IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 12, 2021, accepted November 25, 2021, date of publication November 29, 2021,
date of current version December 6, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3131231

Evolving Pre-Trained CNN Using Two-Layers Optimizer for Road Damage Detection From Drone Images

HUSSEIN SAMMA[1,2], SHAHREL AZMIN SUANDI[3], (Senior Member, IEEE), NOR AZMAN ISMAIL[2], SARINA SULAIMAN[2], AND LEE LI PING[3]

[1]Department of Computer Programming, Faculty of Education–Shabwa, University of Aden, Aden, Yemen
[2]Faculty of Engineering, School of Computing, Universiti Teknologi Malaysia (UTM), Johor Bahru, Johor 81310, Malaysia
[3]Intelligent Biometric Group, School of Electrical and Electronic Engineering, Engineering Campus, Universiti Sains Malaysia, Nibong Tebal, Penang 14300, Malaysia

Corresponding author: Hussein Samma (hussein.samma@utm.my)

This work was supported by the Ministry of Education Malaysia Fundamental Research Grant Scheme (FRGS) under Grant 203.PELECT.6071424.

ABSTRACT There are numerous pre-trained Convolutional Neural Networks (CNN) introduced in the literature, such as AlexNet, VGG-19, and ResNet. These pre-trained CNN models could be reused and applied to tackle different image recognition problems. Unfortunately, these pre-trained CNN models are complex and have a large number of convolutional filters. To tackle such a complexity challenge, this research aims to evolve a pre-trained VGG-19 using an efficient two-layers optimizer. The proposed optimizer performs filters selection of the last layers of VGG-19 guided by the accuracy of the linear SVM classifier. The proposed approach has three main advantages. Firstly, it adopts a powerful two-layers optimizer that works with a micro swarm population. Secondly, it automatically evolves a lightweight deep model which uses a small number of VGG-19 convolutional filters. Thirdly, It applies the developed model for real-world road damage detection from drone-based images. To evaluate the effectiveness of the proposed approach, a total of 529 images were captured by using a drone-based camera for various road damages. Reported results indicated that the proposed model achieved 96.4% F1-score accuracy with a reduction of VGG-19 filter up to 52%. In addition, the proposed two-layers optimizer was able to outperform several related optimizers such as Arithmetic Optimization Algorithm (AOA), Wild Geese Algorithm (WGO), Particle Swarm Optimization (PSO), Comprehensive Learning Particle Swarm Optimization (CLPSO), and Reinforcement Learning-based Memetic Particle Swarm Optimization (RLMPSO).

INDEX TERMS Pre-trained CNN, convolutional neural networks, deep learning, VGG-19, road damage, two-layers optimizer.

I. INTRODUCTION

Road maintenance is essential for keeping the roads in good condition. Such a good condition will provide safe and secure environments for road users. However, manual road inspections by a human is the most common method used for evaluating and detecting road damages [1]. The main challenge of this method is related to the high personal safety risks, especially on highways.

Unmanned Aerial Vehicles (UAV), commonly known as the drone, is a promising solution to automate the roads inspection process. This is due to the higher mobility of the drone, where it can move from one place to another easily; therefore, it can be used to monitor road conditions in larger areas. In the long run, it will bring better road maintenance at lower costs. The essential operation of the drone in performing roads inspection is illustrated in Fig. 1. As can be seen, a flying camera carried by a drone will record road conditions in real-time. Then, these recorded videos will be converted to images that will be presented to a classifier to classify them as normal or abnormal (see Fig. 2).

Recently, deep learning showed great success in recognizing road damages [2], [3]. For instance, Dung and Anh [3] employed a pre-trained CNN model (VGG-16) as the

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Jiang.

FIGURE 1. The basic process of road inspection by a drone.

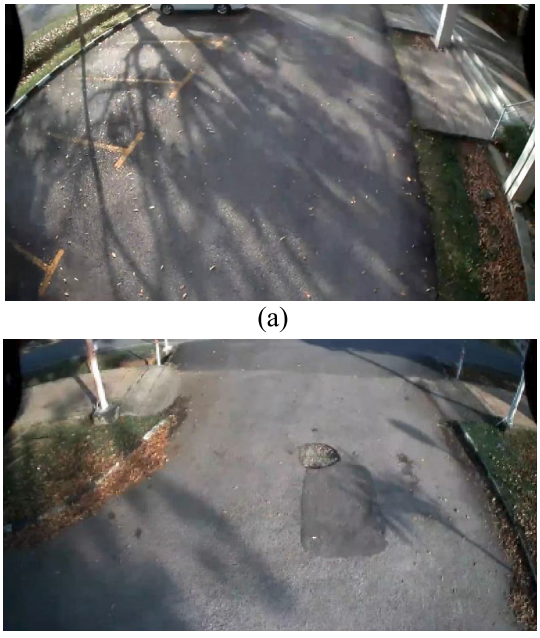

FIGURE 2. Samples of drone images (a) normal road view and (b) abnormal road view (damage).

backbone network for feature extraction with a softmax classifier. Reported results showed that road cracks detection accuracy reached 97.8% on a dataset of concrete crack images. Nevertheless, models in [2] and [3] were applied for non-drone captured images. More importantly, previous studies did not consider model complexity raised by the number of CNN filters that exist in the employed pre-trained networks such as VGG-19. To fill this gap and tackle such a complexity challenge, this study aims to evolve a pre-trained VGG-19 by a two-layers optimizer. The optimizer aims to find the most distinguished VGG-19 filters guided by two measures which are the classification accuracy of the linear SVM classifier and VGG-19 model complexity. The complexity measure is represented by the percentage of the remaining VGG-19 filters. It should be noted that our proposed two-layers optimizer was successfully applied for large-scale benchmark problems in a conference paper published recently [4]. However, in this study, it will be used for evolving the pre-trained VGG-19 model to tackle the problem of road damage detection. The main contribution of this work could be summarized in the following points:

- It adopts an efficient two-layers optimizer that has a global and local search layer and works with the micro swarm population.
- It evolves a lightweight pre-trained VGG-19 feature extractor that has a small number of convolutional filters.
- It applies the optimized lightweight model for real-world road damage detection from drone-based images.
- It compares the outcomes of the proposed two-layers optimizer with other well-known and recent algorithms.

The remaining part of this paper is organized as follows. Section II discusses related work. The details of the proposed approach are explained in Section III. A series of conducted experiments used to evaluate the effectiveness of the proposed approach are shown in Section IV, followed by the conclusions, limitations, and future direction given in Section V. Table 1 lists all abbreviations used in this study.

TABLE 1. List of abbreviations.

CNN	Convolutional Neural Networks
UAV	Unmanned Aerial Vehicles
ROC	Receiver Operating Characteristic
AOA	Arithmetic Optimization Algorithm
WGA	Wild Geese Algorithm
PSO	Particle Swarm Optimization
CLPSO	Comprehensive Learning Particle Swarm Optimization
RLMPSO	Reinforcement Learning-based Memetic Particle Swarm Optimization

II. RELATED WORK

In the literature, many deep learning-based techniques have been used for the problem of road damage detection. Zhang *et al.* [2] proposed a deep ResNet model for concrete crack detection. Reported results showed that the ResNet model achieved an accuracy of 96.84%. An additional end-to-end trainable CNN model named DeepCrack was given by Zou *et al.* [5]. The main aim was to combine the convolutional feature maps at different scales. Therefore, they introduced a multiscale feature-fusion map for road cracks detection. The outcome indicates that DeepCrack achieved F1-score accuracy of over 87%. However, the model was sensitive to noisy cracks. A CNN-based scheme which was evaluated with 500 smartphones images of size 3264 $\times$ 2448, was given by Zhang *et al.* [6]. Another study that introduced a CNN-based scheme with an adaptive thresholding technique

to detect road cracks was presented by Fan *et al.* [7]. In [7], a CNN was used to recognize cracked images, followed by an enhancement technique that was applied to reduce the noisy pixels and emphasize the edges of the cracks. Finally, the filtered images were down-sampled and thresholded to segment road cracks. The outcome indicates that the proposed scheme in [7] reported detection accuracy of 99.92% and pixel-level segmentation accuracy of 98.7%. A modified deep AlexNet model was introduced by Fan *et al.* [8]. Their model was applied to images with various situations of road cracks such as shadowed crack, blurry crack, etc. In their work, a sliding window of 156 x 156 pixels was applied and fed to the modified AlexNet to detect road cracks. Their reported performances showed that an average accuracy of 99.09% had been achieved. Yusof *et al.* [9] presented a fully automated crack detection using CNN, which was applied for two sets of the dataset. The first one consists of 9000 images with two classes namely crack, and non-crack. The second dataset consists of 5700 images with four categories, including non-crack, transverse, longitudinal, and alligator cracks. Their research showed an accuracy of 98%, 99%, and 99% in the measures of recall, precision, and accuracy, respectively.

A very recent deep learning model was investigated by Arya *et al.* [10]. Their approach performed a transfer learning of the pre-trained MobileNet with an SDD object detector to localize road damage. Their model was evaluated using a large-scale dataset for images captured by a smartphone camera installed in a car. The dataset has been collected from different countries, namely India, Japan, and the Czech Republic. The conducted experiment showed low performance in terms of the F1-score measure. A lightweight autoencoder model was given by Shim *et al.* [11]. The key idea of their model is that they classify the input image on pixel-level as normal or abnormal (damage). Their model was evaluated with 1700 images, where 400 images were used for testing, and the rest were used for model training. Results showed that F1-score detection accuracy reached 79.33%. A hierarchal semantic segmentation scheme for road damage inspection was studied by Wei *et al.* [12]. The proposed hierarchal system used a deep U-Net model that consists of four cascaded steps. These steps are road mark detection, tacking, region segmentation, and road damage identification. Their proposed hierarchal model was evaluated on images captured from a 15km segment of highway road in Beijing. Conducted analysis indicated that an average accuracy of 0.945 IoU measure had been achieved. Ali *et al.* devised a real-time deep Faster RCNN model for bridge steel cracks detection [13]. The proposed Faster RCNN was able to achieve a 93.31% precision rate. A novel two-stage deep learning model for simultaneous road crack detection and segmentation was given by Nguyen *et al.* [14]. Basically, the developed two-stage model adopted CNN for performing the detection and segmenting at the pixel level. In particular, the first stage is responsible for locating and detecting damage from a square input image of size 96×96. In case the scanned square is recognized as damage, then it is passed to the second stage for identifying damaged pixels. Reported results on various benchmark datasets showed that the two-stage deep model achieved F1-score accuracy of 91%. Further recent work was studied in [15], where a pre-trained deep CNN model was employed. Their pre-trained CNN was used to predict both road-side weather and surface conditions. Particularly, their model was trained to distinguish three different weather conditions which are clear, light snow, and heavy snow. In addition, three different surface categories, including dry, snowy, and wet. During their experiment, several pre-trained CNN models were investigated, and the results confirm the superiority of the ResNet model against other models. However, the ResNet model is considered a complex network, and it has a large number of filters that consume more computational time. The idea of performing bridge damage assessment using the deep autoencoder model was studied by Sarwar and Cantero [16]. They adopted LSTM with a deep autoencoder to recognize damage information encoded by a 1D input signal. Road's pothole detection based on different deep learning models, including LSTM and CNN, was presented by Varona *et al.* [17]. Results showed that CNN was able to distinguish potholes from non-pothole with an accuracy of 98%. An additional recent deep learning-based scheme was conducted by Li *et al.* [18] for the problem of automatic defect detection of metro tunnel surfaces. Specifically, the Faster RCNN model was employed to localize three different types of defects, namely crack, falling block, and leakage. Faster R-CNN demonstrated the best performances as compared with other deep models such as YOLOv3, SSD, and R-FNN. Similarly, deep Faster R-CNN has been applied for concrete crack detection [19]. They investigated the stability of the deep models against the change in illumination level as well as weather conditions. Results showed a high effect in the accuracy of crack detection rate when the darkness increased. CNN for pavement distress detection was studied by Zhang *et al.* [20]. They adopt VGG-19 with different architecture. Their model has been tested to classify input images of size 150 x 150 pixels into six classes, namely clean, patch, pothole, linear crack, network crack, and pavement marking. The best results were achieved by a small VGG-19 with a recognition accuracy of 83.8%. A lightweight deep model that adopts an autoencoder network was utilized for performing semantic segmentation of road damage [21]. The key idea of their approach is to reduce model complexity by eliminating the decoding stage of the autoencoder network. Besides the achieved good accuracy of the proposed lightweight autoencoder network, their model reduced the computational time by 12.4%.

An IoT-based sensor for capturing images for the road was given in [22]. The captured images are passed to a deep CNN model for performing pixel-level image segmentation. In their approach, CNN was trained using a bio-inspired optimizer, and results showed 99% per-pixel segmentation accuracy. A pyramid-based deep architecture has been devised by Wang and Su [23] for the problem of road pavement damage. The proposed pyramid model was evaluated with a

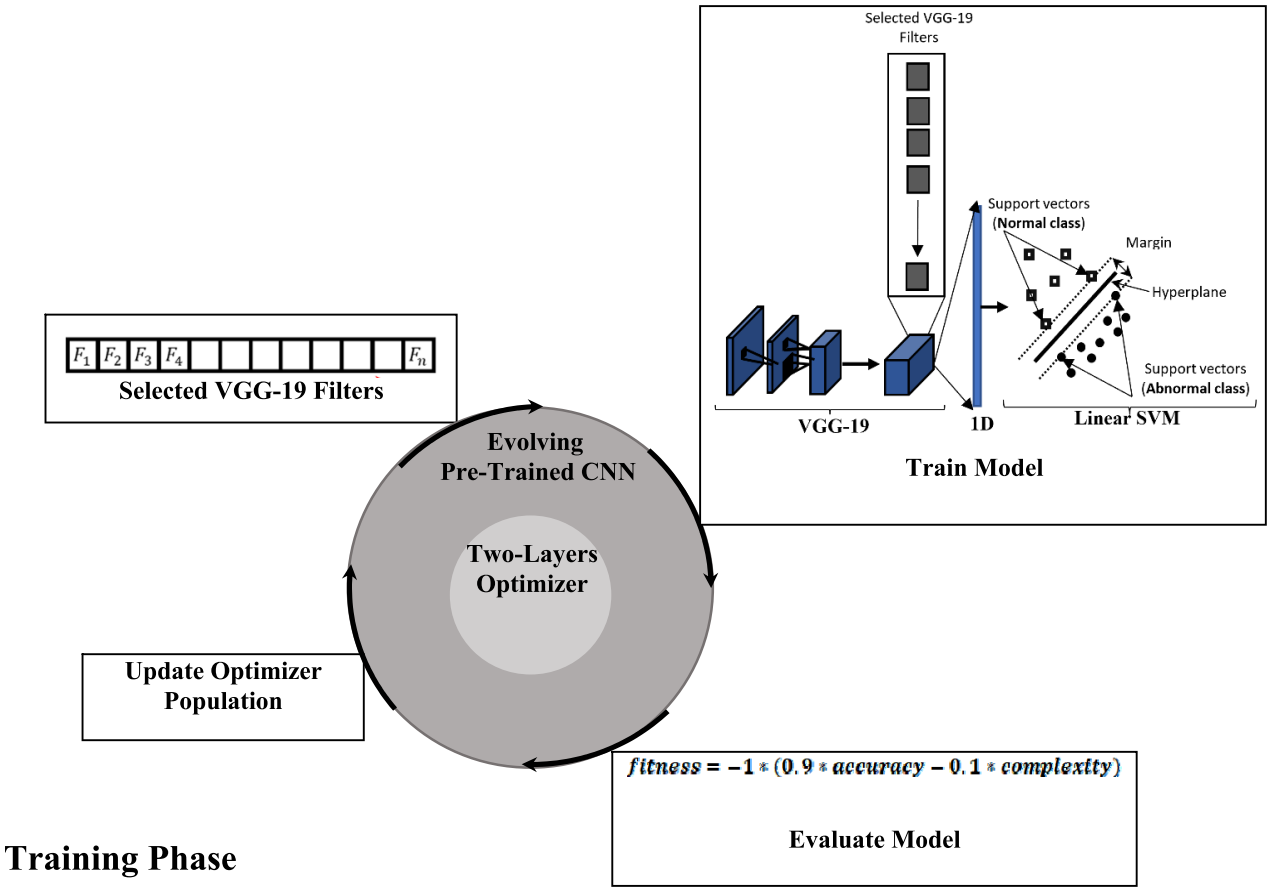

FIGURE 3. Evolving Pre-Trained VGG-19 using two-layers optimizer.

dataset of 500 images, and results indicated a segmentation accuracy of 0.6235 in terms of IoU measure. The concept of transfer learning of pre-trained deep models for damage detection was illustrated by Feng *et al.* [24]. Specifically, Inception-v3 has been transferred and finely tuned to recognize five different types of road cracks. The results indicated that the performance of the transferred deep model significantly outperformed other traditional machine learning models.

III. EVOLVING PRE-TRAINED CNN USING TWO-LAYERS OPTIMIZER

The architecture of the evolved pre-trained model is given in Fig. 3. As can be seen, the proposed architecture contains two main phases, namely the training phase and the testing phase. In the training phase, an efficient two-layers optimizer is used to perform VGG-19 filters selection guided by the performances of a linear SVM classifier. However, in the testing phase, the final optimized VGG-19 with linear SVM

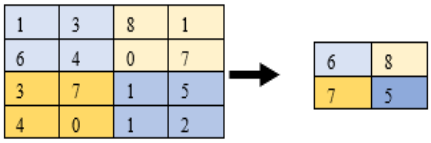

FIGURE 4. VGG-19 max-pooling operation.

is applied on the testing set to measure model performances. The main components of the proposed approach are explained as follows.

A. VGG-19 NETWORK

VGG-19 is a pre-trained network that was developed by a research team from Oxford University [25]. VGG-19 was trained with the ImageNet challenge dataset that contains millions of images. It was trained to classify a total of one thousand different objects such as cars, balloons, strawberries, etc. The internal architecture of VGG-19 is shown in Fig. 5 (a). Basically, it consists of a cascaded 25 layers. The input layer was designed to receive an input image of size 224 × 224 pixels. Then, the input image is passed to three cascaded convolutional layers, each with 64 filters of the size of 3 × 3 pixels. After each convolutional operation, a ReLU activation function is applied to improve the nonlinearity of VGG-19, and it is defined as follows.

$$relu(x) = \max(x, 0) \tag{1}$$

where x is the pixel value; after that, the output of *relu(x)* function is passed to the max-pooling layer as given in Fig. 5(a). The main idea of the max-pooling operation is to reduce the size of the feature map produced by *relu(x)*. Fig. 4 demonstrates the basic operation of max-pooling operation where it takes the maximum value from the 2 × 2 window.

In VGG-19, the convolutional, ReLU, and max-pooling operations are repeated several times until they reach the flattening layer, which produces a 1D vector, as shown in Fig. 4. Basically, the flattening layer is the concatenation of the output of max-pooling operation to be in the form of a 1D vector. It should be noted that when going deeper into VGG-19, the number of convolutional filters is increased, which increases the complexity of the model. Therefore, this research aims to reduce and eliminate less contributed filters that exist in the last layers of VGG-19.

B. TWO-LAYERS OPTIMIZER

The proposed two-layers optimizer was presented in our previous research as a conference paper [4]. Basically, it consists of two-layers which are global search and local search, as given in Fig. 6. It is automatically selected between local and global search based on generated action by the Q-learning algorithm, as explained in [4]. The main stages of two-layers optimizer are explained as follows.

1) INITIALIZATION

This stage is responsible for the initialization of the micro population (3 particles) of the two-stage optimizer. Each

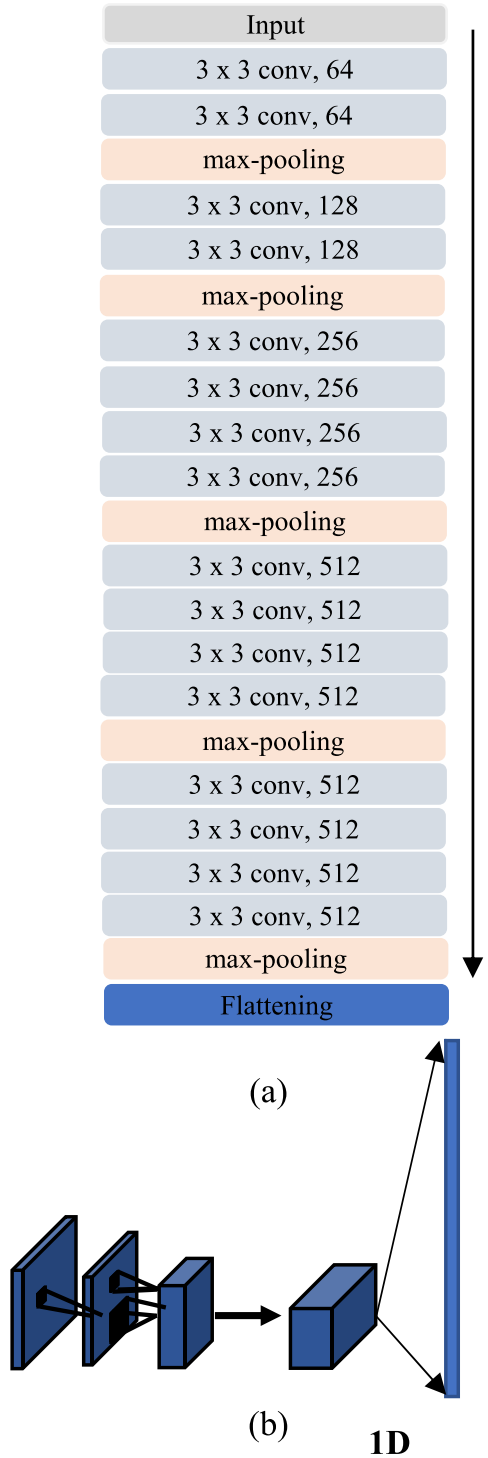

FIGURE 5. VGG-19 network (a)VGG-19 internal architecture, and (b) VGG-19 cascaded layers.

particle X is given a random initial location according to the search space of the conducted problem. In addition, a velocity value V is randomly initialized for each particle, and its value belongs to the same range as X.

2) TRANSITION

In this stage, a Q-learning algorithm is embedded to control the switching from global search to local search and vice

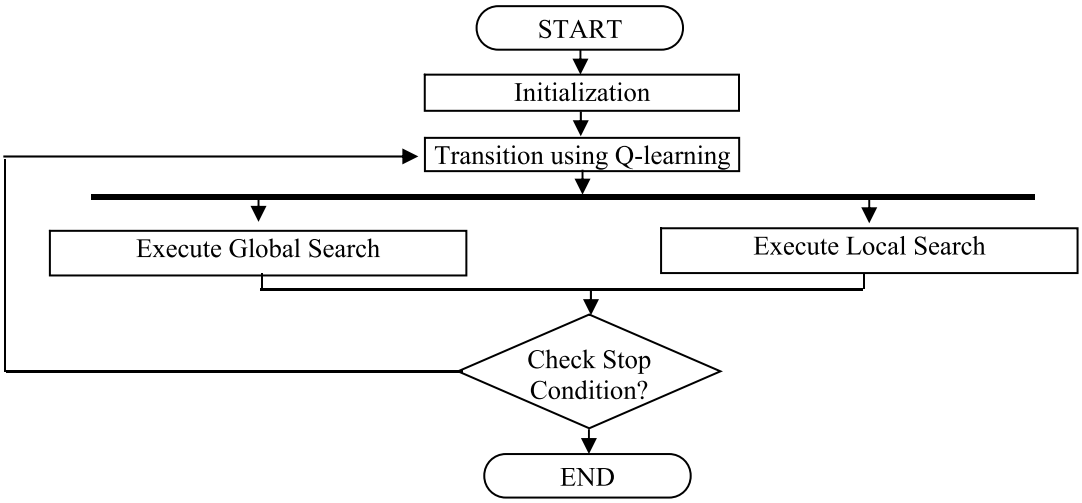

FIGURE 6. Two-layers optimizer.

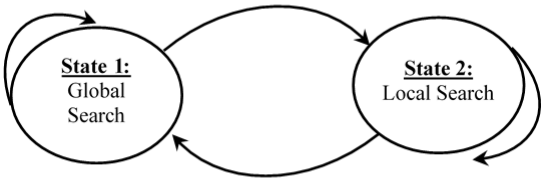

FIGURE 7. State diagram transition of Q-learning.

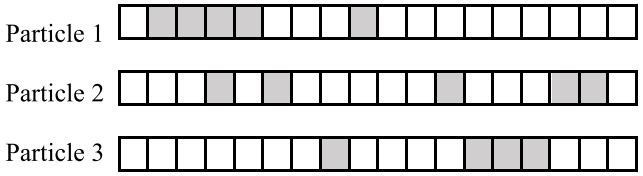

FIGURE 8. Local search mode.

versa. As such, the Q-learning is modeled as a two-state diagram given in Fig. 7.

As mentioned in [4] that the Q-table will be updated with a reward of +1 when the executed search operations were able to improve search performances; otherwise, a penalty of -1 is given.

3) SEARCH EXECUTION

The micro population of the two-stage optimizer will be updated and evolved based on the following equations.

$$X_i^{t+1} = X_i^t + V_i^{t+1} \tag{2}$$

$$V_i^{t+1} = w^* V_i^t + c_1{}^* r_1(pBest_i - X_i^t) + c_2{}^* r_2(gBest_i - X_i^t) \tag{3}$$

where X_i^{t+1} is the new location of currently executed particle i, V_i is particle velocity, ω is the inertia. Parameters c_1 and c_2 are cognitive and social acceleration coefficients, respectively. Variables r_1 and r_2 are random numbers in the range (0,1). $pBest_i$ is the local best position achieved, and $gBest_i$ is the global best position achieved by the micro swarm. Global and local search operations are identical; however, when the optimizer performs a local search, randomly selected bins of the evolved particle will be updated, other bins will remain unchanged, as indicated in Fig. 8.

4) STOP CONDITION

This stage is responsible for checking the maximum number of allocated iterations. In case it is met, then the execution of the two-stage optimizer is stopped, and the best potion $gBest_i$ achieved by the micro population is returned.

C. MODEL TRAINING STEPS

As can be seen in Fig. 5(a) that the last eight layers of VGG-19 consume the highest computational cost because they contain 512 filters each. As such, the main goal of this research is to focus on the last eight layers of VGG-19 and eliminate less contributed filters. To encode VGG-19 filter selection as an optimization problem, this work proposed a binary encoding scheme. The length of this binary vector is 512 × 8, which resulted in a 4096 search vector length. As such, each bin of the encoding scheme is associated with variable F that could take a value of zero or one. If the bin value is one, then the corresponding filter is selected, and it will be activated during the features extraction process. Otherwise, it will be skipped. Referring to the steps of the model training phase shown in Fig. 3, after identifying the VGG-19 filter that will be activated during the features extraction stage, all training images are fed to the optimized VGG-19, and each input image will produce a 1D vector that represents the computed features. Then, a linear SVM classifier will be trained based on computed 1D vectors for each class, i.e., normal and abnormal. In the next step (Evaluate Model), the fitness function will be computed for each particle of the proposed two-layers optimizer according to the following formula.

$$fitness = -1^*(0.9^* accuracy - 0.1^* complexity) \tag{4}$$

$$Accuracy = \frac{TP + TN}{total\_images} \tag{5}$$

TABLE 2. Collected images by the drone.

	Training set	Testing set
Normal	252	64
Abnormal	170	43

where TP is the total number of images correctly classified as road damage, TN is the total number of images correctly classified as non-road damage (Normal). The *complexity* variable is the ratio of the total selected VGG-19 filters with respect to the total number of filters (i.e., 512 x 8 layers).

Therefore, the proposed two-layers optimizer will perform iterative VGG-19 filters selection guided by the fitness function given in Equation (4). It should be noted that higher weightage (i.e., 0.9) is given to accuracy over complexity due to the importance of accuracy. Once the executed two-layers optimizer meets the maximum number of iterations, the best-achieved solution will be returned, and the optimization process will be terminated. After that, the testing phase will be executed based on the optimal VGG-19 filters, and it will be applied to the testing set.

IV. EXPERIMENTAL RESULTS

A. DATASET

In this study, a total of 529 images were captured by an S-Series S30W drone that has a wide-angle camera shown in Fig. 1. These images were taken for both normal roads with 422 images and abnormal (damaged roads) with 107 images. A number of sample images of different road damages are given in Fig. 9.

For evaluation purposes, the collected images were divided randomly into 80% for training and 20% for the testing phase, as indicated in Table 2. The training set contains 252 normal images and 170 abnormal images. However, the testing set has 64 normal images and 43 abnormal images.

B. PERFORMANCE MEASURES

The standard evaluation measures were employed in this study to determine the efficiency of the proposed approach. These measures are precision, recall, and F1-score. The mathematical formula of these measures is defined as follows.

$$precision = \frac{TP}{TP + FP} \tag{6}$$

$$recall = \frac{TP}{TP + FN} \tag{7}$$

$$F1 - score = 2 \times \frac{precision \times recall}{precision + recall} \tag{8}$$

where TP is the total number of images correctly classified as road damage, TN is the total number of images correctly classified as non-road damage (Normal), FP is the total number of images wrongly classified as road damage, but they are normal images, and FN is the total number of images wrongly classified as normal, but they are road damages (Abnormal).

FIGURE 9. Sample images for road damage.

TABLE 3. Performance results on the testing set.

	Accuracy %	Recall %	Precision %	F1-score %
VGG-19 with linear SVM	96.3	90.7	100	95.1
The proposed approach	**97.2**	**93.0**	**100**	**96.4**

In addition, the Receiver Operating Characteristic (ROC) curve is computed to evaluate the performances of the proposed approach graphically.

C. MODEL EVALUATION

The proposed approach is evaluated on the collected images according to the split of the data are given in Table 2. In this experiment, the employed two-layers optimizer was executed ten times on the training data. Specifically, it is used to evolve the pre-trained VGG-19 guided by the fitness function defined in Equation (1), as explained in Section III. For each run, it was executed for 1000 iterations, and the final optimized VGG-19 is evaluated on the testing dataset, as shown in Fig. 3. Furthermore, the proposed approach was compared with the outcomes of VGG-19 (non-optimized) with linear SVM, and the results are detailed in Table 3. It is clearly seen that the proposed approach yields the best performances in all measures, i.e., accuracy, precision, recall, and F1-score. This is due to the benefit of the incorporated two-layers optimizer, which performs simultaneous VGG-19 complexity reductions as well as accuracy enhancement of the linear SVM on the training set (see Equation (1)). In other words, these improvements are related to the reduction in complexity of VGG-19, which resulted in better generalization in linear SVM.

To assess the performances of the proposed approach graphically, the ROC plot of the proposed approach and VGG-19 with linear SVM is given in Fig. 10. From the

TABLE 4. Total filters of the VGG-19.

	Optimized VGG-19	VGG-19
Total number of filters	2650	5504

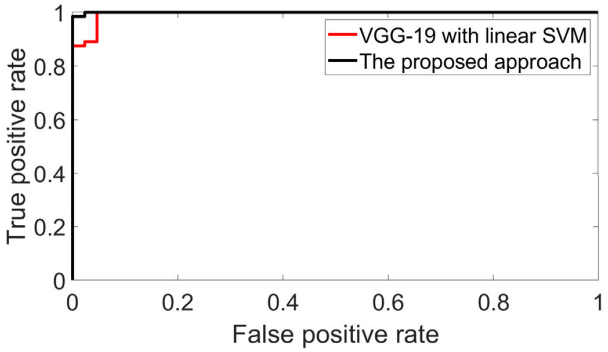

FIGURE 10. ROC curve analysis.

curves, it can be seen that the proposed approach has the largest area under the curve. This confirms the usefulness of the embedded two-layers optimizer in improving model performance.

Additional analysis was performed to analyze model complexity in terms of the total number of VGG-19 filters. Table 4 presents the total number of filters used in the proposed approach against the number of filters used in VGG-19. It can be seen that the non-optimized VGG-19 model has a total of 5504 convolutional filters; meanwhile, the optimized VGG-19 has an average of 2650 filters. This reduction in VGG-19 filters is due to the effectiveness of the incorporated two-layer optimizer.

Nevertheless, there are still some challenging road damage cases where the proposed approach failed to classify. These cases are given in Fig. 11. These cases are abnormal, but they have been classified as normal. This misclassification occurs due to the presence of background objects such as cars, trees, etc. In addition, road damage is not clearly visible, which makes it difficult for VGG-19 to capture road texture features.

Further analyses were conducted by measuring the effect of changing the brightness levels of the input image. This analysis has been applied to the test set without retraining the proposed model. Specifically, two types of illumination change were investigated, namely increasing brightness level by 1.5 and decreasing it by 0.5. For illustration purposes, a sample image with its corresponding illumination variation is shown in Fig. 12.

The experimental analysis is given in Table 5. It is clearly seen that the recall rate has been decreased to 81.4% due to the reduction of the brightness level, which makes the image to be darker, as shown in Fig. 12 (c). This reduction in the brightness level affects texture and shape features that contribute to the detection of road damage. On the other hand, increasing the brightness level by 1.5 did not affect model accuracy. This is owing to texture details of road damage becoming clearly visible and easily encoded by the VGG-19 feature extractor.

FIGURE 11. Misclassified abnormal cases.

TABLE 5. Analysis of brightness level change.

	Accuracy %	Recall %	Precision %	F1-score %
Increasing brightness level (by 1.5)	97.2	93.0	100	96.4
Decreasing brightness level (by 0.5)	92.5	81.4	100	89.7

D. COMPARE WITH THE STANDARD VGG-19

This section compares the performances of the proposed approach against the standard VGG-19 with a softmax classifier. As such, VGG-19 has been retrained with the same training set and evaluated using the same test set. The outcome of this experiment is given in Table 6. It is clearly seen that the proposed approach outperforms VGG-19 with a softmax classifier. This is due to the benefit of the generalization ability of linear SVM classifier. More importantly, SVM can be trained with a small training dataset; however, VGG-19 with softmax classifier required a large number of training instances to produce better recognition performances.

E. COMPUTATIONAL TIME ANALYSIS

The computational time of the proposed optimized approach is compared with the non-optimized VGG-19 that works will all filters (5504 filters). This experiment is executed with a PC that has MATLAB 2021a software and windows 10. The hardware specifications are 32 GB memory with i7-8700 CPU @ 3.2 GHz. The computational time analysis is given in Table 7. It is clearly shown that the proposed approach is able

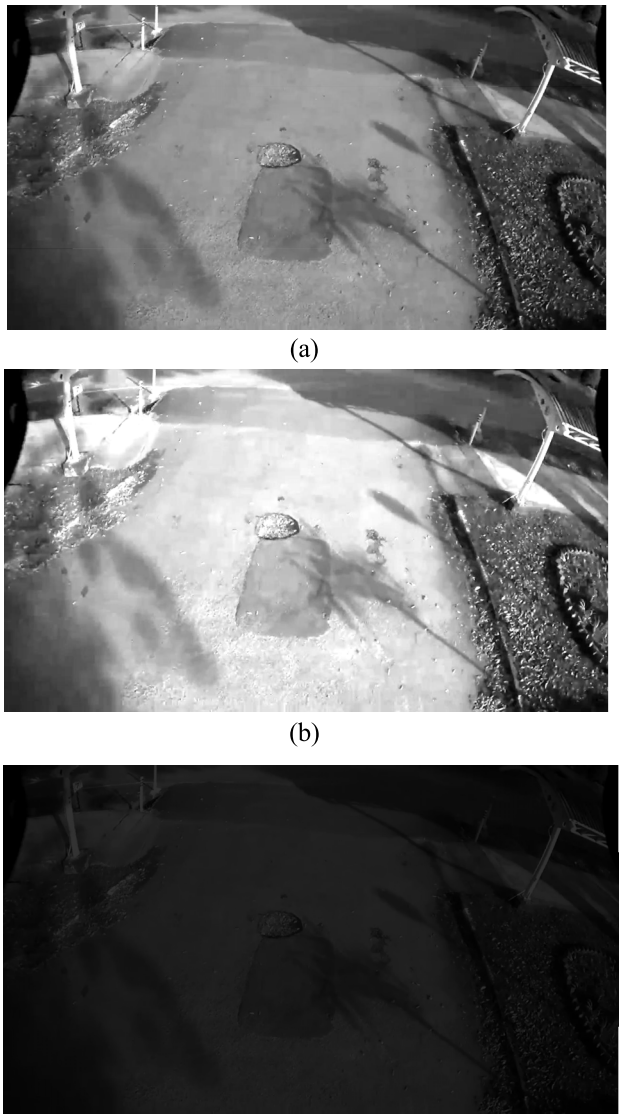

FIGURE 12. Effect of brightness level on test images, (a) original image in grayscale format, (b) after increasing the brightness by 1.5, and (c) after decreasing the brightness by 0.5.

to reduce the computational time to 57%, and it requires only 0.066 seconds per single image. On the other hand, the non-optimized VGG-19 takes a longer time, with 0.115 seconds per single image.

F. COMPARE WITH OTHER OPTIMIZERS

This section aims to compare the outcomes of the proposed two-layers optimizer against other well-known related PSO-based variants such as PSO [26], RLMPSO [27], and CLPSO [28]. In addition, two recent optimization algorithms have been included to validate the performance of the proposed two-layers optimizer. Specifically, AOA [29] and WGA [30] were used in this analysis.

This conducted experiment compares the performances of ten runs produced by each algorithm. Table 7 reports the mean fitness, total number of filters, and F1-score measure for each

TABLE 6. Comparison with the standard VGG-19.

	Accuracy %	Recall %	Precision %	F1-score %
VGG-19 (with softmax classifier)	93.4	83.7	100	91.5
The proposed approach	**97.2**	**93.0**	**100**	**96.4**

TABLE 7. Computational time analysis.

	VGG-19 with linear SVM	The proposed approach
Time (sec)/per image	0.115	**0.066**

TABLE 8. Comparison with other optimizers.

	Fitness	Total number of filters	F1-score %
AOA [29]	-0.8374	3723	95.4
WGA [30]	-0.8398	3698	95.4
RLMPSO [27]	-0.7913	4012	95.4
CLPSO [28]	-0.8120	3673	96.3
PSO [26]	-0.8216	3916	95.4
Two-Layers [4]	**-0.8520**	**2650**	**96.4**

algorithm. It is shown in Table 8 that the proposed two-layers optimizer is able to report the best mean fitness value with -0.8520. This is due to the benefit of dynamic transition from exploration to exploitation by the Q-learning algorithm. In addition, PSO, AOA, WGA, and CLPSO were executed with a large population size of 30 particles. On the other hand, both RLMPSO and the two-layers optimizer have been evolved with a micro population of 3 particles only. Despite that, RLMPSO reported the worst results owing to the high cost of fine-tuning operations consumed by its local search algorithm [27]. In terms of the total number of VGG-19 filters produced by each optimizer, it can be seen that the two-layers optimizer was able to produce the lowest number of filters with 2650. In addition, the reported mean F1-score value of all algorithms is somehow similar, as indicated in Table 7. This is because of the ability of a linear SVM classifier to give a good result even with many VGG-19 filters. It is worth mentioning that using all VGG-19 filters, the linear SVM classifier achieved an accuracy of 95.4% for the F1-score measure, as indicated in Table 3.

G. STATISTICAL TEST ANALYSIS

To measure the superiority of the proposed approach against other methods, this section compares the outcomes statically. Specifically, the Wilcoxon rank-sum test [31] is used in this analysis. The null hypothesis of the Wilcoxon test H_0 assumes that the outcomes of all algorithms are drawn from the same distribution. The significance level was set to

TABLE 9. Wilcoxon rank-sum of the proposed approach against other models ($p < 0.05$).

	VGG-19 (with softmax classifier)	VGG-19 (with linear SVM)
p-value	0.0235	0.0348

TABLE 10. Wilcoxon rank-sum of the proposed two-layers optimizer against other optimizers ($p < 0.05$).

	AOA	WGA	RLMPSO	PSO	CLPSO
p-value	0.0396	0.0327	0.0009	0.0291	0.0012

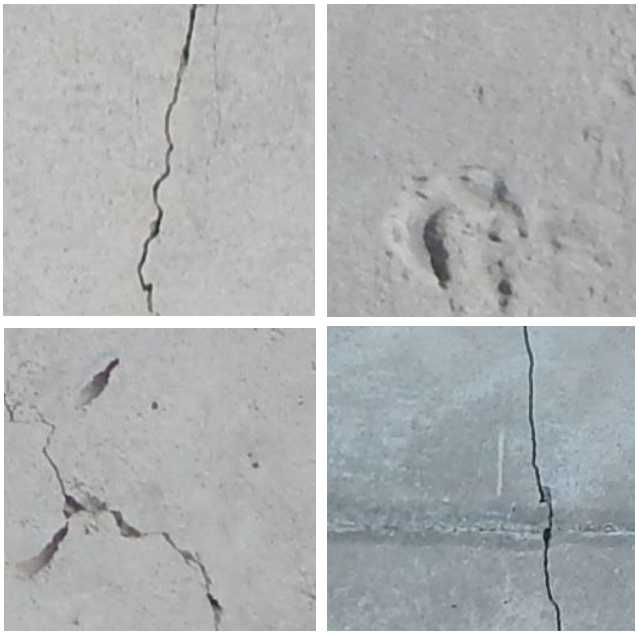

FIGURE 13. Sample of cracked images from SDNET2018 dataset [32].

0.05, which means that the alternative hypothesis H_1 would be accepted when the *p-value* was less than 0.005 (95% confidence level). The results of the Wilcoxon rank-sum test are given in Tables 9 and 10. It can be clearly seen that the proposed approach significantly outperforms both VGG-19 with softmax classifier and VGG-19 with linear SVM (non-optimized), as given in Table 9. However, Table 10 compares the results of the two-layers optimizer against other optimization algorithms, which are AOA, WGA, RLMPSO, PSO, and CLPSO. It can be clearly seen that all computed *p-values* are less than 0.005, which implies that the Wilcoxon test rejects the null hypothesis H_0 for all compared methods.

H. MODEL EVALUATION USING PUBLIC DATASET

To validate the performance of the proposed approach, an additional benchmark dataset has been used in this section. Unfortunately, there is no public dataset for road damage detection from drone images. Therefore, the most related work is concrete crack detection using SDNET2018 public dataset [32]. SDNET2018 contains more than 56k images of cracked and non-cracked images. Some sample images from SDNET2018 are shown in Fig. 13.

TABLE 11. Performance results on the testing set.

	Accuracy %
The proposed approach	**89.52**
AlexNet [32]	**92.25**
GoogleNet [33]	**88.39**

TABLE 12. Comparison with other optimizers.

	Fitness	Total number of filters	Accuracy %
AOA [29]	-0.6967	2987	89.18
WGA [30]	-0.6982	2962	89.13
RLMPSO [27]	-0.6813	3915	89.11
CLPSO [28]	-0.6897	3673	89.31
PSO [26]	-0.6930	2949	89.32
Two-Layers [4]	**-0.7013**	**2917**	**89.52**

In this experiment, the SDNET2018 dataset has been divided into 70% to 30% for training, testing respectively, as in **[33]**. The outcomes of the proposed approach and other reported results in the literature are given in Table 11. The results indicated that the proposed method is able to achieve similar results to some extent. Nevertheless, the proposed approach has many advantages as compared with **[33]** and **[32]**. First, it combines the benefits of the deep learning models (i.e., VGG-19) in performing automatic feature extraction and the strength of the linear SVM classifier in performing classification. Secondly, it is automatic evolving and does not require retraining of deep models (i.e., VGG-19). Specifically, the proposed approach performs simultaneous complexity reduction of VGG-19 and SVM classification performance enhancement guided by the formulated fitness function in Equation (4).

Further analysis has been conducted by comparing the outcomes of the two-layers optimizer with other optimizers on SDNET2018 dataset. The results of this analysis are given in Table 12, and it can be clearly seen that all optimizers achieved almost the same test accuracy. One possible reason is related to the challenge of this problem, and none of the optimizer was able to improve test accuracy performance. Nevertheless, the proposed approach was able to produce the best fitness value and the minimum number VGG-19 filters with an average of 2917 filters.

V. CONCLUSION, LIMITATION AND FUTURE DIRECTIONS

This study proposed a novel pre-trained CNN model which has been evolved with a two-layers optimizer. The presented the approach was employed to classify drone-based captured road images as normal or abnormal (damaged). The proposed approach has been evaluated with a total of 422 normal road images and 107 abnormal images. Reported results indicated that the proposed approach is effective in road damage detection, and it was able to achieve an accuracy of 96.4% in

the F1-score measure. In terms of model complexity, the proposed approach reduced the VGG-19 filter by up to 52%. Finally, the comparative experiment on the employed two-layers optimizer confirms its ability to outperform other well-known optimizers such as AOA, WGA, RLMPSO, PSO, and CLPSO in the VGG-19 filters selection.

Nevertheless, the proposed approach suffers from several limitations, such as the stability against the change in image illumination. This is due to the lack of sufficient training images. In addition, localizing road damage inside the recognized images was not investigated in this study.

As future work, it is suggested that the proposed model could be evaluated using a large dataset with heterogeneous road damage types such as crack, falling block, and leakage. Moreover, the proposed approach could be applied for several computer vision problems such as COVID-19 classification and plant diseases recognition.

REFERENCES

[1] G. R. Rada, *Guide for Conducting Forensic Investigations of Highway Pavements (With Supplemental Material on CD-ROM)*, vol. 747. Columbia, SC, USA: Transportation Research Board, 2013.

[2] J. Zhang, C. Lu, J. Wang, L. Wang, and X.-G. Yue, "Concrete cracks detection based on FCN with dilated convolution," *Appl. Sci.*, vol. 9, no. 13, p. 2686, Jul. 2019.

[3] C. V. Dung and L. D. Anh, "Autonomous concrete crack detection using deep fully convolutional neural network," *Autom. Construct.*, vol. 99, pp. 52–58, Mar. 2019, doi: 10.1016/j.autcon.2018.11.028.

[4] H. Samma, S. A. Suandi, and J. Mohamad-Saleh, "Two-layers particle swarm optimizer," in *Proc. IEEE Int. Conf. Autom. Control Intell. Syst. (I2CACIS)*, Jun. 2020, pp. 165–169.

[5] Q. Zou, Z. Zhang, Q. Li, X. Qi, Q. Wang, and S. Wang, "DeepCrack: Learning hierarchical convolutional features for crack detection," *IEEE Trans. Image Process.*, vol. 28, no. 3, pp. 1498–1512, Mar. 2019.

[6] L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 3708–3712.

[7] R. Fan, M. J. Bocus, Y. Zhu, J. Jiao, L. Wang, F. Ma, S. Cheng, and M. Liu, "Road crack detection using deep convolutional neural network and adaptive thresholding," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 474–479.

[8] S. Li and X. Zhao, "Image-based concrete crack detection using convolutional neural network and exhaustive search technique," *Adv. Civil Eng.*, vol. 2019, pp. 1–12, Apr. 2019.

[9] N. A. M. Yusof, A. Ibrahim, M. H. M. Noor, N. M. Tahir, N. M. Yusof, N. Z. Abidin, and M. K. Osman, "Deep convolution neural network for crack detection on asphalt pavement," in *J. Phys., Conf. Ser.*, vol. 1349, no. 1, 2019, Art. no. 12020.

[10] D. Arya, H. Maeda, S. K. Ghosh, D. Toshniwal, A. Mraz, T. Kashiyama, and Y. Sekimoto, "Deep learning-based road damage detection and classification for multiple countries," *Autom. Construct.*, vol. 132, Dec. 2021, Art. no. 103935.

[11] S. Shim, J. Kim, S.-W. Lee, and G.-C. Cho, "Road surface damage detection based on hierarchical architecture using lightweight auto-encoder network," *Autom. Construct.*, vol. 130, Oct. 2021, Art. no. 103833.

[12] C. Wei, S. Li, K. Wu, Z. Zhang, and Y. Wang, "Damage inspection for road markings based on images with hierarchical semantic segmentation strategy and dynamic homography estimation," *Autom. Construct.*, vol. 131, Nov. 2021, Art. no. 103876.

[13] R. Ali, D. Kang, G. Suh, and Y.-J. Cha, "Real-time multiple damage mapping using autonomous UAV and deep faster region-based neural networks for GPS-denied structures," *Autom. Construct.*, vol. 130, Oct. 2021, Art. no. 103831.

[14] N. H. T. Nguyen, S. Perry, D. Bone, H. T. Le, and T. T. Nguyen, "Two-stage convolutional neural network for road crack detection and segmentation," *Expert Syst. Appl.*, vol. 186, Dec. 2021, Art. no. 115718.

[15] M. N. Khan and M. M. Ahmed, "Weather and surface condition detection based on road-side webcams: Application of pre-trained convolutional neural network," *Int. J. Transp. Sci. Technol.*, Jun. 2021, doi: 10.1016/j.ijtst.2021.06.003.

[16] M. Z. Sarwar and D. Cantero, "Deep autoencoder architecture for bridge damage assessment using responses from several vehicles," *Eng. Struct.*, vol. 246, Nov. 2021, Art. no. 113064.

[17] J. Kim, H. Kim, M. Shim, and E. Choi, "CNN-based network intrusion detection against denial-of-service attacks," *Electronics*, vol. 9, no. 6, p. 916, Jun. 2020.

[18] D. Li, Q. Xie, X. Gong, Z. Yu, J. Xu, Y. Sun, and J. Wang, "Automatic defect detection of metro tunnel surfaces using a vision-based inspection system," *Adv. Eng. Informat.*, vol. 47, Jan. 2021, Art. no. 101206.

[19] K. Hacıefendioğlu and H. B. Başağa, "Concrete road crack detection using deep learning-based faster R-CNN method," *Iranian J. Sci. Technol., Trans. Civil Eng.*, pp. 1–13, Jun. 2021, doi: 10.1007/s40996-021-00671-2.

[20] C. Zhang, E. Nateghinia, L. F. Miranda-Moreno, and L. Sun, "Pavement distress detection using convolutional neural network (CNN): A case study in Montreal, Canada," *Int. J. Transp. Sci. Technol.*, Apr. 2021, doi: 10.1016/j.ijtst.2021.04.008.

[21] S. Shim and G.-C. Cho, "Lightweight semantic segmentation for road-surface damage recognition based on multiscale learning," *IEEE Access*, vol. 8, pp. 102680–102690, 2020.

[22] O. Alfarraj, "Internet of Things with bio-inspired co-evolutionary deep-convolution neural-network approach for detecting road cracks in smart transportation," *Neural Comput. Appl.*, pp. 1–16, Oct. 2020, doi: 10.1007/s00521-020-05401-9.

[23] W. Wang and C. Su, "Convolutional neural network-based pavement crack segmentation using pyramid attention network," *IEEE Access*, vol. 8, pp. 206548–206558, 2020.

[24] C. Feng, H. Zhang, S. Wang, Y. Li, H. Wang, and F. Yan, "Structural damage detection using deep convolutional neural network and transfer learning," *KSCE J. Civil Eng.*, vol. 23, no. 10, pp. 4493–4502, Oct. 2019.

[25] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[26] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. Int. Conf. Neural Netw. (ICNN)*, vol. 4, 1995, pp. 1942–1948.

[27] H. Samma, C. P. Lim, and J. M. Saleh, "A new reinforcement learning-based memetic particle swarm optimizer," *Appl. Soft Comput.*, vol. 43, pp. 276–297, Jun. 2016, doi: 10.1016/j.asoc.2016.01.006.

[28] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE Trans. Evol. Comput.*, vol. 10, no. 3, pp. 281–295, Jun. 2006.

[29] L. Abualigah, A. Diabat, S. Mirjalili, M. A. Elaziz, and A. H. Gandomi, "The arithmetic optimization algorithm," *Comput. Methods Appl. Mech. Eng.*, vol. 376, Apr. 2021, Art. no. 113609.

[30] M. Ghasemi, A. Rahimnejad, R. Hemmati, E. Akbari, and S. A. Gadsden, "Wild Geese algorithm: A novel algorithm for large scale optimization based on the natural life and death of wild geese," *Array*, vol. 11, Sep. 2021, Art. no. 100074.

[31] T. P. Hettmansperger and J. W. McKean, *Robust Nonparametric Statistical Methods*. Boca Raton, FL, USA: CRC Press, 2010.

[32] S. Dorafshan, R. J. Thomas, and M. Maguire, "SDNET2018: An annotated image dataset for non-contact concrete crack detection using deep convolutional neural networks," *Data Brief*, vol. 21, pp. 1664–1668, Dec. 2018.

[33] P. Miao and T. Srimahachota, "Cost-effective system for detection and quantification of concrete surface cracks by combination of convolutional neural network and image processing techniques," *Construct. Building Mater.*, vol. 293, Jul. 2021, Art. no. 123549.

HUSSEIN SAMMA received the B.Eng. degree in computer engineering from Yarmouk University, Jordan, in 2006, the M.Eng. degree in computer engineering from the Jordan University of Science and Technology, in 2009, and the Ph.D. degree in computer vision and machine learning from Universiti Sains Malaysia, in 2016. He is currently working as a Senior Lecturer with the School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia. He is also an Assistance Professor with the Department of Computer Programming, Faculty of Education–Shabwa, University of Aden, Aden, Yemen. His research interests include computer vision, deep learning, swarm intelligence, and soft biometrics. He is a member of the IEEE Computational Intelligence Society, and has served as a Reviewer for several well-known international journals, such as *Machine Vision and Applications*.

SHAHREL AZMIN SUANDI (Senior Member, IEEE) received the B.Eng. degree in electronic engineering and the M.Eng. and D.Eng. degrees in information science from the Kyushu Institute of Technology, Fukuoka, Japan, in 1995, 2003, and 2006, respectively. He joined the industries, such as Sony Video (M) Sdn. Bhd. and Technology Park Malaysia Corporation Sdn. Bhd., for a period of six years, as an Engineer. He is currently a Professor and the Deputy Dean of research, innovation and industry-community engagement with the School of Electrical and Electronic Engineering, Universiti Sains Malaysia, Malaysia. His current research interests include face-based biometrics, real-time object detection and tracking, and pattern classification using deep learning. He served as a Reviewer for several international conferences and journals, including *IET Biometrics*, *IET Computer Vision*, *Multimedia Tools and Applications*, *Neural Computing and Applications*, *Journal of Electronic Imaging*, the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, and IEEE ACCESS.

NOR AZMAN ISMAIL received the Ph.D. degree from Loughborough University, U.K. He has been a Lecturer with the School of Computing, Universiti Teknologi Malaysia (UTM), for more than 20 years. He has contributed to computer graphics and human–computer interaction (HCI), including research, practice, and education.

SARINA SULAIMAN received the Diploma and B.Sc. degrees in computer science from Universiti Teknologi Malaysia (UTM), Malaysia, the M.Sc. degree in science (multimedia system) from Universiti Putra Malaysia (UPM), Malaysia, and the Ph.D. degree in computer science on mobile web pre-caching using social networking from UTM. She has been granted as the Project Leader for 13 projects from the Ministry of Science, Technology and Innovation, Malaysia, the Ministry of Higher Education, Malaysia, and UTM. She also has been assigned as a researcher for 35 projects. She has published more than 90 research publications in journals, book chapters, and conference proceedings. Her research interests include soft computing, web caching and pre-fetching in mobile apps, web usage mining, human–computer interaction, and social networking and wireless communication.

LEE LI PING received the B.S. degree in mechatronics from the School of Electrical and Electronic Engineering, Universiti Sains Malaysia. Her main areas of research interests include artificial intelligence, machine vision, and image processing.