

Research Article

Concrete Spalling Severity Classification Using Image Texture Analysis and a Novel Jellyfish Search Optimized Machine Learning Approach

Nhat-Duc Hoang ^{1,2}, Thanh-Canh Huynh ^{1,2}, and Van-Duc Tran ^{2,3}

¹Institute of Research and Development, Duy Tan University, Da Nang 550000, Vietnam

²Faculty of Civil Engineering, Duy Tan University, Da Nang 550000, Vietnam

³International School, Duy Tan University, Da Nang 550000, Vietnam

Correspondence should be addressed to Nhat-Duc Hoang; hoangnhatduc@duytan.edu.vn

Received 17 May 2021; Revised 13 September 2021; Accepted 7 November 2021; Published 10 December 2021

Academic Editor: Tomas Hanak

Copyright © 2021 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.

During the phase of building survey, spalling and its severity should be detected as earlier as possible to provide timely information on structural health to building maintenance agency. Correct detection of spall severity can significantly help decision makers develop effective maintenance schedule and prioritize their financial resources better. This study aims at developing a computer vision-based method for automatic classification of concrete spalling severity. Based on input image of concrete surface, the method is capable of distinguishing between a minor spalling in which the depth of the broken-off material is less than the concrete cover layer and a deep spalling in which the reinforcing steel bars have been revealed. To characterize concrete surface condition, image texture descriptors of statistical measurement of color channels, gray-level run length, and center-symmetric local binary pattern are used. Based on these texture-based features, the support vector machine classifier optimized by the jellyfish search metaheuristic is put forward to construct a decision boundary that partitions the input data into two classes of shallow spalling and deep spalling. A dataset consisting of 300 image samples has been collected to train and verify the proposed computer vision method. Experimental results supported by the Wilcoxon signed-rank test point out that the newly developed method is highly suitable for concrete spall severity classification with accuracy rate = 93.33%, $F1$ score = 0.93, and area under the receiver operating characteristic curve = 0.97.

1. Introduction

Spalling is a notable defect widely encountered in surface of reinforced concrete structures (refer to Figure 1). The appearance of spalling significantly deteriorates the integrity and durability of reinforced concrete elements. This defect can be caused by severe servicing environment and loads. More importantly, the appearance of spalls may indicate more serious damages in the internal structure of reinforced concrete elements, e.g., corrosion of steel reinforcement.

Spalling should be detected as earlier as possible due to several reasons. First, spall objects badly affect the aesthetics of building structures and therefore bring about discomfort for occupants. Second, if the layer of concrete cover is removed due to spalling, reinforcing steel bars are exposed to

the environment and this fact expedites the corrosion of the steel bars (as shown in Figure 1(b)). Subsequently, the area and the depth of spall objects increase over time. Third, spalling appeared in ceilings, cladding structures, or concrete beams is particularly hazardous for occupants. The materials broken off from spalled areas can cause significant injuries and even loss of human lives.

As a consequence, periodic visual inspection is necessary to detect and evaluate the severity of spalling defects. Most importantly, deep spalling in which the layer of concrete cover has completely broken off and steel reinforcement has been exposed should be detected timely and requires urgent remedy. In Vietnam as well as in other developing countries, visual inspection performed by human technicians and manual visual data processing are the main approaches for

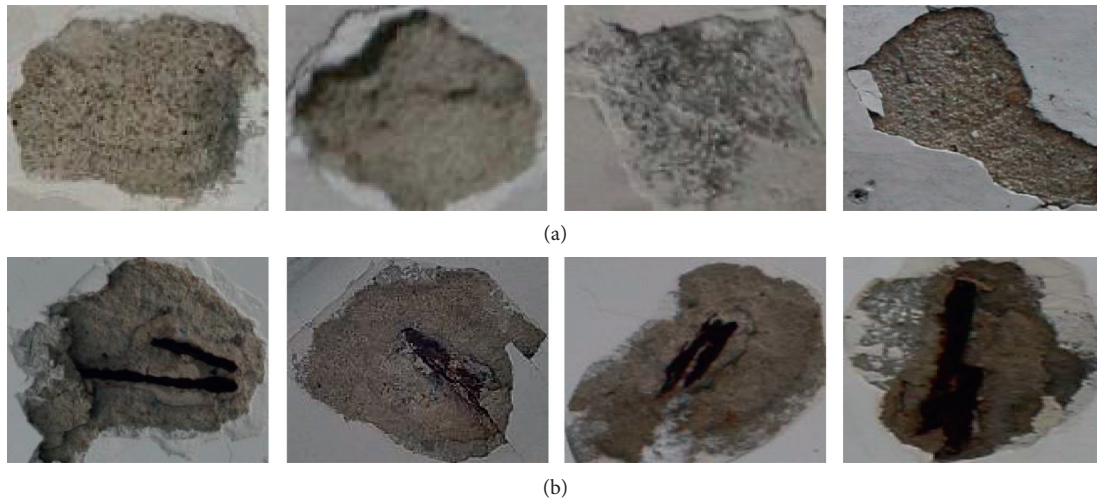


FIGURE 1: Appearances of spall in reinforced concrete surface: (a) shallow spall and (b) deep spall.

spall detection. Although these approaches can help to detect and evaluate the severity of this distress accurately, they are also notoriously known to be labor and time-consuming. With a large surface area of concrete structure, timely inspection and fast visual data processing are virtually impossible for a limited number of inspectors.

Therefore, maintaining good serviceability via periodic visual inspection and evaluation is crucial to keep building environment operational and protect occupants' health. In recent years, due to the availability of low-cost digital cameras as well as a rapid improvement of image processing techniques, computer vision-based structural health monitoring systems have been increasingly used to enhance the productivity of periodic building survey [1–3]. These systems have been demonstrated to be viable tools for building defect detections. They are capable of not only producing acceptable detection accuracy but also guaranteeing consistency in assessment outcomes. The computer vision-based methods yield objective evaluation results; they are not affected by subjective judgments in data processing performed by humans.

Due to the aforementioned advantages, various automated and data-driven methods used for concrete spalling detection have been constructed in the literature. German et al. [4] constructed an automated model for detecting spalled regions on the surfaces of concrete columns based on a local entropy-based thresholding algorithm, a global adaptive thresholding algorithm, and morphological operations; the model is tested with concrete columns during a postearthquake investigation. Dawood et al. [5] proposed a computer vision-based approach for spalling detection and quantification in subway networks; this study employs various image processing techniques including image thresholding, histogram equalization, and filtering in an attempt to detect and quantify the severity of spall objects. This computer vision-based method is validated with a set of 75 image samples and attains an accuracy rate of 89.3%.

Hoang [6] relied on a steerable filter used for feature extraction and machine learning-based data classification to

recognize wall defects including concrete spalling. The method of roughness descriptor based on Hough transformation and similarity analysis is described in Wu et al. [7]; this approach is utilized for recognizing concrete spalling occurring in metro tunnel surface. A model that integrates image processing techniques of texture analysis and machine learning has been proposed in Hoang et al. [8]; a piecewise linear stochastic gradient descent logistic regression has been used to categorized images of concrete surface into two classes of “nonspall” and “spall.”

Abdelkader et al. [9] harnessed the capability of particle swarm optimization metaheuristic coupled with the Tsallis entropy function and discrete wavelet transform to automate the detection of spalling area. Hoang [10] developed an image processing-based spall object detection method relying on Gabor filter for region of interest extraction, texture analysis methods for characterizing feature of concrete surface, and logistic regression models used for data classification; this integrated approach can effectively locate the spall objects but is not capable of classifying spall severity.

Abdelkader et al. [11] developed an entropy-based automated approach for detection and assessment of spalling severities in reinforced concrete bridges; invasive weed optimization-based image segmentation, information theory-based formalism of images, and the Elman neural network are hybridized to formulate the proposed method. Zhao et al. [12] investigated various feature selection strategies used with machine learning models and texture descriptors to detect concrete surface voids.

Recently, deep learning methods have also been applied to tackle the problem of interest. The main advantage of the deep learning models is that the feature extraction phase can be performed automatically [13, 14]. Through various convolutional and pooling operations, useful features such as edges, shapes, texture, and so on can be revealed by the machine and used for the subsequent pattern recognition tasks in a fully connected layer [15]. Wei et al. [16] proposed deep learning-based recognition and quantification for concrete surface bughole; the employed artificial intelligent

method is convolutional neural network (CNN); the main research finding is that the CNN-based model can effectively replace the traditional detection methods carried out by manual inspection.

Another deep learning-based concrete surface void detection method has been put forward in [17]; this method is trained by small-sized image of 28×28 pixels, and its performance outperformed conventional image processing techniques of the Laplacian of Gaussian algorithm and the Otsu method. A CNN-based method used for detecting building defects has been developed in [18]; this method is capable of automatically detecting and localizing key building defects such as mould, deterioration, and stain.

Although CNN-based methods are generally capable tools for detecting spalling and other defects in concrete surface, the deep learning approaches typically demand a large volume of image datasets in order to construct reliable classifiers [13, 19]. This fact requires a great effort in visual data collection and a meticulous data labeling process. In addition, successful implementation of deep learning models also necessitates experience and the trial-and-error process to adjust a significant number of model tuning parameters.

In general, based on recent reviewing works performed by Koch et al. [20]; Feng and Feng [21]; Dong and Catbas [22]; and Yadhunath et al. [23], there is an increasing trend of applying image processing and machine learning for automatically detecting concrete surface distresses including spall. Therefore, investigations of other image processing tools and machine learning frameworks are helpful to provide a broader view on the possibility and capability of computer vision methods in dealing with the task at hand. It is also noted that although various models for spall object detection have been put forward and verified, few studies have constructed spall severity classification models based on two-dimensional digital images. Such models can be immensely helpful for the decision maker and building maintenance agencies to schedule their maintenance and prioritize their budgets spent on treatment of building elements effectively.

In addition, although machine learning methods have been extensively used in computer vision-based structural health monitoring [3, 12, 24–26], hybrid approaches that combine the strengths of machine learning and metaheuristic algorithms are rarely investigated in this field especially for concrete spall recognition. Metaheuristic algorithms can be used to optimize the learning phase of machine learning models and therefore help to achieve better predictive performances [27–33].

Accordingly, the current study aims at contributing to the body of knowledge by constructing a hybrid machine learning and metaheuristic approach used for computer vision-based concrete spall severity recognition. The employed machine learning method is support vector machine [34] because SVM has been proven to be a highly capable tool for pattern recognition especially for nonlinear and multivariate datasets [35–40]. To optimize the performance of SVM, a novel and recently proposed metaheuristic approach of jellyfish search is utilized.

The jellyfish search metaheuristic algorithm is employed to identify the most suitable tuning parameters of the SVM model that yields the desired predictive performance on reinforced concrete spall severity recognition. SVM is used in this study to recognize concrete surface subject to the defects of shallow spall and deep spall. Herein, the first class represents spall objects with its depth smaller than the concrete cover; the latter class contains spall objects having their embedded reinforcement exposed.

Moreover, since the areas of the aforementioned classes have different surfacing properties such as coarseness/fineness, image texture descriptors of statistical measurements of color channels [41], gray-level run length [42, 43], and center-symmetric local binary pattern [44] are used to characterize the surface properties of concrete used for spall severity classification. These texture descriptors have been selected by this study due to their ease of implementation, fast computation, and good discriminative capability [8, 45–50]. In addition, as demonstrated in previous studies [25, 51, 52], the combination of image's color properties and texture is able to bring about good image classification accuracy.

In summary, the main contribution of the current study to the body of knowledge can be stated as follows:

- (i) This study proposes and verifies a computer vision-based method that is capable of categorizing concrete spall severity. This approach can significantly boost the productivity and effectiveness of the periodic survey on the structural health of concrete elements.
- (ii) The proposed approach is a hybridization of JSO metaheuristic and SVM. The JSO algorithm is used to optimize the SVM training phase automatically.
- (iii) The integration of various texture descriptors, which include statistical measurements of color channels, gray-level run length, and center-symmetric local binary pattern, aims at describing the surface feature of concrete surface effectively.
- (iv) The computer vision-based method is trained and optimized automatically with minimum human intervention and effort on parameter fine-tuning.

The subsequent sections of the study are organized as follows. Section 2 reviews the research methodology. The next section describes the structure of the proposed computer vision-based approach employed for spall severity classification. Experimental results are reported in Section 4. Concluding remarks and main research findings are summarized in the last section of the article.

2. Research Methodology

This section of the article presents the research methodology of the current study. The research methodology includes four main sections: image acquisition, image texture computation, model optimization, and model construction. The overall research methodology is depicted in Figure 2. The subsequent parts of this section review the image texture descriptors used for feature extraction, the machine learning, and the metaheuristic algorithm employed for model optimization.

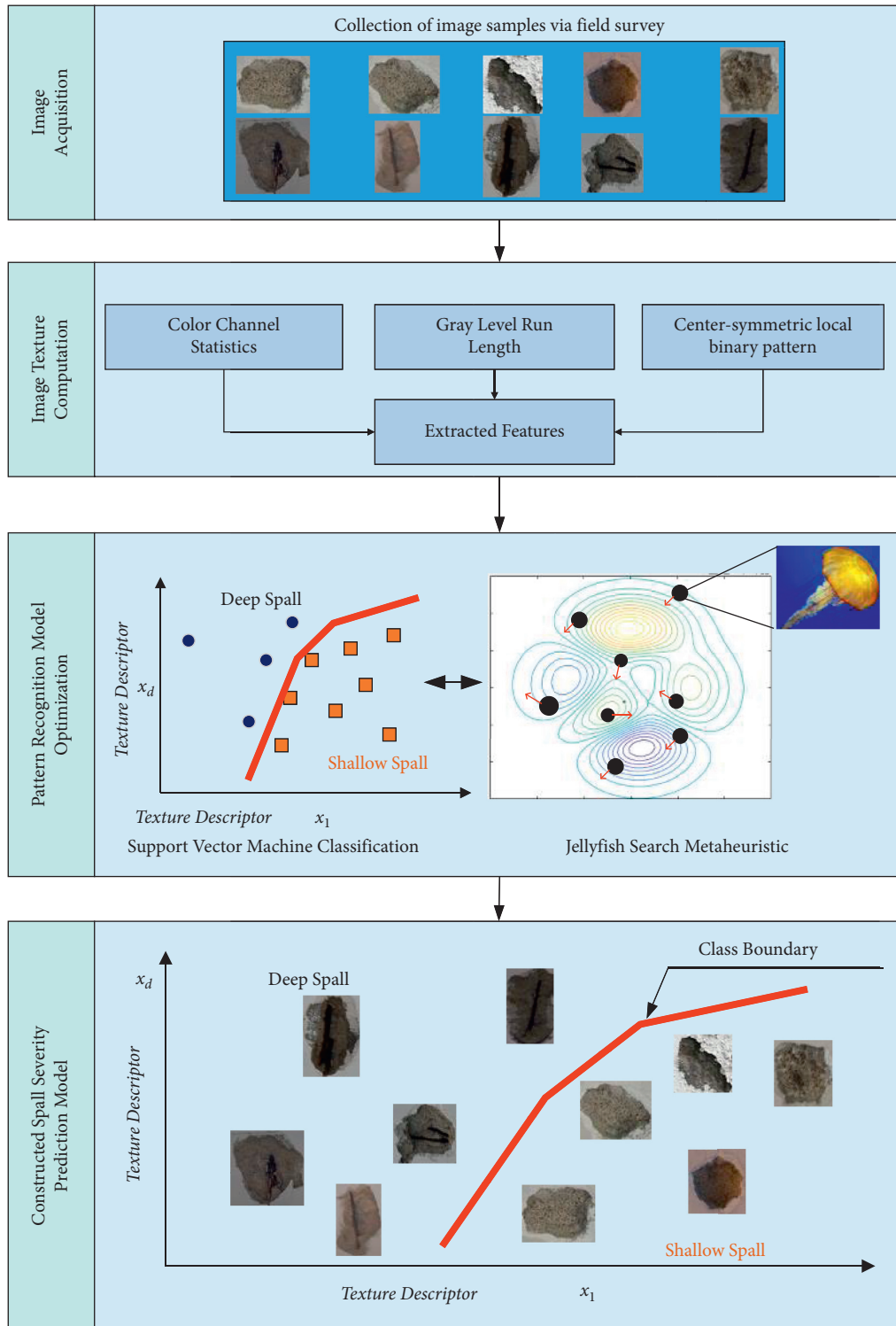


FIGURE 2: Research methodology overview.

2.1. The Employed Image Texture Descriptors. It is observable that surfacing properties of concrete with different categories of spalling severity can be used for pattern classification. Therefore, this study relies on the statistical measurement of image pixel intensity [41], the gray-level run length [42, 43], and the center-symmetric local binary pattern for concrete spall severity classification [44].

2.1.1. Statistical Measurement of Image Pixel Intensity. This study relies on 2-dimensional RGB image samples to recognize concrete spall severity. One image sample has three color channels of red (R), green (G), and blue (B) and is commonly represented by three separated matrices, each of which contains information of pixel intensity in one color channel. To extract the statistical

measurements of image pixel intensity of an image sample I , it is necessary to establish the first-order histogram $P(I)$ describing the statistical distribution of pixels' gray level. Using $P(I)$, the metrics of mean (μ_c), standard deviation (σ_c), skewness (δ_c), kurtosis (η_c), entropy (ρ_c), and range (Δ_c) are computed for a color channel $c = \{R, G, B\}$. Since each color channel yields six statistical measurements, the total number of features describing the pixel intensity distribution of one image sample with three color channels is $6 \times 3 = 18$.

The indices of mean (μ_c), standard deviation (σ_c), skewness (δ_c), kurtosis (η_c), entropy (ρ_c), and range (Δ_c) are obtained in the following equations [51]:

$$\begin{aligned}\mu_c &= \sum_{i=0}^{NL-1} I_{i,c} \times P_c(I), \\ \sigma_c &= \sqrt{\sum_{i=0}^{NL-1} (I_{i,c} - \mu_c)^2 \times P_c(I)}, \\ \delta_c &= \frac{\sum_{i=0}^{NL-1} (I_{i,c} - \mu_c)^3 \times P_c(I)}{\sigma_c^3}, \\ \eta_c &= \frac{\sum_{i=0}^{NL-1} (I_{i,c} - \mu_c)^4 \times P_c(I)}{\sigma_c^4}, \\ \rho_c &= - \sum_{i=0}^{NL-1} P_c(I) \times \log_2(P_c(I)), \\ \Delta_c &= \text{Max}(I_c) - \text{Min}(I_c),\end{aligned}\quad (1)$$

where $NL = 256$ represents the number of discrete intensity values, c is the index of color channels (R , G , or B), and $P(I)$ denotes the first-order histogram of an image.

2.1.2. Gray-Level Run Length (GLRL). GLRL, proposed in [42], is a powerful method for extracting statistical properties of spatial distribution of gray levels. This method utilizes higher-order statistics that analyze the joint distribution of multiple pixels [48]. First, GLRL matrices are computed from a gray-scale image. Subsequently, the occurrence of runs of pixels in a given direction is inspected and statistically quantified. GLRL is useful for characterizing the coarseness or fineness of image region due to the observation that coarse textures are presented by a large number of neighboring pixels featuring the same gray intensity. On the contrary, a small number of neighboring pixels with similar gray-level intensity are observed in fine textures. Given an image of interest, the GLRL constructs a run-length matrix as the number of runs that stems from a location (i, j) of the image in a certain direction [47]. Commonly, for one image sample, four GLRL matrices are computed for the horizontal direction, vertical directions, and two diagonal directions [53].

Let $p(i, j)$ denote a run-length matrix, and the short run emphasis (SRE), long run emphasis (LRE), gray-level non-uniformity (GLN), run length nonuniformity (RLN), and run percentage (RP) are calculated as follows [19, 42]:

$$\begin{aligned}\text{SRE} &= \frac{1}{N_r} \sum_{i=1}^M \sum_{j=1}^N \frac{p(i, j)}{j^2}, \\ \text{LRE} &= \frac{1}{N_r} \sum_{i=1}^M \sum_{j=1}^N p(i, j) \times j^2, \\ \text{GLN} &= \frac{1}{N_r} \sum_{i=1}^M \left(\sum_{j=1}^N p(i, j) \right)^2, \\ \text{RLN} &= \frac{1}{N_r} \sum_{j=1}^N \left(\sum_{i=1}^M p(i, j) \right)^2, \\ \text{RP} &= \frac{N_r}{N_p},\end{aligned}\quad (2)$$

where M and N denote the number of gray levels and the maximum run length, N_r represents the total number of runs and N_p is the number of pixels, and i and j denote the coordinates of a pixel within an image sample.

In addition to the aforementioned indices, Chu et al. [54] proposed the low gray-level run emphasis (LGRE) and high gray-level run emphasis (HGRE) described as follows:

$$\begin{aligned}\text{LGRE} &= \frac{1}{N_r} \sum_{j=1}^N \sum_{i=1}^M \frac{p(i, j)}{i^2}, \\ \text{HGRE} &= \frac{1}{N_r} \sum_{j=1}^N \sum_{i=1}^M p(i, j) \times i^2.\end{aligned}\quad (3)$$

Dasarathy and Holder [55] put forward additional indices extracted from GLRL matrices. These indices are 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); their equations are given by

$$\begin{aligned}\text{SRLGE} &= \frac{1}{N_r} \sum_{j=1}^N \sum_{i=1}^M \frac{p(i, j)}{i^2 \times j^2}, \\ \text{SRHGE} &= \frac{1}{N_r} \sum_{j=1}^N \sum_{i=1}^M \frac{p(i, j) \times i^2}{j^2}, \\ \text{LRLGE} &= \frac{1}{N_r} \sum_{j=1}^N \sum_{i=1}^M \frac{p(i, j) \times j^2}{i^2}, \\ \text{LRHGE} &= \frac{1}{N_r} \sum_{j=1}^N \sum_{i=1}^M p(i, j) \times i^2 \times j^2.\end{aligned}\quad (4)$$

2.1.3. Center-Symmetric Local Binary Pattern (CS-LBP). CS-LBP, proposed in [44], is a modified version of the standard local binary pattern (LBP) [56, 57]. CS-LBP inherits the capability of LBP in describing the texture of an interest region via the distribution of its local structures as well as the intolerance against illumination changes. Both CS-LBP and LBP are widely recognized as simple yet effective texture descriptors [50, 58]. Nevertheless, one major drawback of the original LBP is that it yields a long histogram and therefore produces a large number (i.e., 256) of features to be learnt. A large number of data dimensions usually impose a significant challenge for machine learning model which relies on the data to construct classifiers of interest [59, 60]. Furthermore, the standard texture descriptor is not robust in describing flat image regions [44, 58].

To improve the performance of LBP, CS-LBP is devised by proposing a new scheme of pairwise pixel comparison as shown in Figure 3. Given a patch of $\times 3$ pixels, the CS-LBP compares center-symmetric pairs of pixels in the neighborhood to yield different binary patterns. The function Δ is employed for comparing pairs of pixels; its formula is given by

$$\Delta(x) = \begin{cases} 1, & \text{if } x > T, \\ 0, & \text{otherwise,} \end{cases} \quad (5)$$

where T denotes a thresholding value employed to inspect the significance of the gray intensity differences of 2 pixels.

The center-symmetric pairs of pixel are compared to characterize the local structure of image texture. Therefore, the total number of extracted features is only 16 instead of 256 as required by LBP. In addition, to meliorate the robustness on flat image region, a thresholding value T is used to determine the significance of the gray-level differences between two pixels of interest. The thresholding value T is commonly set to be 3 as suggested in [61]. Accordingly, the formula used to compute the CS-LBP descriptor is given by

$$F_{\text{CS-LBP}}(i, k) = \sum_{i=0}^{(N/2)-1} \Delta(p_i - p_{i+(N/2)}) \times 2^i, \quad (6)$$

where i and k denote the coordination of a pixel within an image sample and $N = 8$ which is the number of neighboring pixels.

2.2. Support Vector Machine Classification (SVC). The pattern recognition method of SVC was first proposed in [34]. This method is a highly effective tool suitable for dealing with classification tasks in high-dimensional space. In this study, SVC is used to categorize the input image data into two class labels of deep and shallow concrete spalling. Let $D = \{(x, y) | x \in \text{Sand } y = f(x)\}$ denote a training dataset. Herein, the input feature refers to numerical data extracted by the aforementioned texture descriptors of the statistical measurement of color channels, GLRL, and CS-LBP. Using SVC, an approximated function $\hat{f}(x): X \rightarrow \{-1, +1\}$ can be established with the label -1 which means “shallow spall” and the label $+1$ which corresponds to “deep spall.”

To cope with nonlinear separable datasets, SVC relies on kernel functions to construct a mapping from the original input space to a high-dimensional feature space within

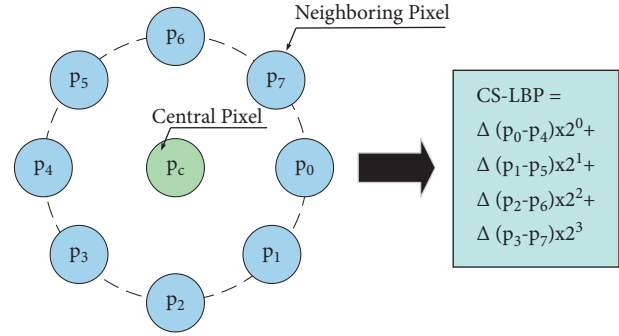


FIGURE 3: Demonstration of the CS-LBP texture descriptor.

which linear separation of datasets is feasible. The data mapping and the construction of a hyperplane used for data separation are demonstrated in Figure 4. To establish such hyperplane, the following nonlinear optimization problem must be solved:

$$\begin{aligned} \text{minimize } J_p(w, e) &= \frac{1}{2}w^T w + C \frac{1}{2} \sum_{k=1}^N e_k^2 \\ \text{subjected to } & y_k(w^T \varphi(x_k) + b) \geq 1 - e_k, \quad k = 1, \dots, N, e_k \geq 0, \end{aligned} \quad (7)$$

where $w \in R^n$ and $b \in R$ are the parameters of the hyperplane, e denotes the vector of slack variables, and C and $\varphi(x)$ represent the penalty coefficient and the nonlinear data mapping function, respectively.

In the formulation of a SVC model, the explicit form of the mapping function is not required. Instead of that, the dot product of $\varphi(x)$ denoting a kernel function $K(x_k, x_l)$ can be obtained. For nonlinear pattern recognition, the kernel function of choice is the radial basis kernel function (RBKF) [62]; its formula is given by

$$K(x_k, x_l) = \exp\left(-\frac{\|x_k - x_l\|^2}{2\sigma^2}\right), \quad (8)$$

where σ denotes a hyperparameter of the kernel function.

2.3. Jellyfish Search (JS) Metaheuristic. As can be shown in the previous section, the establishment of a SVC model used for spall severity classification requires a suitable determination of the penalty coefficient C and the kernel function-based data mapping which is reflected in the tuning parameter σ of the RBKF. The penalty coefficient C indicates the amount of penalty suffered by misclassified data samples during the model training phase; the tuning parameter σ of the RBKF controls the locality of the kernel function which influences the generalization of a SVC model [63].

It is noted that the task of searching for those hyperparameters can be considered as a global optimization problem [28, 32, 64–71]. Moreover, since C and σ are searched in continuous space, the number of parameter combinations is infinitely large. This fact makes an exhaustive search for the best

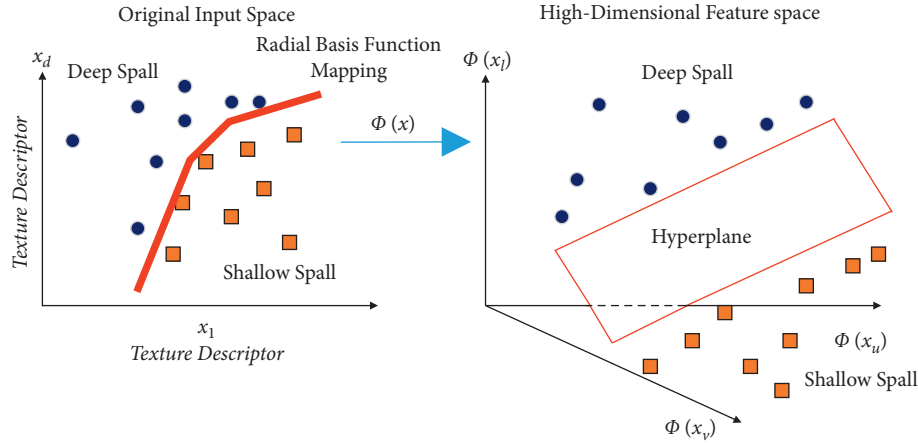


FIGURE 4: Demonstration of the SVC-based spalling severity classification.

hyperparameters infeasible. Therefore, this study employs on the JS metaheuristic to tackle such optimization problem.

The JS metaheuristic, proposed in [72], is a nature-inspired algorithm highly suitable for solving global optimization problems. This metaheuristic is motivated by the behaviors of jellyfish in the ocean. Herein, each searching agent is modeled as a jellyfish. The movements of searching agents in an artificial ocean which is the search space of interest mimic their actual movements in the real-world ocean which are governed by the ocean current, the motion within a swarm, and a time control mechanism for motion mode changing.

Chou and Truong [72] proposed three idealized rules to formulate the JS optimization algorithm. The first rule is that the jellyfish may either follow the ocean current or change their locations within a swarm and there is a time control function that governs their switching of motion type. The second rule is that the jellyfish alter their location in order to search for better food source. The third rule is that the fitness of a location (reflected in the value of the cost function) as well as the jellyfish at this location is proportionate to the amount of food.

After a swarm of jellyfish is randomly generated, the searching agents start to explore and exploit the artificial ocean to search for better food source. The first type of jellyfish movement is following the ocean current. Herein, the direction of the current is expressed as follows:

$$T = X_{\text{Best}} - \beta \times \text{rand} \times \frac{\sum_{i=1}^{\text{NJ}} X_i}{\text{NJ}}, \quad (9)$$

where T denotes the direction of the ocean current, X_{Best} is the location of the current best jellyfish, NJ is the number of jellyfish, $\beta=3$ is the scaling factor, and rand denotes a uniform random number within $[0, 1]$.

Accordingly, the location of a jellyfish is updated via

$$\begin{aligned} X_i(t+1) &= X_i(t) + \text{rand} \times T \\ &= X_i(t) + \text{rand} \times \left(X_{\text{Best}} - \beta \times \text{rand} \times \frac{\sum_{i=1}^{\text{NJ}} X_i}{\text{NJ}} \right). \end{aligned} \quad (10)$$

Inside a swarm, jellyfish demonstrate both passive and active motions [73, 74]. Initially, when a swarm has just been established, the jellyfish tend to exhibit passive motion. Subsequently, the jellyfish have the tendency to follow active motion. The passive motion is mathematically formulated as follows:

$$X_i(t+1) = X_i(t) + \gamma \times \text{rand} \times (\text{UB} - \text{LB}), \quad (11)$$

where $\gamma=0.1$ is a motion coefficient and LB and UB are the lower and upper boundaries of the search variables.

The active motion of jellyfish is determined by the quantity of food stored in a randomly selected location. Generally, jellyfish approach a better food source in a swarm. The location of an individual within a swarm is iteratively revised as follows:

$$\begin{aligned} X_i(t+1) &= X_i(t) + \text{rand} \times \text{DJ}, \\ \text{DJ} &= \begin{cases} X_j(t) - X_i(t), & \text{if } f(X_i) \geq f(X_j), \\ X_i(t) - X_j(t), & \text{if } f(X_i) < f(X_j), \end{cases} \end{aligned} \quad (12)$$

where DJ denotes the direction of a jellyfish, X_i is the target jellyfish, X_j is a randomly selected jellyfish within the swarm, and f denotes the cost function of the problem of interest.

Furthermore, to govern the movement of jellyfish between following the ocean current and moving inside the swarm, the time control mechanism including a time control function $c(t)$ and a constant $C_0=0.5$ is employed. The time control function yields a random value ranging from 0 to 1. If the value of $c(t)$ surpasses C_0 , the jellyfish attach to the ocean current. On the contrary, the jellyfish move within a swarm. The time control function is mathematically described as follows:

$$c(t) = \left| \left(1 - \frac{t}{T_{\text{Max}}} \right) \times (2 \times \text{rand} - 1) \right|, \quad (13)$$

where T_{Max} denotes the maximum number of searching iterations.

2.4. The Collected Image Samples. The objective of this work is to process image samples of reinforced concrete surface for the task of spalling severity classification. To achieve such

objective, this study has carried out field surveys in Danang city (Vietnam) to collect image samples of reinforced concrete surface. This image set includes two categories of shallow spalling and deep spalling. The first category consists of spalling objects in which the depth of spalling is smaller than the concrete cover layer. The second category includes spalling objects in which reinforcing bars have been exposed to the outside environment.

The total number of collected image samples is 300; the number of data in each category is 150 to ensure a balanced classification problem. The collected image samples are illustrated in Figure 5. It is noted that the image samples have been captured by the 18-megapixel resolution Canon EOS M10 and the 16.2-megapixel resolution Nikon D5100. The labels of the image data have been assigned by human inspectors.

3. The Computer Vision-Based Jellyfish Search Optimized Support Vector Classification (JSO-SVC) for Concrete Spalling Severity Classification

This section of the article aims at describing the overall structure of the proposed computer vision-based approach used for automatic classification of concrete spalling severity. The overall structure of the newly developed approach consists of three modules: (i) image texture computation, (ii) JS-based model optimization, and (iii) SVC-based spalling severity categorization based on input image samples. Figure 6 demonstrates integrated modules of the proposed model named as JSO-SVC. The proposed method for automatic classification of concrete spalling severity is an incorporation of image texture descriptions, supervised machine learning-based pattern recognition, and stochastic search-based model optimization.

The image texture description methods of statistical measurements of pixel intensity, GLRL, and CS-LBP are used to extract texture-based features from the collected digital images. The SVC pattern recognizer assisted by the JS stochastic search is employed to establish a class boundary that divided the input feature space into two categories of “shallow spall” and “deep spall.” The role of the JS stochastic search is to optimize the parameter setting of the SVC model. It is noted that the texture computation module has been developed by the authors in Microsoft Visual Studio with Visual C# .NET. Furthermore, the SVC model optimized by the JS algorithm is coded in MATLAB environment with the help of the Statistics and Machine Learning Toolbox [75] and the source code of JS which can be accessed at [76]. The optimized computer vision-based model which relies on the module of texture computation and the JS-SVC model has been coded and compiled in Visual C# .NET Framework 4.7.2 and the built-in functions provided by the Accord.NET Framework [77].

To characterize the properties of concrete surface, this study relies on texture description methods of statistical measurements of pixel intensity, GLRL, and CS-LBP. The first texture descriptor measures statistical indices of the

three color channels (red, green, and blue). For each channel, six indices of mean, standard deviation, skewness, kurtosis, entropy, and range are computed. Therefore, the first descriptor produces $3 \times 6 = 18$ features. Moreover, since one objective of the study is to detect the appearance of reinforcing bars within an image sample, the occurrence of runs of pixels in a given direction can be useful. Thus, it is beneficial to employ the GLRL approach in the feature extraction phase. Four GLRL matrices with orientations of 0° , 45° , 90° , and 135° are computed, each of which yields 11 statistical measurements. Accordingly, the GLRL descriptor produces $4 \times 11 = 44$ features. Finally, the CS-LBP texture description method is computed to characterize the local pattern of image regions. It is noted that to compute the CS-LBP, the number of neighboring pixels around a central pixel is 8. In other words, the radius of this texture descriptor is 1 pixel. As mentioned earlier, the CS-LBP yields 16 texture-based features. Accordingly, the total number of texture-based features used for spall severity classification is $18 + 44 + 16 = 78$. The texture computation processes for the two class labels of interest area are demonstrated in Figure 7.

To train and validate the proposed JSO-SVC model, the collected dataset has been randomly partitioned into a training set (90%) and a testing set (10%). The training set is used for model construction phase; the testing set is reserved for inspecting the model predictive capability when predicting unseen data sample. In addition, prior to the model training phase, the Z-score normalization is commonly employed preprocess the extracted features [78]. Accordingly, all of the extracted features are approximately centered at 0 and have a unit standard deviation. The Z-score equation is given by

$$X_Z = \frac{X_D - M_X}{STD_X}, \quad (14)$$

where X_Z and X_D denote the normalized and the original input data, respectively, and M_X and STD_X represent the mean value and the standard deviation of the original input data, respectively.

In addition, the jellyfish stochastic search with 20 jellyfish is used to assist the SVC training phase. It is noted that the number of optimization iterations of the JS metaheuristic is 100. The JS algorithm's parameters including the scaling factor (β), the motion coefficient (γ), and the parameter C_0 of the time control function are set to be 3, 0.1, and 0.5 according to the suggestions of Chou and Truong [72].

This stochastic search engine optimizes the model selection of the SVC model used for spall severity classification via an appropriate setting of the model hyperparameters. Through operations based on ocean current following and motions within a swarm, a population of jellyfish gradually explores and exploits an artificial ocean and identifies a good set of the penalty coefficient and the RBFK parameter. Herein, the lower and upper boundaries of the searched variables are [0.1, 0.01] and [1000, 1000], respectively. Furthermore, to effectively optimize the machine learning model, a 5-fold cross-validation-based objective function has been employed. This objective function of the JSO-SVC is given by [19]

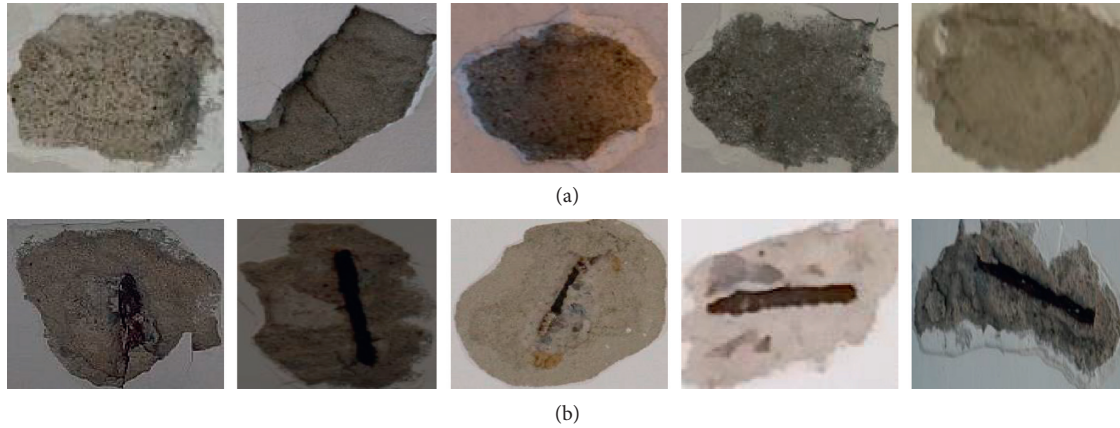


FIGURE 5: Demonstration of the collected image samples: (a) class label -1 (shallow spall) and (b) class label $+1$ (deep spall).

$$F_{\text{JSO-SVC}} = \frac{\sum_{k=1}^5 (\text{FNR}_k + \text{FPR}_k)}{5}, \quad (15)$$

where FNR_k and FPR_k denote false negative rate (FNR) and false positive rate (FPR) computed in the k th data fold, respectively.

The FNR and FPR metrics are given by

$$\begin{aligned} \text{FNR} &= \frac{\text{FN}}{\text{FN} + \text{TP}}, \\ \text{FPR} &= \frac{\text{FP}}{\text{FP} + \text{TN}}, \end{aligned} \quad (16)$$

where FN, FP, TP, and TN refer to the false negative, false positive, true positive, and true negative data samples, respectively.

4. Experimental Results and Discussion

As mentioned earlier, the JS-SVC model has been coded and compiled in Visual C# .NET Framework 4.7.2. Moreover, experiments with the compiled computer program have been performed on the ASUS FX705GE-EW165T (Core i7 8750H and 8 GB Ram) platform. The JS metaheuristic is used to fine-tune the SVC-based spall severity classification approach. After 100 iterations, the JS metaheuristic has located the best values of the search variables as follows: the penalty coefficient = 867.6788 and the RBKF parameter = 58.6156. The best-found cost function value is 1.0696. The optimization process of the jellyfish swarm is demonstrated in Figure 8.

As described in the previous section, the collected dataset which includes 300 data samples has been randomly separated into a training set (90%) and a testing set (10%). Moreover, to reliably evaluate the predictive performance of the proposed JSO-SVC, this study has repeated the model training and testing processes with 20 independent runs. The statistical measurements obtained from these 20 independent runs are employed to quantify the model predictive capability in the task of concrete spalling severity recognition. This repeated model evaluation aims at reducing the

variation caused by the randomness in the data separation process.

In addition, to demonstrate the JSO-SVC predictive performance, the random forest classification (RFC) model [79] and convolutional neural network (CNN) models [80] have been employed as benchmark approaches. The RFC and CNN are selected for result comparison in this study because these two machine learning approaches have been successfully applied in various works related to computer vision-based or nondestructive testing-based structural health monitoring/diagnosis [14, 26, 81–88].

The RFC has been constructed with the MATLAB's Statistics and Machine Learning Toolbox [75]. Adaptive moment estimation (Adam) [89] and root mean square propagation (RMSprop) [90] are the two state-of-the-art approaches for training the deep neural network. The CNN models trained by Adam and RMSprop are denoted as CNN-Adam and CNN-RMSprop, respectively. These two models are constructed with the help of the MATLAB deep learning toolbox [91]. The model structures of the benchmark methods have been identified via several trial-and-error experiments with the collected dataset. The number of classification trees used in the random forest ensemble is 50. In addition, the two CNN models have been trained with 3000 epochs and the batch size of 8. The employed CNN models have been trained with the learning rate parameter = 0.001; moreover, L_2 regularization with the regularization coefficient of 0.0001 has been employed to mitigate model overfitting [91]. To implement the deep neural computing models, the size of the input images has been standardized to be 32×32 pixels. The model structure of the employed CNN models is shown in Table 1.

In addition, to appraise the prediction capability of the proposed JSO-SVC and the employed benchmark approaches, a set of performance measurement metrics is employed in this section. Since the problem of spall severity has been modeled as a two-class classification problem, the indices of classification accuracy rate (CAR), precision, recall, negative predictive value (NPV), $F1$ score, and area under the receiver operating characteristic curve (AUC) [92, 93] are employed to quantify the classification model

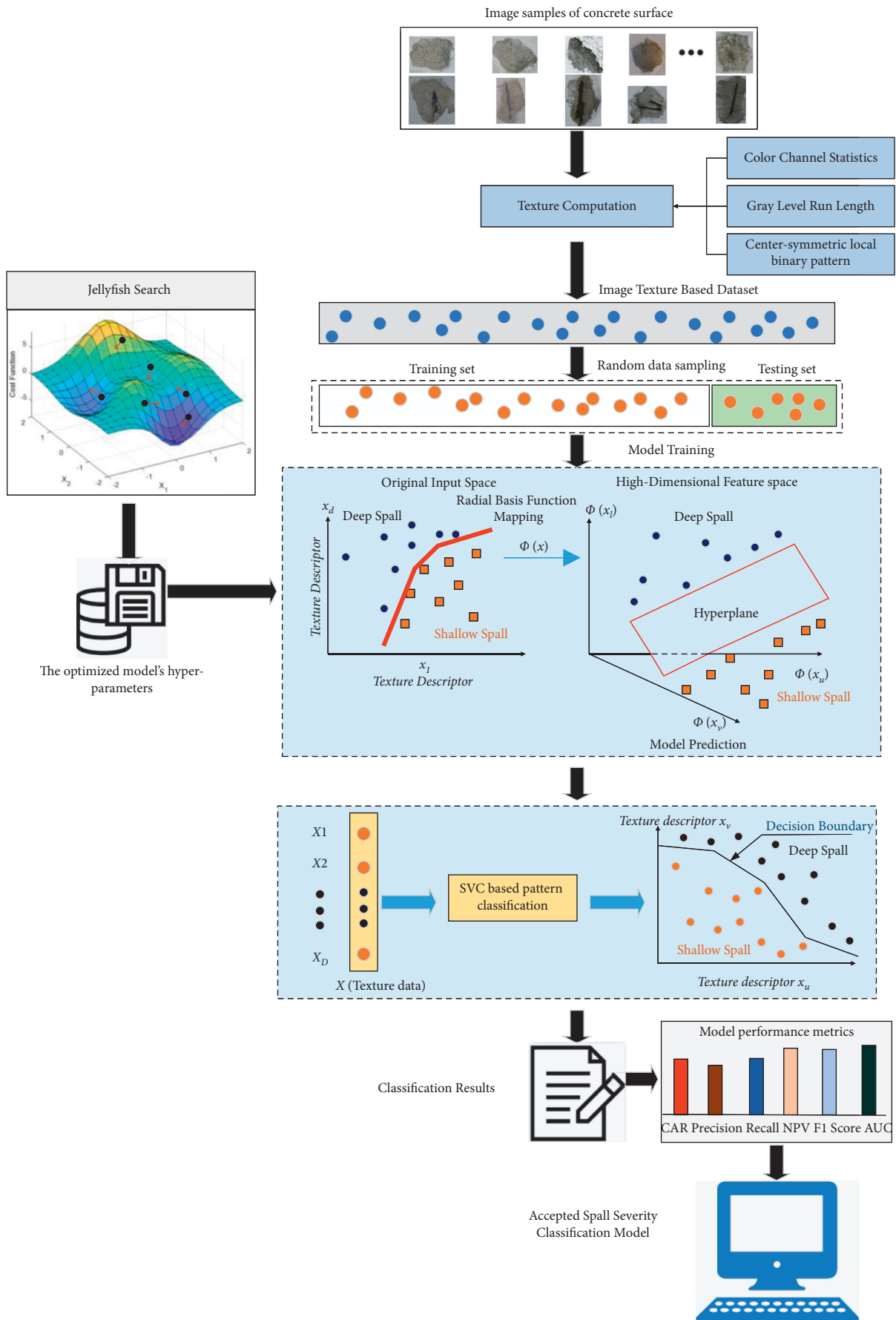
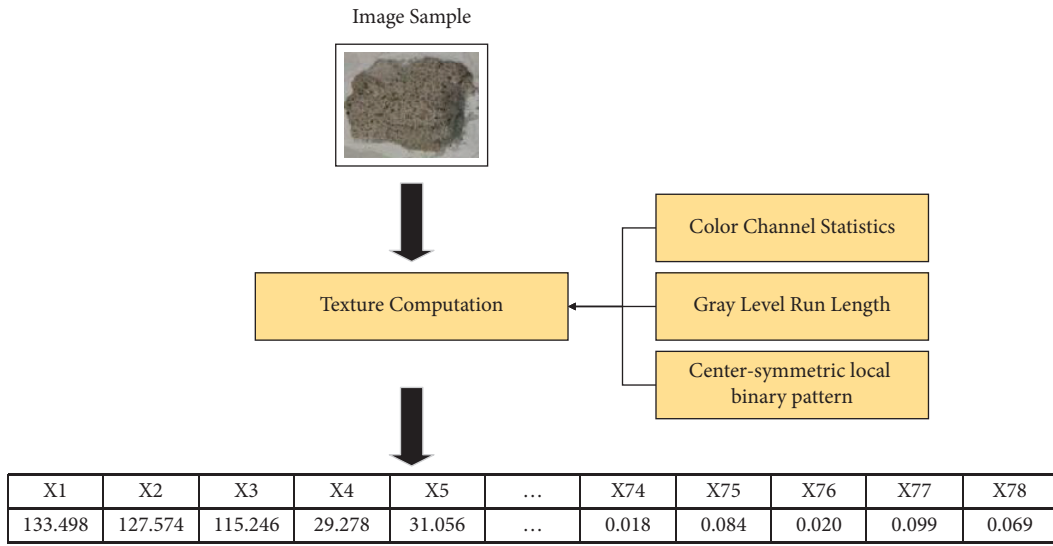
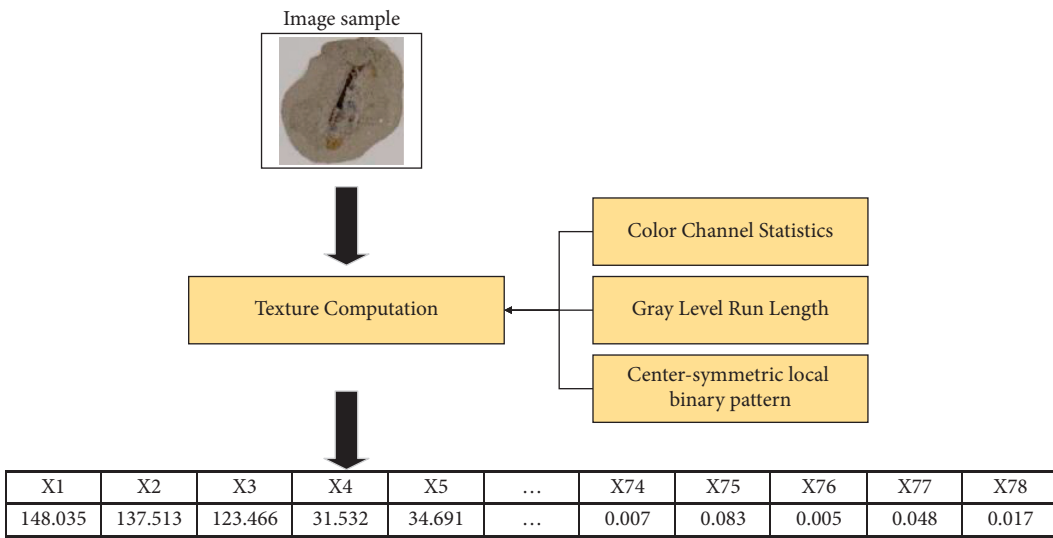


FIGURE 6: The proposed computer vision method for automatic concrete spall severity classification.



(a)



(b)

FIGURE 7: Demonstration of the feature extraction process: (a) a sample of the “shallow spall” class and (b) a sample of the “deep spall” class.

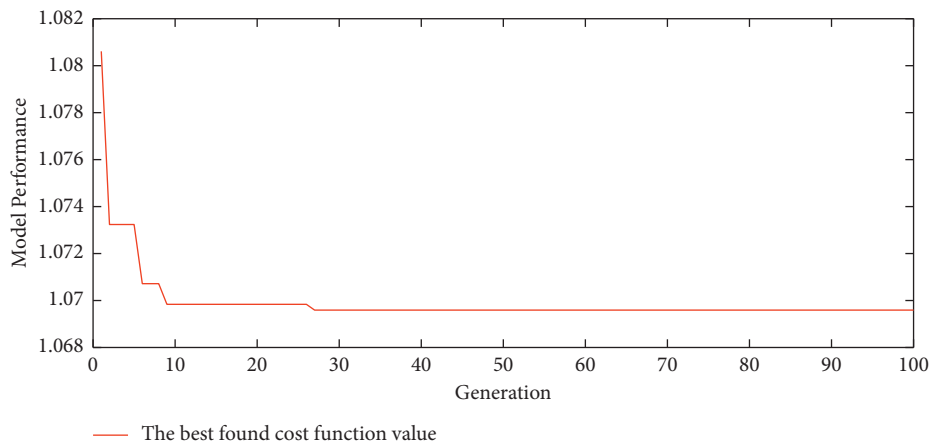


FIGURE 8: The JS-based model optimization progress.

TABLE 1: The structures of the employed CNN model.

CNN layers	Convolutional layers		Pooling layers
	Number of filters	Filter size	Filter size
1	32	8	4
2	32	4	4
3	16	2	2

performance. For the plotting of the receiver operating characteristic curve and computation of the AUC, readers are guided to the previous work of van Erkel and Pattynama [94]. The detailed calculations of CAR, precision, recall, NPV, and $F1$ score are given by [92, 95]

$$\begin{aligned}
 \text{CAR} &= \frac{N_C}{N_A} 100\%, \\
 \text{precision} &= \frac{\text{TP}}{\text{TP} + \text{FP}}, \\
 \text{recall} &= \frac{\text{TP}}{\text{TP} + \text{FN}}, \\
 \text{NPV} &= \frac{\text{TN}}{\text{TN} + \text{FN}}, \\
 \text{F1 score} &= \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}},
 \end{aligned} \tag{17}$$

where N_C and N_A denote the numbers of correctly predicted data and the total number of data, respectively. As mentioned earlier, FN, FP, TP, and TN represent the false negative, false positive, true positive, and true negative data samples, respectively.

The prediction performances of the proposed JSO-SVC and other benchmark methods are shown in Table 2 which reports the mean and standard deviation (Std) of the employed performance measurement metrics. Observable from this table, the proposed hybridization of JS and SVC has attained the most accurate classification of concrete spalling severity with CAR=93.333%, precision=0.932, recall=0.936, NPV=0.963, and $F1$ score=0.933. The model construction phase of the JSO-SVC requires a computational time of 1067.4 s. The computational time of the proposed approach is roughly 3.3 s.

The RFC is the second best method with CAR=87.500%, precision=0.871, recall=0.890, NPV=0.892, and $F1$ score=0.877, followed by CNN-Adam (CAR=81.500%, precision=0.877, recall=0.750, NPV=0.788, and $F1$ score=0.799) and CNN-RMSprop (CAR=79.167%, precision=0.809, recall=0.777, NPV=0.794, and $F1$ score=0.785). With CAR>90% and $F1$ score>0.9, it can be seen that the predictive result of the JSO-SVC is highly accurate. The performance of the decision tree ensemble of RFC with CAR=87.5% and $F1$ score=0.877 is fairly accurate and acceptable. Meanwhile, with CAR values of around 80% and $F1$ score approaching 0.8, the performance of the CNN models is clearly inferior to the machine learning approaches of JSO-SVC and RFC. The boxplots demonstrating the statistical distributions of the models' performance in terms of CAR and $F1$ score

obtained from 20 independent runs are provided in Figures 9 and 10.

Besides the aforementioned metrics, ROC curves and AUC are also important indicators of classification performance. A ROC curve is a graph depicting the performance of a model when classification threshold varies. The horizontal axis of the graph is the false positive rate and the vertical axis of the graph is the true positive rate. The ROC curves of the proposed JSO-SVC and other benchmark models are provided in Figures 11–14. From those curves, the AUC values can be computed. AUC measures the two-dimensional area beneath the ROC curves. This indicator depicts an aggregate evaluation of the model performance with all possible values of the classification threshold. AUC varies between 0 and 1 with 0 indicating a useless classifier and 1 demonstrating a perfect classifier. Observed from the experimental outcomes, the JSO-SVC has also attained the highest AUC of 0.969, followed by RFC (AUC=0.944), CNN-Adam (AUC=0.896), and CNN-RMSprop (AUC=0.855). The boxplot of the AUC results is illustrated in Figure 15.

In addition, to reliably assert the superiority of the newly developed JSO-SVC model used for concrete spalling severity classification, this study has employed the Wilcoxon signed-rank test [96] with the significant level (p value)=0.05. The Wilcoxon signed-rank test is a widely employed nonparametric statistical hypothesis test used for model performance comparison [97]. One significant advantage of this statistical hypothesis test is that it does not require the assumption of normally distributed data [65]. Therefore, the Wilcoxon signed-rank test offers robust statistical power and is likely to yield statistical significant outcome.

The important performance measurement metrics of CAR, $F1$ score, and AUC are subject to this nonparametric hypothesis test. Herein, the null hypothesis is that the means of the prediction performances of two models are actually equal. The Wilcoxon signed-rank test outcomes are reported in Tables 3–5. Observably, with p values <0.05, the null hypothesis can be rejected and the superiority of the proposed hybrid method can be firmly stated.

The experimental results have shown the superiority of the JSO-SVC over deep neural computing approaches of CNN models. It can be seen that although the CNN models have been demonstrated to be powerful methods in various computer vision tasks, their performance largely depends on the size of the training samples [48]. The main advantages of CNN lie in its capability of automatic feature representation via convolutional operators and its hierarchical architecture for learning high-level features from raw data. However, both of these advantages can only be realized with a sufficiently large number of image samples with correct ground truth labeling. As stated in [98], when the number of training samples is insufficient, the performance of deep learning models can be inferior to those of hand-crafted features-based prediction approaches.

For the case in which there are a limited number of data samples, the CNN models have difficulties in properly fine-tuning their internal structures with a huge number of parameters needed to be specified in various hidden layers

TABLE 2: Experimental result comparison.

Phase	Metrics	JSO-SVC		RFC		CNN-Adam		CNN-RMSprop	
		Mean	Std	Mean	Std	Mean	Std	Mean	Std
Training	CAR (%)	95.926	0.574	98.500	0.441	90.278	3.341	89.056	5.207
	Precision	0.950	0.008	0.978	0.005	0.932	0.052	0.915	0.062
	Recall	0.969	0.007	0.993	0.006	0.875	0.079	0.870	0.113
	NPV	0.969	0.007	0.993	0.006	0.887	0.060	0.886	0.076
	F1 score	0.960	0.006	0.985	0.004	0.899	0.039	0.885	0.068
	AUC	0.983	0.002	0.998	0.001	0.966	0.025	0.952	0.039
Testing	CAR (%)	93.333	3.801	87.500	4.941	81.500	5.669	79.167	8.298
	Precision	0.932	0.048	0.871	0.070	0.877	0.090	0.809	0.104
	Recall	0.936	0.062	0.890	0.069	0.750	0.118	0.777	0.136
	NPV	0.935	0.057	0.892	0.062	0.788	0.076	0.794	0.098
	F1 score	0.933	0.042	0.877	0.047	0.799	0.067	0.785	0.096
	AUC	0.969	0.020	0.944	0.046	0.896	0.060	0.855	0.080

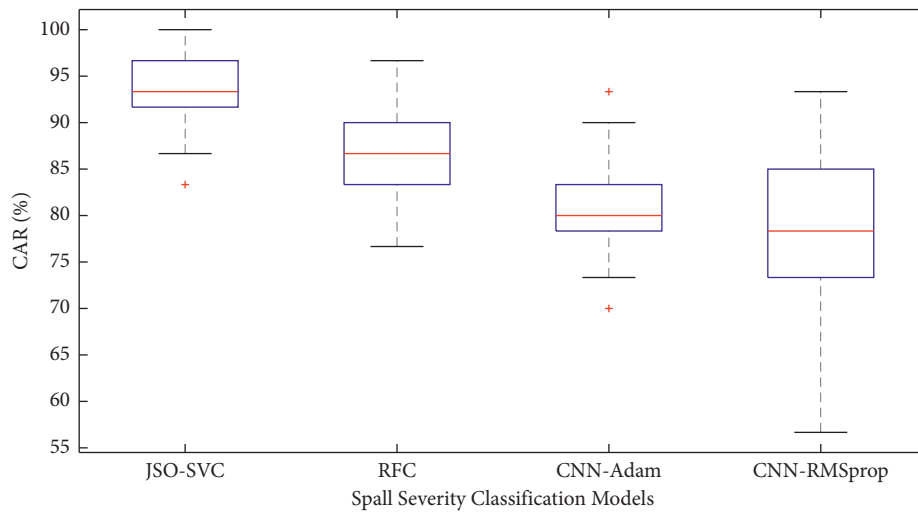


FIGURE 9: Boxplot of CAR values obtained from the employed machine learning models.

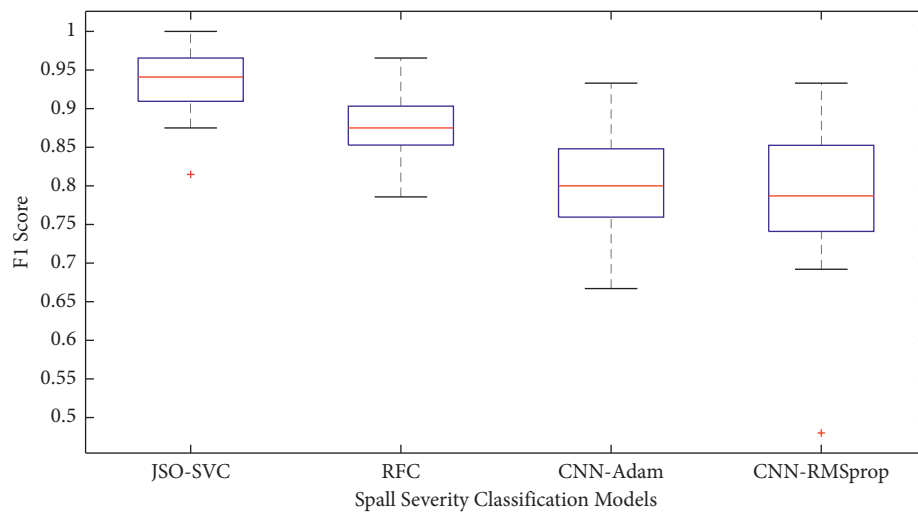


FIGURE 10: Boxplot of F1 score values obtained from the employed machine learning models.

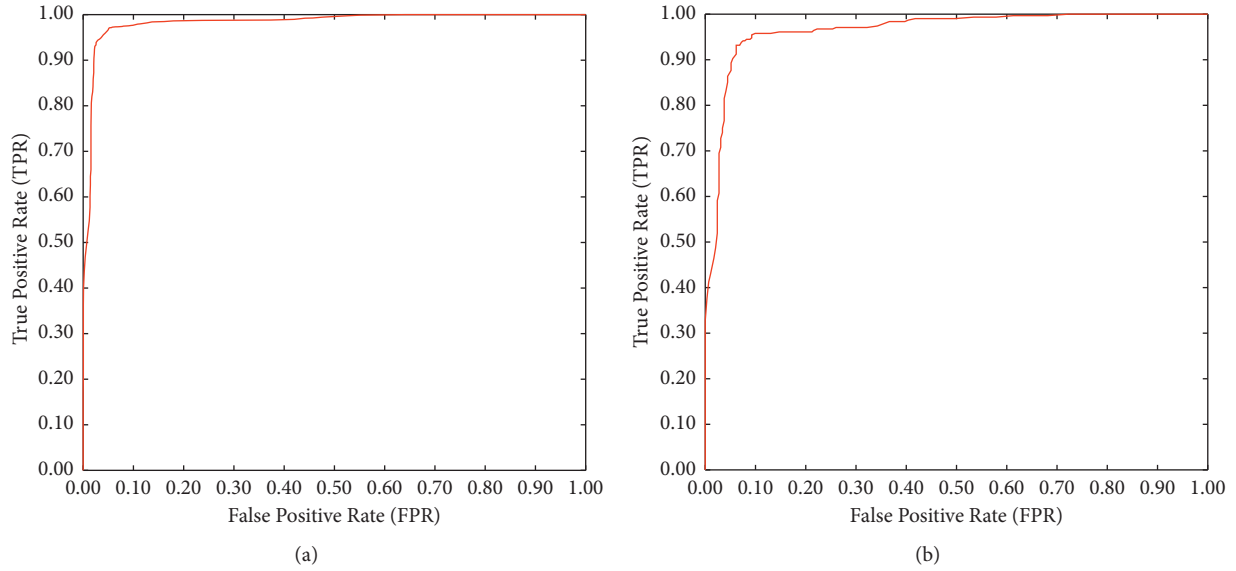


FIGURE 11: ROCs of JSO-SVC: (a) training phase and (b) testing phase.

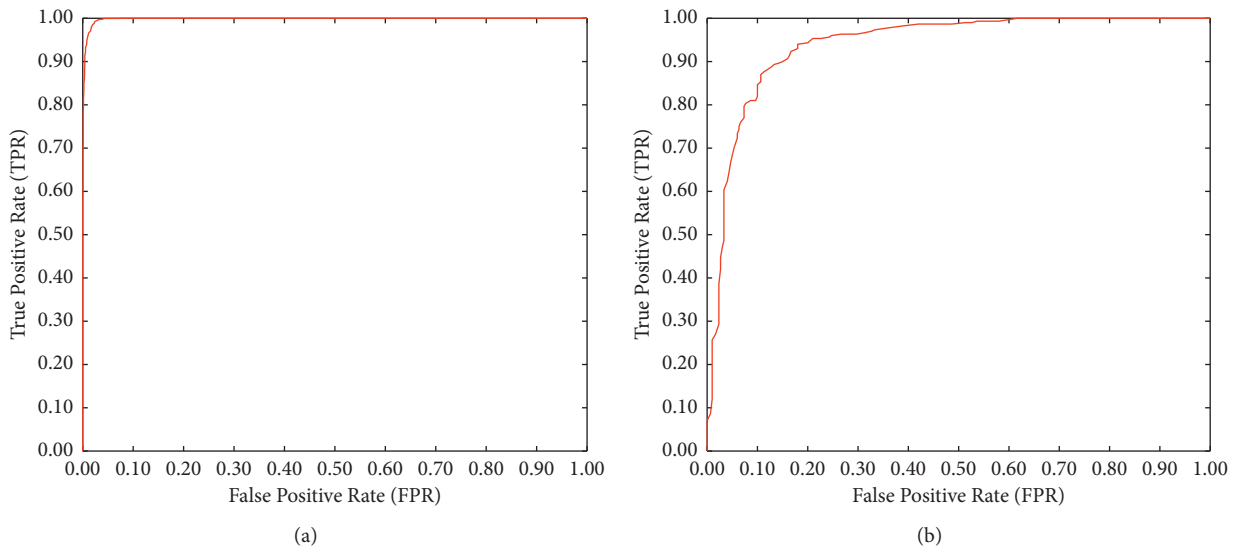


FIGURE 12: ROCs of RFC: (a) training phase and (b) testing phase.

[58, 99]. Therefore, with a dataset of 300 image samples, the hybrid machine learning method of JSO-SVC with texture-based feature extraction is capable of outperforming the CNN methods. The model optimization process via gradient descent algorithms employed by CNN encounters certain difficulty in identifying the fittest set of network parameters because the number of trained parameters greatly outnumbers the data size. This fact is partly reflected in the stability of the predictive outcomes of the deep learning models.

To quantify the stability of the model prediction, this study has employed the coefficient of variation (COV) [100]. This index is defined as the ratio of the standard deviation to the mean. Generally speaking, a small COV value indicates a

small variation of prediction result and is associated with a reliable model. The COV indices of the proposed model, RFC, and the two CNN models are reported in Figure 16. Considering the metrics of CAR, $F1$ score, and AUC, the COV of JSO-SVC (with $COV_{CAR} = 4.07\%$, $COV_{F1-score} = 4.54\%$, and $COV_{AUC} = 2.04\%$) is significantly lower than that of the CNN-Adam ($COV_{CAR} = 6.96\%$, $COV_{F1-score} = 8.35\%$, and $COV_{AUC} = 6.65\%$) and CNN-RMSprop ($COV_{CAR} = 10.48\%$, $COV_{F1-score} = 12.28\%$, and $COV_{AUC} = 9.36\%$).

On the other hand, the proposed approach based on the SVC is the pattern recognizer that lends itself to learning with small or medium-size datasets because the SVC focuses on sparseness property when building a classification model

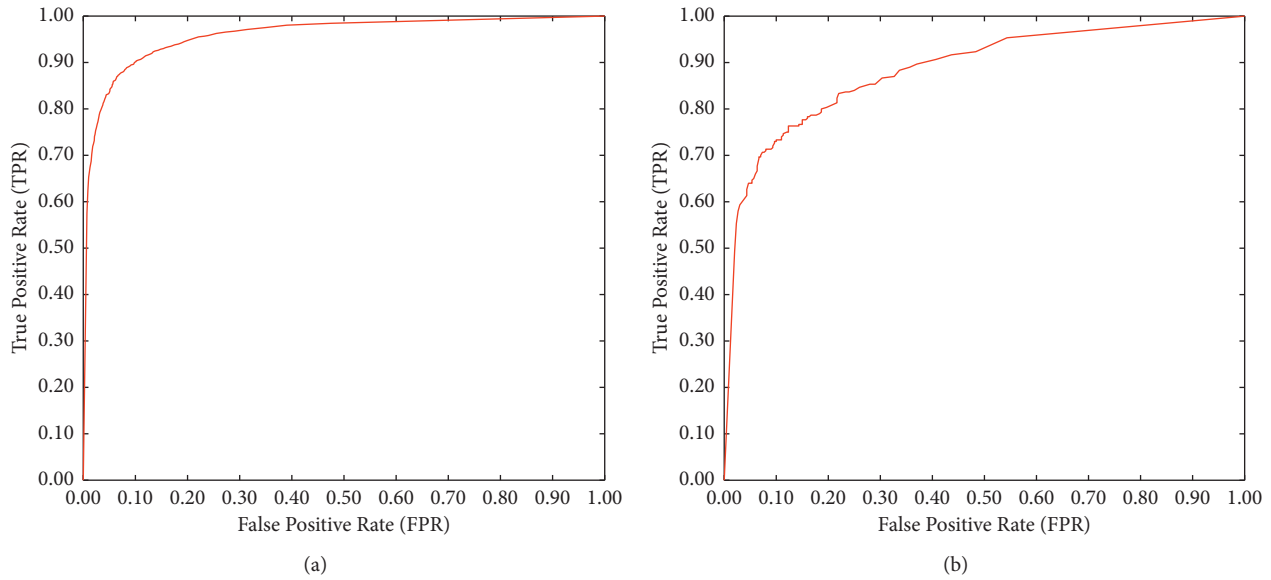


FIGURE 13: ROCs of CNN-Adam: (a) training phase and (b) testing phase.

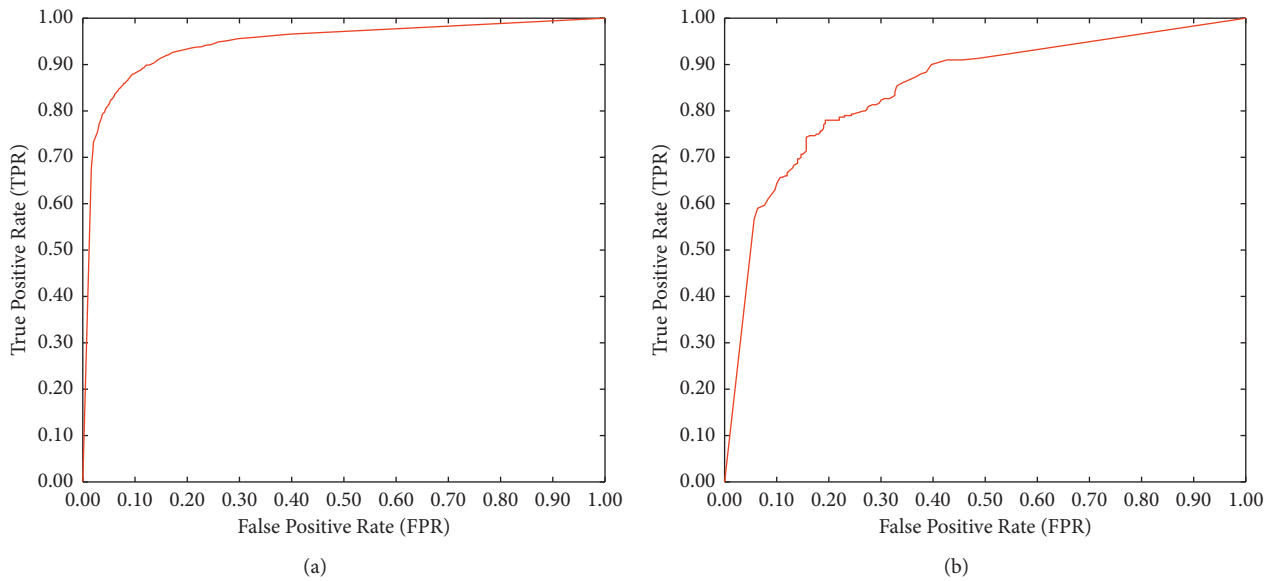


FIGURE 14: ROCs of CNN-RMSprop: (a) training phase and (b) testing phase.

from data. Put differently, the final SVC model only resorts to a small subset of the dataset to construct the classification boundary. The data points contained in such small subset are called support vectors, and they are highly relevant and informative for carrying out the task of concrete spall severity classification. This is a significant advantage of the JSO-SVC because a sparse concrete spall severity classification model is less likely to suffer from data overfitting. This point is clearly demonstrated via the learning performance (CAR = 95.926%) and testing performance (93.333%) of the JSO-SVC. The accuracy rates of the proposed approaches in the training and testing phases are relatively close to each other.

The classification model based on the integration of JSO and SVC also features a high degree of learning stability due to its sparseness property because the sparse model is capable of mitigating the effect of noisy samples within the collected dataset. Moreover, the SVC model construction boils down to solving a quadratic programming problem which can guarantee a learning convergence to a global optimal solution. This feature of the JSO-SVC also facilitates the reliability and stability of the spall severity recognition performance. The aforementioned analysis on COV has revealed these facts. The COV of JSO-SVC, which is less than 5%, is comparatively lower than that of other benchmark models.

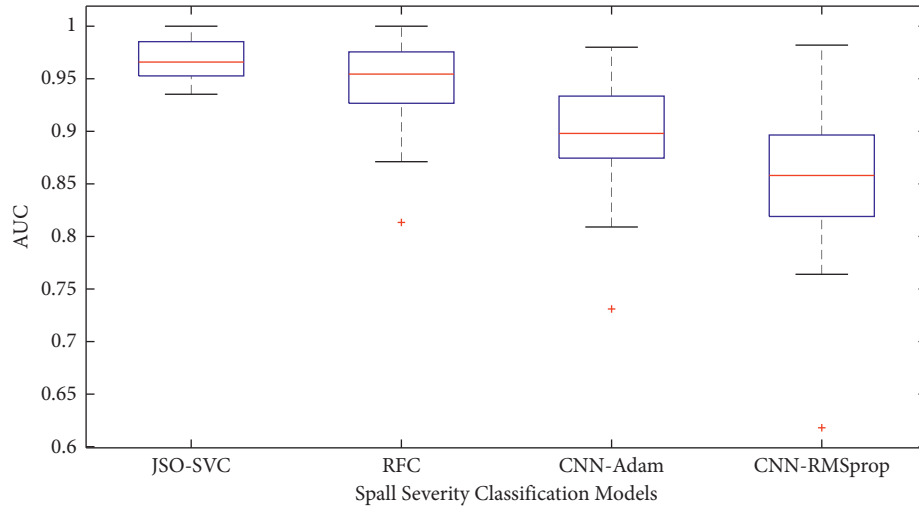


FIGURE 15: Boxplot of AUC values obtained from the employed machine learning models.

TABLE 3: p values obtained from the Wilcoxon signed-rank test results with CAR index.

Models	JSO-SVC	RFC	CNN-Adam	CNN-RMSprop
JSO-SVC	x	0.0022	0.0001	0.0002
RFC	0.0022	x	0.0043	0.0023
CNN-Adam	0.0001	0.0043	x	0.1744
CNN-RMSprop	0.0002	0.0023	0.1744	x

TABLE 4: p values obtained from Wilcoxon signed-rank test results with $F1$ score index.

Models	JSO-SVC	RFC	CNN-Adam	CNN-RMSprop
JSO-SVC	x	0.0028	0.0001	0.0003
RFC	0.0028	x	0.0025	0.0012
CNN-Adam	0.0001	0.0025	x	0.6274
CNN-RMSprop	0.0003	0.0012	0.6274	x

TABLE 5: p values obtained from Wilcoxon signed-rank test results with AUC index.

Models	JSO-SVC	RFC	CNN-Adam	CNN-RMSprop
JSO-SVC	x	0.0859	0.0002	0.0001
RFC	0.0859	x	0.0137	0.0007
CNN-Adam	0.0002	0.0137	x	0.0793
CNN-RMSprop	0.0001	0.0007	0.0793	x

However, one disadvantage of the proposed spall severity recognition model is that the optimization process required determining an optimal set of parameters of the SVC can be costly. The reason is that the SVC-based model training and prediction phases operate inside the cost

function computing phase of the utilized JSO. Another limitation of the JSO-SVC-based spall severity classifier is that it has not been equipped with advanced feature selection. Such drawbacks ought to be addressed in future extensions of the current work.

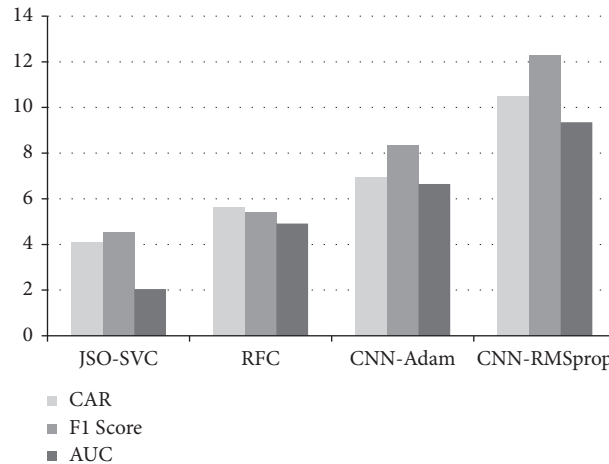


FIGURE 16: Model comparison in terms of the coefficient of variation (COV).

5. Concluding Remarks

This study has proposed and verified a computer vision-based approach for automatic classification of concrete spalling severity. The proposed approach is an integration of image texture analysis methods, metaheuristic optimization, and machine learning-based pattern recognition. The texture descriptors of statistical measurement of color channels, GLRL, and CS-LBP are used to characterize images of concrete surface with respect to color, gray pixel run length, and local structure. With such extracted features, the SVC machine learning optimized by JS metaheuristic is employed to construct a decision boundary that separates the input data into two classes of deep spalling and shallow spalling.

A dataset including 300 image samples has been collected to train the proposed computer vision method. Experimental results point out that the integrated model can help to attain the most desired spall severity classification with CAR = 93.333%, precision = 0.932, recall = 0.936, NPV = 0.963, F1 score = 0.933, and AUC = 0.969. These results are significantly better than those of the benchmark methods including RFC and CNN models. Therefore, the newly developed JSO-SVC can be a potential tool to assist building maintenance agencies in the task of periodic structural health survey. Further improvements of the current approach may include the following:

- (i) The utilization of the hybrid model to detect other concrete surface defects such as crack, bughole, algal colonization, and so on.
- (ii) The employment of other sophisticated texture descriptors for representing characteristics of concrete surface and better dealing with noise in the surface background.
- (iii) Increasing the size of the current dataset to mediorate the applicability of the current method.
- (iv) Investigating the possibility of combing hand-crafted texture-based features with deep learning models used for concrete spalling severity classification.

- (v) Employing advanced techniques of metaheuristic-based model optimization and feature selection to enhance the performance of the spall severity recognition task.

Data Availability

The dataset used to support the findings of this study has been deposited in the repository of GitHub at https://github.com/NDHoangDTU/ConcreteSpallSeverity_JS0_SVC. The first 78 columns of the data are texture-based features extracted from image samples. The last column is the label of the data instances with 0 = “shallow spalling” and 1 = “deep spalling.”

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant no. 107.01-2019.332.

References

- [1] K. Mostafa and T. Hegazy, “Review of image-based analysis and applications in construction,” *Automation in Construction*, vol. 122, Article ID 103516, 2021.
- [2] M. Nain, S. Sharma, and S. Chaurasia, “Safety and compliance management system using computer vision and deep learning,” *IOP Conference Series: Materials Science and Engineering*, vol. 1099, no. 1, Article ID 012013, 2021.
- [3] H. Sun, H. V. Burton, and H. Huang, “Machine learning applications for building structural design and performance assessment: state-of-the-art review,” *Journal of Building Engineering*, vol. 33, Article ID 101816, 2021.
- [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] 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.
- [6] N.-D. Hoang, “Image processing-based recognition of wall defects using machine learning approaches and steerable filters,” *Computational Intelligence and Neuroscience*, vol. 2018, Article ID 7913952, 18 pages, 2018.
- [7] H. Wu, X. Ao, Z. Chen, C. Liu, Z. Xu, and P. Yu, “Concrete spalling detection for metro tunnel from point cloud based on roughness,” *Descriptor Journal of Sensors*, vol. 2019, Article ID 8574750, 12 pages, 2019.
- [8] N.-D. Hoang, Q.-L. Nguyen, and X.-L. Tran, “Automatic detection of concrete spalling using piecewise linear stochastic gradient descent logistic regression and image texture analysis,” *Complexity*, vol. 2019, Article ID 5910625, 14 pages, 2019.
- [9] E. M. Abdelkader, O. Moselhia, M. Marzoukb, and T. Zayedc, “Evaluation of spalling in bridges using machine vision method,” in *Proceedings of the 37th International Symposium on Automation and Robotics in Construction (ISARC 2020)*, Kitakyushu, Japan, 2020.
- [10] N.-D. Hoang, “Image processing-based spall object detection using gabor filter, texture analysis, and adaptive moment estimation (Adam) optimized logistic regression models,” *Advances in Civil Engineering*, vol. 2020, Article ID 8829715, 16 pages, 2020b.
- [11] E. Abdelkader, O. Moselhi, M. Marzouk, and T. Zayed, “Entropy-based automated method for detection and assessment of spalling severities in reinforced concrete bridges,” *Journal of Performance of Constructed Facilities*, vol. 35, no. 1, Article ID 04020132, 2021.
- [12] Z. Zhao, T. Liu, and X. Zhao, “Variable selection from image texture feature for automatic classification of concrete surface voids,” *Computational Intelligence and Neuroscience*, vol. 2021, pp. 1–10, 2021.
- [13] J. K. Chow, Z. Su, J. Wu, P. S. Tan, X. Mao, and Y. H. Wang, “Anomaly detection of defects on concrete structures with the convolutional autoencoder,” *Advanced Engineering Informatics*, vol. 45, Article ID 101105, 2020.
- [14] B. Kim and S. Cho, “Automated multiple concrete damage detection using instance segmentation deep learning model,” *Applied Sciences*, vol. 10, no. 22, p. 8008, 2020.
- [15] C. Zhang, C. c. Chang, and M. Jamshidi, “Concrete bridge surface damage detection using a single-stage detector,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 35, no. 4, pp. 389–409, 2019.
- [16] F. Wei, G. Yao, Y. Yang, and Y. Sun, “Instance-level recognition and quantification for concrete surface bughole based on deep learning,” *Automation in Construction*, vol. 107, Article ID 102920, 2019.
- [17] G. Yao, F. Wei, Y. Yang, and Y. Sun, “Deep-learning-based bughole detection for concrete surface image,” *Advances in Civil Engineering*, vol. 2019, Article ID 8582963, 12 pages, 2019.
- [18] H. Perez, J. H. M. Tah, and A. Mosavi, “Deep learning for detecting building defects using convolutional neural networks,” *Sensors*, vol. 19, no. 16, p. 3556, 2019.
- [19] N.-D. Hoang and Q.-L. Nguyen, “A novel approach for automatic detection of concrete surface voids using image texture analysis and history-based adaptive differential evolution optimized support vector machine,” *Advances in Civil Engineering*, vol. 2020, Article ID 4190682, 15 pages, 2020.
- [20] 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.
- [21] D. Feng and M. Q. Feng, “Computer vision for SHM of civil infrastructure: from dynamic response measurement to damage detection - a review,” *Engineering Structures*, vol. 156, pp. 105–117, 2018.
- [22] C.-Z. Dong and F. N. Catbas, “A review of computer vision-based structural health monitoring at local and global levels,” *Structural Health Monitoring*, vol. 20, no. 2, pp. 692–743, 2020.
- [23] R. Yadhunath, S. Srikanth, A. Sudheer, C. Jyotsna, and J. Amudha, “Detecting surface cracks on buildings using computer vision: an experimental comparison of digital image processing and deep learning,” in *Proceedings of the Soft Computing and Signal Processing*, pp. 197–210, Springer Singapore, Hyderabad, India, July 2021.
- [24] P. Asadi, M. Gindy, and M. Alvarez, “A machine learning based approach for automatic rebar detection and quantification of deterioration in concrete bridge deck ground penetrating radar B-scan images,” *KSCE Journal of Civil Engineering*, vol. 23, no. 6, pp. 2618–2627, 2019.
- [25] 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.
- [26] T.-T. Le, V.-H. Nguyen, and M. V. Le, “Development of deep learning model for the recognition of cracks on concrete surfaces,” *Applied Computational Intelligence and Soft Computing*, vol. 2021, Article ID 8858545, 10 pages, 2021.
- [27] M. H. Ahmadi, M. A. Ahmadi, M. A. Nazari, O. Mahian, and R. Ghasempour, “A proposed model to predict thermal conductivity ratio of Al₂O₃/EG nanofluid by applying least squares support vector machine (LSSVM) and genetic algorithm as a connectionist approach,” *Journal of Thermal Analysis and Calorimetry*, vol. 135, no. 1, pp. 271–281, 2019.
- [28] J.-S. Chou and N.-M. Nguyen, “Metaheuristics-optimized ensemble system for predicting mechanical strength of reinforced concrete materials,” *Structural Control and Health Monitoring*, vol. 28, no. 5, p. e2706, 2021.
- [29] S. Fan, S. Cao, and Y. Zhang, “Temperature prediction of photovoltaic panels based on support vector machine with pigeon-inspired optimization,” *Complexity*, vol. 2020, Article ID 9278162, 12 pages, 2020.
- [30] N.-D. Hoang, T.-C. Huynh, and V.-D. Tran, “Computer vision-based patched and unpatched pothole classification using machine learning approach optimized by forensic-based investigation metaheuristic,” *Complexity*, vol. 2021, Article ID 3511375, 17 pages, 2021.
- [31] E. Li, J. Zhou, X. Shi et al., “Developing a hybrid model of salp swarm algorithm-based support vector machine to predict the strength of fiber-reinforced cemented paste backfill,” *Engineering with Computers*, vol. 37, no. 4, pp. 3519–3540, 2020.
- [32] Y.-k. Pan, K.-w. Xia, W.-j. Niu, and Z.-p. He, “Semi-supervised SVM by hybrid whale optimization algorithm and its application in oil layer recognition,” *Mathematical Problems in Engineering*, vol. 2021, Article ID 5289038, 19 pages, 2021.
- [33] P. Parsa and H. Naderpour, “Shear strength estimation of reinforced concrete walls using support vector regression

- improved by Teaching-learning-based optimization, Particle Swarm optimization, and Harris Hawks Optimization algorithms,” *Journal of Building Engineering*, vol. 44, Article ID 102593, 2021.
- [34] V. N. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, New Jersey USA, 1998.
- [35] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, “A comprehensive survey on support vector machine classification: applications, challenges and trends,” *Neurocomputing*, vol. 408, pp. 189–215, 2020.
- [36] A. Çevik, A. E. Kurtoğlu, M. Bilgehan, M. E. Gülşan, and H. M. Albegmprli, “Support vector machines in structural engineering: a review,” *Journal of Civil Engineering and Management*, vol. 21, no. 3, pp. 261–281, 2015.
- [37] R. Y. Goh and L. S. Lee, “Credit scoring: a review on support vector machines and metaheuristic approaches,” *Advances in Operations Research*, vol. 2019, Article ID 1974794, 30 pages, 2019.
- [38] R. Gupta, M. A. Alam, and P. Agarwal, “Modified support vector machine for detecting stress level using EEG signals,” *Computational Intelligence and Neuroscience*, vol. 2020, Article ID 8860841, 14 pages, 2020.
- [39] N. Naicker, T. Adeliyi, and J. Wing, “Linear support vector machines for prediction of student performance in school-based education,” *Mathematical Problems in Engineering*, vol. 2020, Article ID 4761468, 7 pages, 2020.
- [40] J. Zhou, Y. Qiu, S. Zhu et al., “Optimization of support vector machine through the use of metaheuristic algorithms in forecasting TBM advance rate,” *Engineering Applications of Artificial Intelligence*, vol. 97, Article ID 104015, 2021.
- [41] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, Academic Press, Printed in the United States of America, 2009, https://en.wikipedia.org/wiki/Cambridge,_Massachusetts.
- [42] M. M. Galloway, “Texture analysis using gray level run lengths,” *Computer Graphics and Image Processing*, vol. 4, no. 2, pp. 172–179, 1975.
- [43] T. Xiaoou Tang, “Texture information in run-length matrices,” *IEEE Transactions on Image Processing*, vol. 7, no. 11, pp. 1602–1609, 1998.
- [44] M. Heikkilä, M. Pietikäinen, and C. Schmid, “Description of interest regions with local binary patterns,” *Pattern Recognition*, vol. 42, no. 3, pp. 425–436, 2009.
- [45] D. Barnat-Hunek, Z. Omiotek, M. Szafraniec, and R. Dzierżak, “An integrated texture analysis and machine learning approach for durability assessment of lightweight cement composites with hydrophobic coatings modified by nanocellulose,” *Measurement*, vol. 179, Article ID 109538, 2021.
- [46] Y. Cai, G. Xu, A. Li, and X. Wang, “A novel improved local binary pattern and its application to the fault diagnosis of diesel engine,” *Shock and Vibration*, vol. 2020, Article ID 9830162, 15 pages, 2020.
- [47] N.-D. Hoang and V.-D. Tran, “Image processing-based detection of pipe corrosion using texture analysis and metaheuristic-optimized machine learning approach,” *Computational Intelligence and Neuroscience*, vol. 2019, Article ID 8097213, 13 pages, 2019.
- [48] A. Humeau-Heurtier, “Texture feature extraction methods: a survey,” *IEEE Access*, vol. 7, pp. 8975–9000, 2019.
- [49] N. Kayhan and S. Fekri-Ershad, “Content based image retrieval based on weighted fusion of texture and color features derived from modified local binary patterns and local neighborhood difference patterns,” *Multimedia Tools and Applications*, vol. 80, no. 21–23, pp. 32763–32790, 2021.
- [50] L. Liu, P. Fieguth, Y. Guo, X. Wang, and M. Pietikäinen, “Local binary features for texture classification: t,” vol. 62, pp. 135–160, 2017.
- [51] N.-D. Hoang, “Image processing-based pitting corrosion detection using metaheuristic optimized multilevel image thresholding and machine-learning approaches,” *Mathematical Problems in Engineering*, vol. 2020, Article ID 6765274, 19 pages, 2020a.
- [52] N. Tadi Bani and S. Fekri-Ershad, “Content-based image retrieval based on combination of texture and colour information extracted in spatial and frequency domains,” *The Electronic Library*, vol. 37, no. 4, pp. 650–666, 2019.
- [53] S. Dash and M. R. Senapati, “Gray level run length matrix based on various illumination normalization techniques for texture classification,” *Evolutionary Intelligence*, vol. 14, no. 2, pp. 217–226, 2018.
- [54] 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.
- [55] 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.
- [56] T. Ojala and M. Pietikäinen, “Unsupervised texture segmentation using feature distributions,” *Pattern Recognition*, vol. 32, no. 3, pp. 477–486, 1999.
- [57] T. Ojala, M. Pietikäinen, and D. Harwood, “A comparative study of texture measures with classification based on featured distributions,” *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, 1996.
- [58] Y. El merabet, Y. Ruichek, and A. El idrissi, “Attractive-and-repulsive center-symmetric local binary patterns for texture classification,” *Engineering Applications of Artificial Intelligence*, vol. 78, pp. 158–172, 2019.
- [59] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer, /Heidelberg, Germany, 2011, https://www.google.com/search?rlz=1C1GCEB_enIN980IN980&q=Berlin&stick=H4sIAAAIAAAAAAONgVuLQz9U3MDQ1K1zEyuaUWpSTmQcAUzZQwhUAAAA&sa=X&ved=2ahUKewj2io2ZvKLOAhVJzDgGHfm7DWsQmxMoAXoECQMaw April 6, 2011).
- [60] M. S. Nixon and A. S. Aguado, *Feature Extraction & Image Processing for Computer Vision*, Academic Press, Elsevier, Cambridge, Massachusetts, 2012.
- [61] M. Heikkilä and M. Pietikäinen, “A texture-based method for modeling the background and detecting moving objects,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 657–662, 2006.
- [62] M.-Y. Cheng and N.-D. Hoang, “Typhoon-induced slope collapse assessment using a novel bee colony optimized support vector classifier,” *Natural Hazards*, vol. 78, no. 3, pp. 1961–1978, 2015.
- [63] L. H. Hamel, *Knowledge Discovery with Support Vector Machines*, John Wiley & Sons, New Jersey USA, 2009.
- [64] J.-S. Chou and T. T. H. Truong, “Sliding-window metaheuristic optimization-based forecast system for foreign exchange analysis,” *Soft Computing*, vol. 23, no. 10, pp. 3545–3561, 2019.
- [65] T. V. Dinh, H. Nguyen, X.-L. Tran, and N.-D. Hoang, “Predicting rainfall-induced soil erosion based on a hybridization of adaptive differential evolution and support vector machine classification,” *Mathematical Problems in Engineering*, vol. 2021, Article ID 6647829, 20 pages, 2021.
- [66] H. Luo and S. G. Paal, “Metaheuristic least squares support vector machine-based lateral strength modelling of

- reinforced concrete columns subjected to earthquake loads,” *Structure*, vol. 33, pp. 748–758, 2021.
- [67] D. Prayogo, M.-Y. Cheng, Y.-W. Wu, and D.-H. Tran, “Combining machine learning models via adaptive ensemble weighting for prediction of shear capacity of reinforced-concrete deep beams,” *Engineering with Computers*, vol. 36, pp. 1135–1153, 2019.
- [68] C. Qi and X. Tang, “Slope stability prediction using integrated metaheuristic and machine learning approaches: a comparative study,” *Computers & Industrial Engineering*, vol. 118, pp. 112–122, 2018.
- [69] D. Segerer, M. Mbuthia, and A. Nyete, “Particle swarm optimized hybrid kernel-based multiclass support vector machine for microarray cancer data analysis,” *BioMed Research International*, vol. 2019, Article ID 4085725, 11 pages, 2019.
- [70] D. Tien Bui, H. Shahabi, A. Shirzadi et al., “A novel integrated approach of relevance vector machine optimized by imperialist competitive algorithm for spatial modeling of shallow landslides,” *Remote Sensing*, vol. 10, no. 10, p. 1538, 2018.
- [71] Z.-Z. Zhu, Y.-W. Feng, C. Lu, and C.-W. Fei, “Reliability optimization of structural deformation with improved support vector regression model,” *Advances in Materials Science and Engineering*, vol. 2020, Article ID 3982450, 8 pages, 2020.
- [72] J.-S. Chou and D.-N. Truong, “A novel metaheuristic optimizer inspired by behavior of jellyfish in ocean,” *Applied Mathematics and Computation*, vol. 389, Article ID 125535, 2021b.
- [73] G. L. Mariottini and L. Pane, “Mediterranean jellyfish venoms: a review on scyphomedusae,” *Marine Drugs*, vol. 8, no. 4, pp. 1122–1152, 2010.
- [74] D. Zavodnik, “Spatial aggregations of the swarming jellyfish *Pelagia noctiluca* (Scyphozoa),” *Marine Biology*, vol. 94, no. 2, pp. 265–269, 1987.
- [75] MathWorks, *Statistics and Machine Learning Toolbox User’s Guide*, Mathworks Inc., Massachusetts USA, 2017, https://www.mathworks.com/help/pdf_doc/stats/stats.pdf Date of last access.
- [76] J.-S. Chou and D.-N. Truong, *Jellyfish Search Optimizer (JS) MATLAB Central*, <https://www.mathworks.com/matlabcentral/fileexchange/78961-jellyfish-search-optimizer-js>, 2021a.
- [77] Accord, “Accord.NET framework,” 2019, <http://accord-framework.net/> Last Access Date.
- [78] F. Chollet, *Deep Learning with Python*, Manning Publications, New York USA, 2015.
- [79] H. Tin Kam Ho, “The random subspace method for constructing decision forests,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 832–844, 1998.
- [80] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [81] P.-j. Chun, I. Ujike, K. Mishima, M. Kusumoto, and S. Okazaki, “Random forest-based evaluation technique for internal damage in reinforced concrete featuring multiple nondestructive testing results,” *Construction and Building Materials*, vol. 253, Article ID 119238, 2020.
- [82] 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.
- [83] T.-C. Huynh, “Vision-based autonomous bolt-looseness detection method for splice connections: design, lab-scale evaluation, and field application,” *Automation in Construction*, vol. 124, Article ID 103591, 2021.
- [84] M. Sharma, M. K. Sharma, N. Leeprechanon, W. Anotaipaboon, and K. Chaiyasarn, “Digital image patch based randomized crack detection in concrete structures,” in *2017 International Conference on Computer Communication and Informatics (ICCCI)*, vol. 5-7, pp. 1–7, Coimbatore, India, January 2017.
- [85] W. R. L. d. Silva and D. S. d. Lucena, “Concrete cracks detection based on deep learning image classification,” *Proceedings*, vol. 2, no. 8, p. 489, 2018.
- [86] S. Ullah, M. Jeong, and W. Lee, “Nondestructive inspection of reinforced concrete utility Poles with ISOMAP and random forest,” *Sensors*, vol. 18, no. 10, p. 3463, 2018.
- [87] H. Wan, L. Gao, M. Su, Q. Sun, and L. Huang, “Attention-based convolutional neural network for pavement crack detection,” *Advances in Materials Science and Engineering*, vol. 2021, Article ID 5520515, 13 pages, 2021.
- [88] W. Wang, W. Hu, W. Wang et al., “Automated crack severity level detection and classification for ballastless track slab using deep convolutional neural network,” *Automation in Construction*, vol. 124, Article ID 103484, 2021.
- [89] D. P. Kingma and J. Ba, “Adam: a method for stochastic optimization,” in *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, San Diego, January 2015.
- [90] T. Tieleman and G. Hinton, *Lecture 6.5 - RMSProp COURSERA: Neural Networks for Machine Learning*, Technical report, 2012.
- [91] MathWorks, *Deep Learning Toolbox the MathWorks, Inc.*, <https://www.mathworks.com/help/deeplearning/index> (Last Access Date), 2019.
- [92] V. López, A. Fernández, S. García, V. Palade, and F. Herrera, “An insight into classification with imbalanced data: empirical results and current trends on using data intrinsic characteristics,” *Information Sciences*, vol. 250, pp. 113–141, 2013.
- [93] L. Yu, C. Wang, H. Chang, S. Shen, F. Hou, and Y. Li, “Application research of intelligent classification Technology in enterprise data classification and gradation system,” *Complexity*, vol. 2020, Article ID 6695484, 9 pages, 2020.
- [94] A. R. van Erkel and P. M. T. Pattynama, “Receiver operating characteristic (ROC) analysis: basic principles and applications in radiology,” *European Journal of Radiology*, vol. 27, no. 2, pp. 88–94, 1998.
- [95] B. T. Pham, D. Tien Bui, and I. Prakash, “Bagging based Support Vector Machines for spatial prediction of landslides,” *Environmental Earth Sciences*, vol. 77, no. 4, p. 146, 2018.
- [96] M. Hollander and D. A. Wolfe, *Nonparametric Statistical Methods*, John Wiley & Sons, Hoboken, NJ, 1999.
- [97] W. J. Conover, *Practical Nonparametric Statistics*, John Wiley & Sons, Hoboken, NJ, 1999.
- [98] J. Chen, V. M. Patel, L. Liu et al., “Robust local features for remote face recognition,” *Image and Vision Computing*, vol. 64, pp. 34–46, 2017.
- [99] C.-X. Ren, Z. Lei, D.-Q. Dai, and S. Z. Li, “Enhanced local gradient order features and discriminant analysis for face recognition,” *IEEE Transactions on Cybernetics*, vol. 46, no. 11, pp. 2656–2669, 2016.
- [100] B. Everitt, *The Cambridge Dictionary of Statistics*, Cambridge University Press, Cambridge, UK New York, 1998.