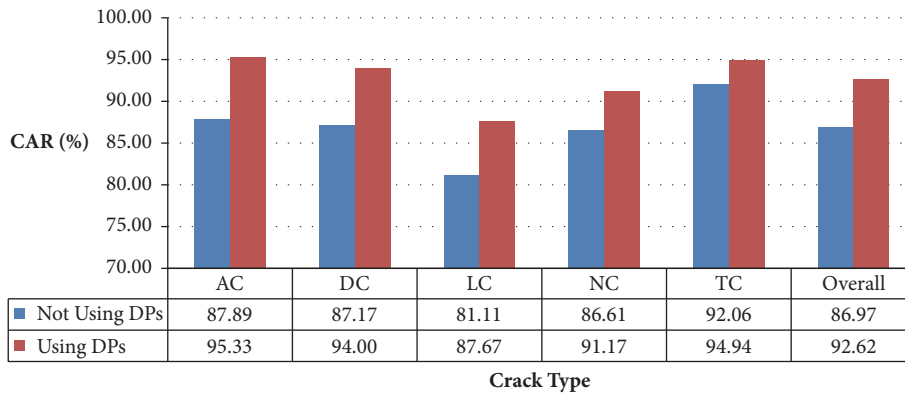


FIGURE 10: Comparison of the LSSVM model results with DPs and without DPs. Note: AC: alligator crack; DC: diagonal crack; LC: longitudinal crack; NC: noncrack; TC: transverse crack.

only consists of the horizontal and vertical PIs. The LSSVM model is also used in this analysis for result comparison. The performances of the LSSVM that uses and does not use DPs are shown in **Figure 10**. Notably, focusing on the DC class, the result of the model using DPs (94.00%) is significantly better than that without the inclusion of DPs (87.17%). Moreover, it is interesting to observe that the model using DPs also outperforms that not using DPs in classifying data of other class labels (AC, LC, NC, and TC).

### 5. Conclusion

To improve the accuracy of the pavement crack classification task, this study constructs an intelligent model that combines image processing and machine learning approaches. The image processing techniques of MF, SF, and PI are used to extract features from digital images. A data set of 1500 images with five class labels of AC, DC, NC, LC, and TC has been prepared. The six machine learning algorithms NBC, CT,

BPANN, RBFNN, SVM, and LSSVM have been employed to construct pavement classification models from the collected data set. Experimental results point out that LSSVM and SVM are the most capable machine learning algorithms for classifying the current data set of pavement images. The performance of LSSVM is slightly better than that of SVM. The overall classification accuracy rates of LSSVM and SVM are 92.62% and 91.91%, respectively. In addition, experiments with LSSVM show that the inclusion of DPs can clearly improve the prediction performance of the machine learning model. Accordingly, the LSSVM using the feature extraction method proposed in this study can be a promising alternative for assisting transportation agencies in the task of pavement condition survey.

### 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 research work.

### References

- [1] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, "Deep Convolutional Neural Networks with transfer learning for computer vision-based data-driven pavement distress detection," *Construction and Building Materials*, vol. 157, pp. 322–330, 2017.
- [2] N. Hoang and Q. Nguyen, "A novel method for asphalt pavement crack classification based on image processing and machine learning," *Engineering with Computers*, 2018.
- [3] S. Li, Y. Cao, and H. Cai, "Automatic Pavement-Crack Detection and Segmentation Based on Steerable Matched Filtering and an Active Contour Model," *Journal of Computing in Civil Engineering*, vol. 31, Article ID 04017045, 2017.
- [4] Y. O. Ouma and M. Hahn, "Wavelet-morphology based detection of incipient linear cracks in asphalt pavements from RGB camera imagery and classification using circular Radon transform," *Advanced Engineering Informatics*, vol. 30, no. 3, pp. 481–499, 2016.
- [5] H. Zakeri, F. M. Nejad, and A. Fahimifar, "Image Based Techniques for Crack Detection, Classification and Quantification in Asphalt Pavement: A Review," *Archives of Computational Methods in Engineering*, vol. 24, 2017.
- [6] A. Cubero-Fernandez, F. J. Rodriguez-Lozano, R. Villatoro, J. Olivares, and J. M. Palomares, "Efficient pavement crack detection and classification," *Eurasip Journal on Image and Video Processing*, vol. 2017, no. 1, 2017.
- [7] M. S. Kaseko, Z. P. Lo, and S. G. Ritchie, "Comparison of Traditional and Neural Classifiers for Pavement-Crack Detection," *Journal of Transportation Engineering*, vol. 120, no. 4, pp. 552–569, 1994.
- [8] H. Lee and J. Kim, "Development of a Crack Type Index Transportation Research Record," *Journal of the Transportation Research Board*, vol. 1940, pp. 99–109, 2005.
- [9] A. Ayenu-Prah and N. Attoh-Okine, "Evaluating Pavement Cracks with Bidimensional Empirical Mode Decomposition," *EURASIP Journal on Advances in Signal Processing*, Article ID 861701, 2008.
- [10] K. Wang, Q. Li, and W. Gong, "Wavelet-Based Pavement Distress Image Edge Detection with a Troun Algorithm Transportation Research Record," *Journal of the Transportation Research Board*, vol. 2024, pp. 73–81, 2008.
- [11] L. Ying and E. Salari, "Beamlet Transform-Based Technique for Pavement Crack Detection and Classification," *Computer-Aided Civil and Infrastructure Engineering*, vol. 25, pp. 572–580, 2010.
- [12] M. Gavilán et al., "Adaptive road crack detection system by pavement classification," *Sensors*, vol. 9628, 2011.
- [13] S. Mokhtari, L. Wu, and H. Yun, "Comparison of Supervised Classification Techniques for Vision-Based Pavement Crack Detection Transportation Research Record," *Journal of the Transportation Research Board*, vol. 2595, pp. 119–127, 2016.
- [14] S. Mathavan, K. Vaheesan, A. Kumar et al., "Detection of pavement cracks using tiled fuzzy Hough transform," *SPIE*, p. 15, 2017.
- [15] Y. Fujita, K. Shimada, M. Ichihara, and Y. Hamamoto, "A method based on machine learning using hand-crafted features for crack detection from asphalt pavement surface images," in *Proceedings of the International Conference on Quality Control by Artificial Vision*, vol. 2017, p. 8, SPIE, 2017.
- [16] K. Zhang, H. D. Cheng, and B. Zhang, "Unified Approach to Pavement Crack and Sealed Crack Detection Using Preclassification Based on Transfer Learning," *Journal of Computing in Civil Engineering*, vol. 32, no. 2, Article ID 04018001, 2018.
- [17] D. Ai, G. Jiang, L. Siew Kei, and C. Li, "Automatic Pixel-Level Pavement Crack Detection Using Information of Multi-Scale Neighborhoods," *IEEE Access*, vol. 6, pp. 24452–24463, 2018.
- [18] S. Chambon and J.-M. Moliard, "Automatic road pavement assessment with image processing: Review and comparison," *International Journal of Geophysics*, vol. 2011, 2011.
- [19] T. B. J. Coenen, A. Golroo, and D. Lo Presti, "A review on automated pavement distress detection methods," *Cogent Engineering*, 2017.
- [20] M. Sonka, V. Hlavac, and R. Boyle, *Image processing, Analysis, and Machine Vision*, Cengage Learning, Printed in the United States of America, 2013.
- [21] E. Arias-Castro and D. L. Donoho, "Does median filtering truly preserve edges better than linear filtering?" *The Annals of Statistics*, vol. 37, no. 3, pp. 1172–1206, 2009.
- [22] H. Rababaah, *Asphalt pavement crack classification: a comparative study of three AI approaches: multilayer perceptron, genetic algorithms, and self-organizing maps*, Indiana University South Bend, 2005.
- [23] W. T. Freeman and E. H. Adelson, "Steerable filters for early vision, image analysis, and wavelet decomposition," in *Proceedings of the Third International Conference on Computer Vision*, pp. 406–415, 1990.
- [24] J. Braz, A. Ranchordas, H. Araújo, and J. Jorge, *Advances in Computer Graphics and Computer Vision*, Springer Verlag, Berlin, Heidelberg, Germany, 2007.
- [25] N. D. Hoang, *An Artificial Intelligence Method for Asphalt Pavement Pothole Detection Using Least Squares Support Vector Machine and Neural Network with Steerable Filter Based Feature Extraction Advances in Civil Engineering*, 2018, In Press.
- [26] M. Jacob and M. Unser, "Design of steerable filters for feature detection using Canny-like criteria," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp. 1007–1019, 2004.

- [27] A. Chouchane, M. Belahcene, and S. Bourennane, "3D and 2D face recognition using integral projection curves based depth and intensity images," *International Journal of Intelligent Systems Technologies and Applications*, vol. 14, no. 1, pp. 50–69, 2015.
- [28] M. Y. Cheng and N. D. Hoang, "Slope Collapse Prediction Using Bayesian Framework with K-Nearest Neighbor Density Estimation: Case Study in Taiwan," *Journal of Computing in Civil Engineering*, vol. 30, no. 1, p. 04014116, 2016.
- [29] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, John Wiley & Sons, 2001.
- [30] D. Tien Bui and N.-D. Hoang, "A Bayesian framework based on a Gaussian mixture model and radial-basis-function Fisher discriminant analysis (BayGmmKda V1.1) for spatial prediction of floods," *Geoscientific Model Development*, vol. 10, no. 9, pp. 3391–3409, 2017.
- [31] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*, Wadsworth and Brooks Monterey, Calif, USA, 1984.
- [32] W. Chen et al., "A comparative study of logistic model tree, random forest, and classification and regression tree models for spatial prediction of landslide susceptibility," *CATENA*, vol. 151, pp. 147–160, 2017.
- [33] B. T. Pham, D. Tien Bui, I. Prakash, and M. B. Dholakia, "Rotation forest fuzzy rule-based classifier ensemble for spatial prediction of landslides using GIS," *Natural Hazards*, vol. 83, no. 1, pp. 97–127, 2016.
- [34] M. S. Alkhasawneh, U. K. Ngah, L. T. Tay, N. A. Mat Isa, and M. S. Al-Batah, "Modeling and Testing Landslide Hazard Using Decision Tree," *Journal of Applied Mathematics*, 2014.
- [35] V.-H. Dang, T. B. Dieu, X.-L. Tran, and N.-D. Hoang, "Enhancing the accuracy of rainfall-induced landslide prediction along mountain roads with a GIS-based random forest classifier," *Bulletin of Engineering Geology and the Environment*, 2018.
- [36] L. Rokach and O. Maimon, *Datamining and knowledge discovery handbook*, Springer, USA, 2010.
- [37] T.-H. Tran and N.-D. Hoang, "Predicting colonization growth of algae on mortar surface with artificial neural network," *Journal of Computing in Civil Engineering*, vol. 30, Article ID 04016030, 2016.
- [38] S. Kiran, B. Lal, and S. S. Tripathy, "Shear strength prediction of soil based on probabilistic neural network," *Indian Journal of Science and Technology*, vol. 9, no. 41, 2016.
- [39] K. Ostad-Ali-Askari, M. Shayannejad, and H. Ghorbanizadeh-Kharazi, "Artificial neural network for modeling nitrate pollution of groundwater in marginal area of Zayandeh-rood River, Isfahan, Iran," *KSCE Journal of Civil Engineering*, vol. 21, no. 1, pp. 134–140, 2017.
- [40] L. Sadowski, M. Nikoo, and M. Nikoo, "Hybrid Metaheuristic-Neural Assessment of the Adhesion in Existing Cement Composites," *Coatings*, vol. 7, no. 4, p. 49, 2017.
- [41] H. Tanyildizi, "Prediction of the Strength Properties of Carbon Fiber-Reinforced Lightweight Concrete Exposed to the High Temperature Using Artificial Neural Network and Support Vector Machine," *Advances in Civil Engineering*, vol. 10, 2018.
- [42] M. H. Beale, M. T. Hagan, and H. B. Demuth, *Neural Network Toolbox User's Guide*, The MathWorks, Inc, 2012.
- [43] J. Heaton, *Introduction to Neural Networks for C#*, Heaton Research, Inc., USA, 2008.
- [44] M. H. Beale, M. T. Hagan, and H. B. Demuth, *Neural Network Toolbox User's Guide*, The MathWorks, Inc, 2018.
- [45] M. T. Hagan, H. B. Demuth, M. H. Beale, and O. D. Jesús, *Neural Network Design*, vol. 2, Martin Hagan, 2nd edition, 2014.
- [46] D. S. Broomhead and D. Lowe, "Radial basis functions, multi-variable functional interpolation and adaptive networks," Technical report Royal Signals and Radar Establishment, 1988.
- [47] I. Aljarah, H. Faris, S. Mirjalili, and N. Al-Madi, "Training radial basis function networks using biogeography-based optimizer," *Neural Computing and Applications*, vol. 29, no. 7, pp. 529–553, 2018.
- [48] D. Yu, Y. Li, H. Sun, Y. Ren, Y. Zhang, and W. Qi, "A Fault Diagnosis Method for Oil Well Pump Using Radial Basis Function Neural Network Combined with Modified Genetic Algorithm," *Journal of Control Science and Engineering*, vol. 7, 2017.
- [49] S. Chen, C. F. N. Cowan, and P. M. Grant, "Orthogonal least squares learning algorithm for radial basis function networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 2, no. 3, pp. 302–309, 1991.
- [50] V. N. Vapnik, *Statistical Learning Theory*, John Wiley & Sons Inc, 1998.
- [51] L. H. Hamel, *Knowledge Discovery with Support Vector Machines*, John Wiley & Sons Inc, 2009.
- [52] B. T. Pham, A. Jaafari, I. Prakash, and D. T. Bui, "A novel hybrid intelligent model of support vector machines and the MultiBoost ensemble for landslide susceptibility modeling," *Bulletin of Engineering Geology and the Environment*, 2018.
- [53] Y. Zhou, W. Su, L. Ding, H. Luo, and P. E. Love, "Predicting Safety Risks in Deep Foundation Pits in Subway Infrastructure Projects: Support Vector Machine Approach," *Journal of Computing in Civil Engineering*, vol. 31, Article ID 04017052, 2017.
- [54] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.
- [55] D. Prayogo and Y. T. T. Susanto, "Optimizing the Prediction Accuracy of Friction Capacity of Driven Piles in Cohesive Soil Using a Novel Self-Tuning Least Squares Support Vector Machine," *Advances in Civil Engineering*, vol. 9, 2018.
- [56] D. Tien Bui, T. A. Tuan, N.-D. Hoang et al., "Spatial prediction of rainfall-induced landslides for the Lao Cai area (Vietnam) using a hybrid intelligent approach of least squares support vector machines inference model and artificial bee colony optimization," *Landslides*, vol. 14, no. 2, pp. 447–458, 2017.
- [57] J. A. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle, *Least Square Support Vector Machines*, World Scientific Publishing Co. Pte. Ltd., 2002.
- [58] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer, New York, NY, USA, 2011.
- [59] A. Rocha and S. K. Goldenstein, "Multiclass from binary: Expanding One-versus-all, one-versus-one and ECOC-based approaches," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 2, pp. 289–302, 2014.
- [60] K. B. Duan, J. C. Rajapakse, and M. N. Nguyen, "One-Versus-One and One-Versus-All Multiclass SVM-RFE for Gene Selection in Cancer Classification," in *Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics*, pp. 47–56, Springer, Berlin, Heidelberg, Germany, 2007.
- [61] Matwork, *Statistics and Machine Learning Toolbox User's Guide*, Matwork Inc., 2018, [https://www.mathworks.com/help/pdf\\_doc/stats/stats.pdf](https://www.mathworks.com/help/pdf_doc/stats/stats.pdf), Date of last access: 04/28/2018.

- [62] K. De Brabanter et al., “LS-SVMlab Toolbox User’s Guide version 1.8,” 2010.
- [63] N. Hoang and D. T. Bui, “Predicting earthquake-induced soil liquefaction based on a hybridization of kernel Fisher discriminant analysis and a least squares support vector machine: a multi-dataset study,” *Bulletin of Engineering Geology and the Environment*, vol. 77, no. 1, pp. 191–204, 2018.






**Hindawi**

Submit your manuscripts at  
[www.hindawi.com](http://www.hindawi.com)

