IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 10 October 2023, accepted 9 December 2023, date of publication 10 January 2024,
date of current version 17 January 2024.

Digital Object Identifier 10.1109/ACCESS.2023.3343619

RESEARCH ARTICLE

Improving the Classification Performance of Asphalt Cracks After Earthquake With a New Feature Selection Algorithm

MEHMET YILMAZ[1], ERKUT YALÇIN[1], SAIF KIFAH[2], FATIH DEMIR[3], ABDULKADIR ŞENGÜR[4], ROZERIN DEMIR[1], AND RAJA MAJID MEHMOOD[2], (Senior Member, IEEE)

[1]Civil Engineering Department, Engineering Faculty, Fırat University, 23119 Elâzığ, Turkey
[2]School of Computing and Data Science, Xiamen University Malaysia, Sepang 43900, Malaysia
[3]Software Engineering Department, Engineering Faculty, Fırat University, 23119 Elâzığ, Turkey
[4]Electric and Electronic Department, Fırat University, 23119 Elâzığ, Turkey

Corresponding author: Raja Majid Mehmood (rmeex07@ieee.org)

This work was supported in part by the Xiamen University Malaysia Research Fund (XMUMRF) under Grant XMUMRF/2022-C9/IECE/0035, and in part by the Tubitak 1002C Project under Grant 123D058.

ABSTRACT Large-scale earthquakes can cause huge loss of life and material losses. After an earthquake, highways are the most commonly used type of transportation for the delivery of the necessary aid teams and materials to the scene of the event. If the highways are not well maintained, it may cause serious disruption of transportation after the earthquake or aftershocks. In this study, field studies were conducted in the provinces where the earthquake was felt severely after the earthquakes in Turkey on February 6, 2023. In these studies, images were collected according to the condition of asphalt cracks on the highways. These images were labeled as in need of urgent maintenance (Major) and not in need of urgent maintenance (Minor) and a new dataset was created. The classification performance of popular pre-trained CNN models is evaluated on this dataset. First, classification algorithms other than softmax were used to improve the classification performance. The Combined Metaheuristic Optimization-Relieff (CMO-R) algorithm was designed to improve the classification performance by one more level. Extensive experiments were conducted on the dataset, and the VGG16 model demonstrated superior performance, reaching an accuracy of 80.32% without encountering overfitting.

INDEX TERMS Classification, deep learning, new asphalt cracks dataset, new feature selection algorithm.

I. INTRODUCTION

Two major earthquakes occurred in Turkey on February 6, 2023, in Pazarcık and Elbistan districts of Kahramanmaraş province. Large-scale loss of life and material damage occurred in 11 different provinces of Turkey. Immediately after the earthquakes, both human and logistical aid started to arrive from different provinces of Turkey and different countries. However, there were serious delays in this aid due to asphalt deformations on the highways. As a result of the field studies, it was seen that highways with previously deformed asphalt prevented road transportation. Moreover, in the event of the next earthquake or other disasters, asphalt cracks that could prevent transportation were identified by experts in the field, and images of these asphalt cracks were taken. However, the effects of the current earthquakes were so great that not all sites could be surveyed for the current condition of the highways, as access to certain earthquake-affected routes was closed. Therefore, pre-earthquake highway maintenance works are of great importance to prevent such transportation disruptions [1], [2]. However, a significant amount of specialized personnel is required to identify these maintenance works. Recently, artificial intelligence systems have started to give superior performances in automated decision support systems. Especially after 2012, with the development of deep learning models, great progress has been made in solving

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

© 2024 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

automatic classification, regression, and segmentation problems. These models have been applied in many fields such as medicine, engineering, economics, and law.

In the proposed approach, a new deep learning-based technique is used to determine the maintenance urgency of highways from asphalt crack images after the February 6 earthquakes in Turkey.

The important contributions of the proposed approach are as follows.

- A new dataset that provides asphalt crack status has been created with the field studies carried out after the earthquake.
- By testing 5 different transfer learning approaches and 6 different classifiers, a detailed analysis of the baseline classification performance is presented.
- Designed a new feature selection algorithm that combines the power of 10 different metaheuristic algorithms with index matching and incorporates the ReliefF algorithm.

II. LITERATURE REVIEW

The main advantage of deep learning techniques is the ability to extract meaningful features from images [3]. For this, convolutional filters with adjustable weights are used. However, in classical machine learning techniques, it is necessary to design a highly discriminative feature extraction algorithm.

The automatic classification of asphalt cracks represents a prevalent challenge within the realm of computer vision [4]. Numerous empirical investigations have underscored the escalating prominence and efficacy of Convolutional Neural Networks (CNNs) in the context of crack detection [5], [6], [7] In a study conducted by Gopalakrishnan et al. [8], a pre-trained VGG-16 CNN model was employed for the binary classification of pavement cracks into the categories of ''crack'' and ''no crack.'' Huyan et al. [9] introduced the CrackU-net model, which adapted the established U-net architecture to facilitate the pixel-level segmentation of cracks in a dataset comprising 3000 images. Mandal et al. [10] developed a pavement crack recognition system by leveraging the YOLO v2 deep learning framework, leveraging a dataset consisting of 9053 road photographs collected from a moving vehicle. The work of Majidifard et al. [11] proposed a comprehensive approach to asphalt crack categorization, involving the utilization of YOLO net for crack detection and U-Net for crack segmentation. Guan et al. [12] established an automated methodology for pixel-level classification of asphalt cracks, incorporating both deep learning and image processing techniques while taking into account both the color and depth attributes of the images.

Most existing studies focus on detecting asphalt cracks rather than grading their severity, highlighting the need for a methodology capable of discerning and classifying crack severity levels. Li et al. [13] pioneered a deep learning-based approach tailored to the automatic classification of diverse types of cracks observed on asphalt pavements, creating an extensive dataset encompassing five fracture categories, notably including fatigue cracks. Zhu et al. [14] employed unmanned aerial vehicles (UAVs) for the acquisition of pavement images and harnessed deep learning techniques to recognize six distinct distress types, encompassing four categories of cracks such as fatigue cracks. Liu et al. [15] advocated a two-tiered methodology for the identification and delineation of pavement cracks, predicated on the implementation of CNNs, with a specific focus on detecting four distinct crack types, prominently featuring fatigue cracks. For three distinct image types, Liu et al. [16] harnessed Grad CAM-based models to expound upon the interpretability of the CNN model's outputs, thereby facilitating the identification of diverse crack severity categories, including fatigue, longitudinal, and transverse cracks. Finally, Tran et al. [17] proffered a two-stage automated crack identification and severity classification methodology, entailing the deployment of image processing techniques to estimate the severity levels of linear fractures.

III. MATERIAL

Within the framework of this comprehensive study, the dataset under scrutiny was meticulously sourced from on-site investigations specifically undertaken to scrutinize the intricate asphalt deformations observed on highways subsequent to the Kahramanmaraş Pazarcık and Elbistan earthquakes, both registering magnitudes of 7.7 and 7.6. The geographical scope of these investigations encompassed regions profoundly impacted by the seismic activity, namely Kahramanmaraş, Hatay, Malatya, Adıyaman, and Gaziantep provinces.

The data acquisition process involved the use of a state-of-the-art professional Canon camera, capable of capturing images in JPEG format and boasting a remarkable resolution of 24 megapixels. The meticulous capture of a total of 528 images was carried out with a focus on the distinct asphalt deformations present in the aftermath of the seismic events. These images were strategically collected in varying quantities from each province, with 111 images obtained from Adıyaman province, 38 from Gaziantep province, 191 from Hatay province, 101 from Kahramanmaraş province, and 87 from Malatya province.

Asphalt cracks are labeled by experts in the field under two classes: minor and major cracks. During labeling, 10 photographs with distorted images were eliminated. Of the total 518 images, 225 were labeled as major and 293 as minor. Although the width and height of the images are not standardized, all pre-trained models except the AlexNet model (227×227) were resized to (224×224).

IV. PROPOSED METHODOLOGY

A representative illustration of the proposed method is given in Figure 2. In general terms, the proposed method consists of 4 main stages. In the first stage, an evaluation is performed on the dataset with a transfer learning approach including the VGG16 model. The reason for choosing the transfer learning approach is that the size of the dataset is not too large.

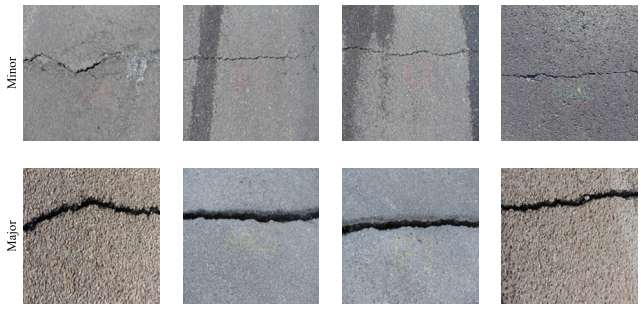


FIGURE 1. Sample images in the data set created as a result of field studies.

Therefore, a transfer learning strategy was chosen instead of a model that learns from scratch. In stage 2, deep features were extracted from the activation of the dataset with the network weighted by transfer learning. The deep features were extracted from the fully connected layers fc7 (4096) and fc8 (1000) of the VGG16 model. Then, the extracted features were combined to obtain a 5096-dimensional feature set. In stage 3, the Combined Metaheuristic Optimization-Relieff (CMO-R) feature selection algorithm was used both to reduce computational costs and to improve classification performance. In the CMO-R algorithm, 10 different metaheuristic optimization algorithms consisting primarily of Atom Search Optimization (ASO), Equilibrium Optimizer (EO), Generalized Normal Distribution Optimization (GNDO), Henry Gas Solubility Optimization (HGSO), Harris Hawks Optimization (HHO), Marine Predators Algorithm (MPA), Manta Ray Foraging Optimization (MRFO), Particle Swarm Optimization (PSO), Slime Mould Algorithm (SMA), and Whale Optimization Algorithm (WOA) techniques were used. These algorithms are preferred both because they provide fast output and their performance is high. This rule uses index values chosen by 10 metaheuristic optimization algorithms (MOAs). If at least 2 metaheuristic optimization algorithms had a common index, that index was included in the feature selection set. In other words, instead of combining the features of the 10 MOA algorithms, mostly matching indices were preferred. This made the feature set more effective. A total of 518 features were selected through the index-matching process. To further reduce the feature size and improve performance, the ReliefF algorithm was applied to the 518 feature set. The ReliefF algorithm is easy and effective. With the ReliefF algorithm, 518 features are weighted. A certain thresholding value was selected for the features with high-weight values. With this technique, the feature set size was finally reduced to 58. In the classification stage, 6 popular classifiers (Medium Neural Network (MNN), Support Vector Machine (SVM), Decision Tree (DT), K-nearest Neighbor (KNN) Linear Discriminant (LD), and Naive Bayes (NB)) in machine learning were preferred. The best performance was achieved with the MNN algorithm and this classifier was used in the proposed approach.

V. METHODOLOGY TECHNIQUES

A. TRANSFER LEARNING

The practicality of data acquisition in engineering applications is often impeded by logistical challenges and associated costs, resulting in limited datasets for deep learning endeavors. Paradoxically, the performance of deep learning models exhibits a pronounced sensitivity to dataset size. Addressing this quandary necessitates the exploration of potential solutions, with transfer learning emerging as a salient avenue. Transfer learning is a strategic framework concerned with the preservation and transfer of knowledge derived from solving one problem domain to the domain of another, albeit closely related, problem [18]. The core objective of transfer learning, within the framework of a source domain and learning task, along with a target domain and learning task, is the enhancement of the learning process governing the prediction function within the target domain through the application of insights garnered from the source domain and learning task [3]. Notably, within the context of deep neural networks, four principal modalities of transfer learning are discernible: instance-based transfer learning, mapping-based transfer learning, network-based transfer learning, and adversarial-based transfer learning [19].

The utility of transfer learning has been robustly substantiated within civil engineering, particularly in the domain of image analysis tasks, such as the identification of structural damage. Such applications are frequently fraught with challenges associated with either constrained data accessibility or substantial resource allocation demands for data collection. Consequently, datasets available for such applications are often characterized by their meager proportions. Paradoxically, contemporary convolutional neural network (CNN) models are distinguished by their architectural intricacy, which renders them reliant on extensive datasets to mitigate the risk of overfitting. Transfer learning, therefore, has arisen as an efficacious recourse to surmount these impediments within the purview of civil engineering.

In the context of transfer learning applied to CNNs for tasks such as the categorization of asphalt pavement crack severity, the operational paradigm typically encompasses two pivotal phases. In the initial phase, a CNN model is systematically trained to utilize an extensive image dataset emanating from the source domain, frequently leveraging resources akin to ImageNet [20], encompassing a compendium of 1.2 million images distributed across 1000 categories. This pre-trained CNN model, herein termed the ''pre-trained CNN model,'' is readily accessible through online repositories. Subsequently, in the second phase, the pre-trained CNN model is transposed into the target domain. While preserving the convolutional layer parameters (i.e., the pre-trained convolutional layers), requisite adaptations are effected within the fully connected layers (FC layers) to accommodate the specific output label requirements, which, in this instance, constitute a binary classification task. Subsequently, the adapted pre-trained CNN

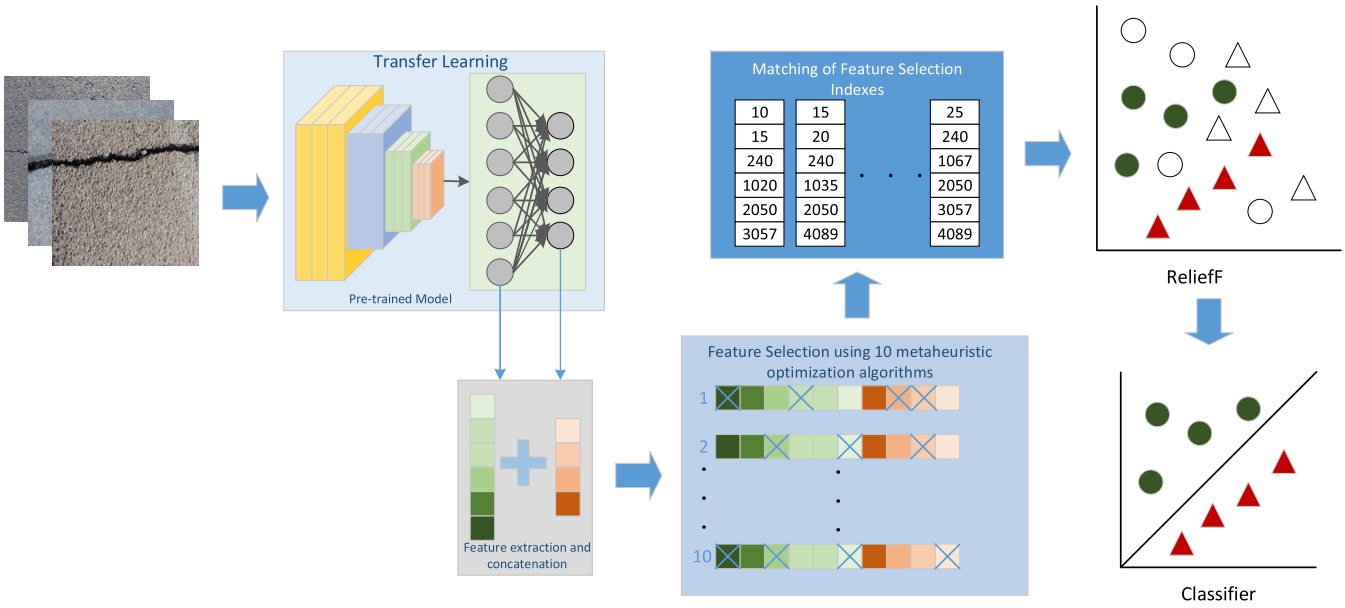


FIGURE 2. Representative illustration of the proposed approach.

model undergoes fine-tuning via exposure to the dataset specifically tailored to the intended application.

For the purposes of transfer learning in this context, the selection gravitated toward the VGG16 model, predicated on its notable proficiency in the classification of the dataset. VGG16 represents a convolutional neural network architecture incubated by the Visual Geometry Group (VGG) at the University of Oxford [20]. It stands as an eminent exemplar within the realm of deep learning and computer vision. Noteworthy for its user-friendliness and efficacy as a pre-trained model across diverse computer vision tasks, VGG16's architecture comprises a constellation of 16 layers, with 13 dedicated to convolutional operations and the remaining 3 configured as fully connected layers. Notably, its architectural design is characterized by the utilization of compact 3×3 convolutional filters, characterized by a unitary stride, alongside the incorporation of max-pooling layers adopting 2×2 windows and employing a stride of two. This uniform arrangement of convolutional and pooling layers confers the capacity to capture features across multiple scales.

B. METAHEURISTIC ALGORITHMS

Metaheuristic algorithms are optimization strategies for solving difficult optimization problems. Engineering, operations research, computer science, and other fields make extensive use of metaheuristics. They are especially beneficial for non-linear, non-convex, and multimodal optimization problems [21].

Natural processes such as evolution, swarming behavior, and annealing inspire metaheuristic algorithms, which use ideas from these processes to guide the search for optimal solutions. These algorithms do not ensure that the global optimum will be found, but they do strive to discover a suitable solution in a fair amount of time [22].

For feature selection in various machine learning and optimization problems, metaheuristic algorithms have multiple advantages. Metaheuristic methods are meant to search for solutions across the whole search space, to find optimal or near-optimal solutions. This global search capability qualifies them for feature selection, as they can identify pertinent features that local search methods may miss. Metaheuristics use randomness or probabilistic search algorithms to explore diverse regions of the search space in a non-deterministic way. This randomization can aid in escaping local optima and exploring diverse feature combinations, resulting in increased feature space exploration. Metaheuristics can be used for a wide range of optimization problems, including feature selection, without the need for considerable problem-specific changes. They can be tailored and modified to meet a variety of objectives and limitations in feature selection activities. Because many real-world datasets contain a large number of features, feature selection can be computationally difficult. Metaheuristic algorithms can frequently handle high-dimensional data by intelligently exploring feature space and identifying essential features while reducing computational overhead. Complex and non-convex objective functions, such as those found in machine learning models, may be used in feature selection. Metaheuristics are built to handle such complex functions by iteratively refining solutions and gradually enhancing the objective value. Metaheuristics find a balance between search space exploration and exploitation. They rotate between exploration (diversification), which aids in the discovery of new promising solutions, and exploitation (intensification), which refines solutions in the search space's promising areas. This balance

is especially relevant for feature selection, where it is critical to investigate various feature subsets while optimizing speed. Metaheuristics do not require explicit knowledge of the objective function's underlying mathematical properties or gradients. This property qualifies them for feature selection in situations where the objective function is complex or not easily differentiable. Many metaheuristic algorithms can be parallelized or distributed across numerous processing units, resulting in faster convergence and more efficient exploration of the feature space. Unlike certain optimization methods, which require derivatives of the objective function, metaheuristics can work without them. When dealing with non-differentiable or noisy objective functions, this is useful. Metaheuristics enable the investigation of trade-offs between competing aims, such as maximizing classification accuracy while lowering the number of selected characteristics.

In summary, metaheuristic algorithms offer a powerful and versatile method of feature selection, allowing for the successful exploration of complex and high-dimensional feature spaces to find optimal or near-optimal subsets of features for diverse machine-learning applications. ASO, EO, GNDO, HGSO, HHO, MPA, MRFO, PSO, SMA, and WOA algorithms are used in this study. These algorithms are used because they are new and popular in metaheuristics. Detailed information about these algorithms is given in [23], [24], [25], [26], [27], [28], [29], [30], [31], and [32].

C. RELIEFF ALGORITHM

The ReliefF algorithm is a method for selecting features in machine learning and data mining. It is especially beneficial for classification and regression tasks, where the goal is to choose the most important features (variables or attributes) from a given dataset to increase the accuracy and efficiency of a predictive model. The ReliefF is an improvement on the original Relief algorithm, which Kira and Rendell introduced in 1992. The ReliefF, which stands for "Relief Feature Selection," is a dataset processing tool that can handle both continuous and categorical variables [33], [34], [35], [36], [37]. The algorithm is mostly used in supervised learning for feature selection. The general steps of the ReliefF algorithm are as follows.

Step 1 (Initializations): The algorithm begins by selecting a data instance (sample) at random from the dataset.

Step 2 (Distance Computation): ReliefF computes the "near-hit" and "near-miss" examples from the same and different classes for the selected instance. It uses a distance metric to quantify the difference between attribute values for these instances.

Step 3 (Feature Relevance Update): Based on the differences computed in the previous stage, ReliefF adjusts the relevance score for each attribute. Higher relevance scores are assigned to features that contribute more to discriminating between instances of various classes.

Step 4 (Weight Update): ReliefF changes the weights of characteristics depending on the relevance scores after processing a sample. Weights are assigned to more relevant attributes.

Step 5 (Iteration): Steps 1–4 are repeated for a predetermined number of iterations or until convergence is reached.

Step 6 (Feature Selection): The final stage is to rank the qualities according to their relevance scores or weights. The top-ranked characteristics are used in the following machine-learning model.

The final stage is to rank the qualities according to their relevance scores or weights. The top-ranked characteristics are used in the following machine-learning model. The weight equation of the ReliefF feature selection algorithm is given in Eq. 1.

$$W(x^b) = W(x^b) - \frac{\sum_{j=1}^{k} diff\,(B, E_i, S_j)}{mxk} + \sum_{c \neq class(R_i)} \left[\frac{P\,(C)}{1 - P(class\,(E_i))} x \sum_{j=1}^{t} \frac{diff\,(B, E_{i,M_j})}{mxk} \right] \tag{1}$$

where *both* feature is symbolized by x^b, the feature set is symbolized by B, and the instances of the feature set are E_i and S_j and the hyperparameter is symbolized by t.

D. CMO-R ALGORITHM

The Combined Metaheuristic Optimization-ReliefF (CMO-R) feature selection algorithm is a new algorithm that combines the feature selection power of 10 different metaheuristic optimization algorithms and the ReliefF algorithm. In this algorithm, the features that the 10 different metaheuristic algorithms find strong are first identified. Then, if the feature found by one algorithm matches the feature found by the other algorithm, it is considered a strong feature. As the last step, the feature importance weights of these strong features are calculated with the ReliefF algorithm. The features with the highest weights are selected according to the number of features that the user will specify. The pseudo-code of the CMO-R algorithm is given in Algorithm 1.

E. PERFORMANCE METRICS

Accuracy (ACC) is the most popular metric used in classification problems. This is why accuracy is used as the main evaluation metric in this study. Sensitivity (Sn), precision (Pr), specificity (Sp), and F-score metrics should also be used when there is an imbalance in the number of samples in the classes. The Area Under Curve (AUC) value also provides information about the classification success in the classes. All of these metrics are calculated from the true positive (TP), true negative (TN), false positive (FP), and false negative values in the confusion matrix. The ACC, Sn, Sp, Pr, and F-score metrics calculations are given in Eqs. 2-6, respectively.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

Algorithm 1 The pseudo-code of the CMO-R Algorithm

CMO-R Algorithm

```
Input: feature vector (fea), label, and Output:
feature_selection
begin
function findRepeatingNumbers(matrix):
for each row in the matrix:
   for each number in row:
      if number is in repeating numbers:
         continue
      else number repeats in the matrix
         add number to repeating numbers
  return repeating numbers
function ReliefF(X,Y):
   calculate weights by Eq.1
    indices = sort(weights)
   selected _indices = indices[selected_features]
return selected _indices
function CMO-R(X,Y):
   indices_1 = ASO(X,Y)
   indeces_2 = EO(X,Y)
   indices_3 = GNDO(X,Y)
   indices_4 = HGSO(X,Y)
   indeces_5 = HHO(X,Y)
   indeces_6 = MPA(X,Y)
   indices_7 = MRFO(X,Y)
   indices_8 = PSO(X,Y)
   indices_9 = SMA(X,Y)
   indeces_10 = WOA(X,Y)
   combined_indices= [indices_1,...,indices10]
   mathcing_indices =
   findRepeatingNumbers(combined_indices)
  fea_selected1 = fea[matching_indices]
   feature_indices = reliefF[fea_selected1,Y]
   selected_features =
   fea_selected1[feature_indices]
return selected_features
feature_selection = CMO-R(fea,label)
end
```

$$Sn = \frac{TP}{TP + FN} \tag{3}$$

$$Sp = \frac{TN}{TN + FP} \tag{4}$$

$$Pr = \frac{TP}{TP + FP} \tag{5}$$

$$F - score = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{6}$$

The AUC metric is calculated as the area under the ROC plot. The ROC plot is plotted as the sensitivity (true positive rate) change against the 1-specificity (false positive rate) metric.

VI. EXPERIMENTAL STUDIES

All coding of the proposed approach and other algorithms were performed on MATLAB (2022b) installed Windows 11 pro equipped with the i9 intel processor, 64 GB memory, and 24 GB graphical card (NVIDIA RTX 4090). MATLAB program was run in GPU mode. The dataset for training and validation processes was divided using the 10-fold cross-validation. Training option parameters for all transfer learning approaches (VGG16, SquzeeNet, AlexNet, ResNet50, and MobileNetV2) were selected as the same. The mini-batch size, initial learning rate, and validation frequency

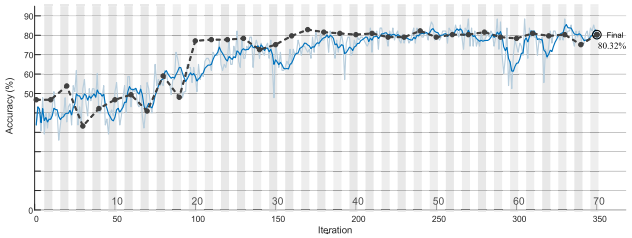


FIGURE 3. Training and test accuracy graphs for transfer learning with the VGG16 model.

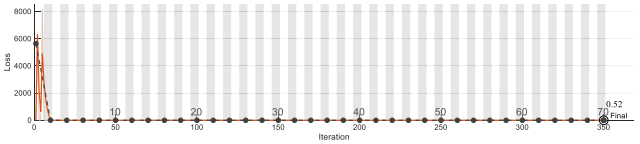


FIGURE 4. Training and test loss graphs for transfer learning with the VGG16 model.

parameters were 64, 0.001, and 30, respectively. The optimizer solver was tuned as the Stochastic Gradient Descent with Momentum (SGDM). The VGG16 model with transfer learning was trained and tested on the dataset. Accuracy and loss graphs (loss function: cross-entropy) for training and testing are given in Figures 3 and 4. At the end of 350 iterations, the training and test accuracies were 80.32% (Figure 3). As seen in Figure 4, the loss value of the cross-entropy function was 0.52.

In the other stage of this study, 5096 deep features were extracted by using the weights of the fc7 and fc8 layers of the VGG16 model. In the classification phase, a new algorithm strategy has been applied to both reduce the computational cost and increase the classification success. The CMO algorithm selected 518 distinctive features from the VGG16-based feature set. After this step, a threshold value of -0.0075 was applied to the Relieff-based weight values in the CMO-R algorithm, and 58 features were selected. Relieff-based weight values in the feature set obtained from the CMO algorithm are given in Fig. 5. In Fig. 6, the 3D representations of deep features are given in three different cases. In case 1, CMO-R and CMO feature selection algorithms were not used for the feature set. In cases 2 and 3, CMO and CMO-R feature selection algorithms were applied to the feature set. As can be seen in Fig. 6, the formal distinctiveness between minor and major classes gradually increases from case 1 to case 3.

In Fig. 7, confusion matrix results are given for the MNN classifier with the CMO-R algorithm. The proposed approach achieved a 91.5% accuracy. True positive (TP), true negative (TN), false positive (FP), and false negative values in the confusion matrix were 213, 268, 25, and 12, respectively. As can be seen from the confusion matrix, there is not much difference in accuracy and error rates between classes.

VII. DISCUSSION

In this section, ablation studies are performed for the strategies used in the model. Studies on asphalt cracks are

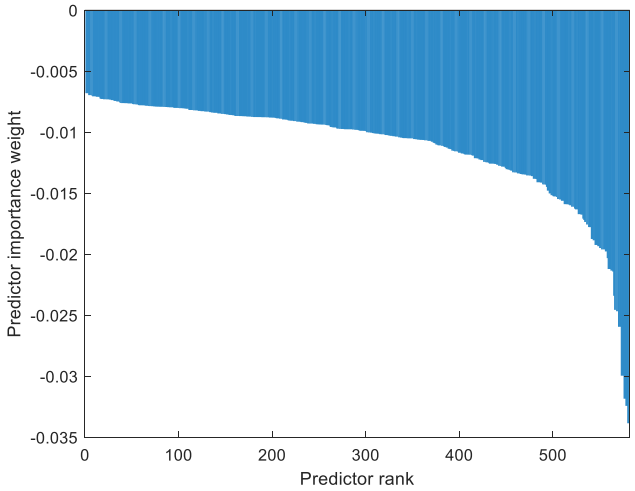


FIGURE 5. Relieff-based feature weights for 518 features (CMO algorithm).

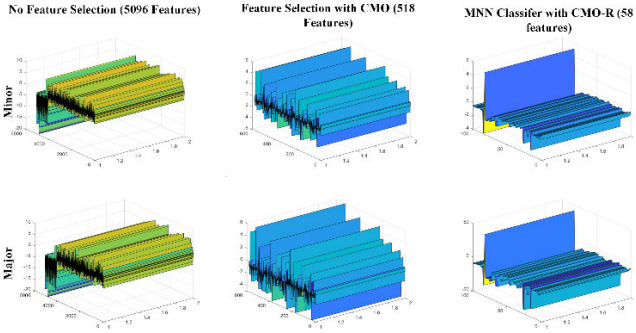


FIGURE 6. 3D feature representations for three cases: case 1 (no feature selection), case 2 (feature selection with CMO), case 3 (feature selection with CMO-R).

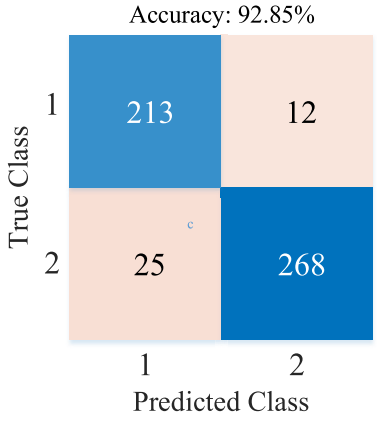


FIGURE 7. Confusion matrix results for the MNN classifier + CMO-R feature selection.

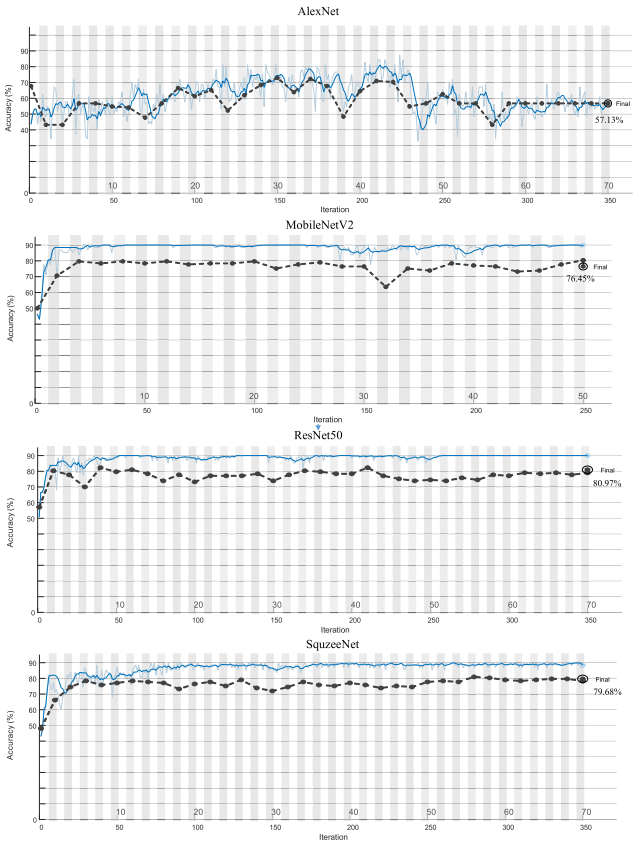


FIGURE 8. Accuracy scores of other popular pre-trained CNN models for transfer learning.

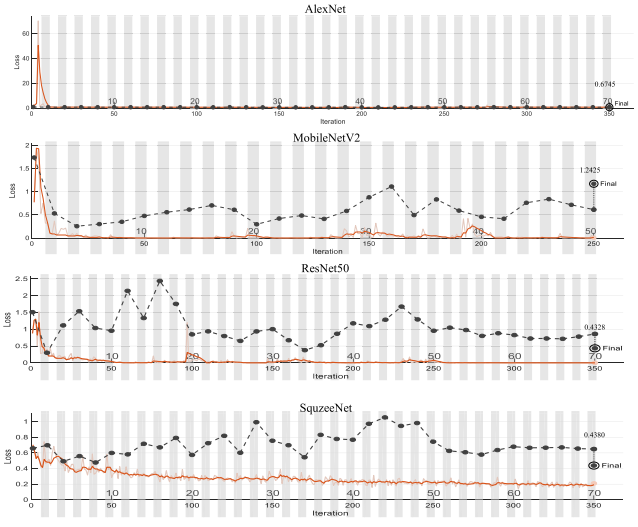


FIGURE 9. Loss scores of other popular pre-trained CNN models for transfer learning.

also included. The transfer learning results of other popular pre-trained CNN models in the first ablation study are given in Figure 8. In the first ablation study, other popular pre-trained CNN models were trained and tested with transfer learning. Figures 8 and 9 show the accuracy and loss results for Alexnet, MobileNetV2, ResNet50, and SquzeeNet, respectively. As can be seen from Figure 9, the best test accuracy (80.92%) score was achieved with the ResNet50 model, while the worst test accuracy (57.13%) score was

FIGURE 10. Confusion matrix results on deep features (4096 features) for VGG16 models.

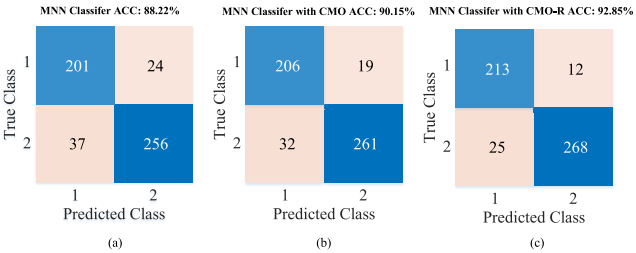


FIGURE 11. Confusion matrix results for MNN classifier according to feature selection cases: (a) case 1 (no feature selection), (b) case 2 (CMO feature selection), (c) case 3 (CMO-R feature selection).

TABLE 1. Performance metrics of classifiers on deep features (4096 features) for VGG16 models.

Classifier	Classes	TP	TN	FP	FN	Sn	Sp	Pr	F-Score	ACC
MNN	Minor	201	256	37	24	0.89	0.87	0.84	0.86	88.2%
	Major	256	201	24	37	0.87	0.89	0.91	0.89	
SVM	Minor	224	219	74	1	1	0.75	0.75	0.86	85.5%
	Major	219	224	1	74	0.75	1	1	0.86	
DT	Minor	196	256	37	29	0.87	0.87	0.84	0.85	87.3%
	Major	256	196	29	37	0.87	0.87	0.9	0.88	
KNN	Minor	199	255	38	26	0.88	0.87	0.84	0.86	87.6%
	Major	255	199	26	38	0.87	0.88	0.91	0.89	
LD	Minor	223	233	60	2	0.99	0.8	0.79	0.88	88.0%
	Major	233	223	2	60	0.8	0.99	0.99	0.88	
NB	Minor	198	256	37	27	0.88	0.87	0.84	0.86	87.6%
	Major	256	198	27	37	0.87	0.88	0.9	0.88	

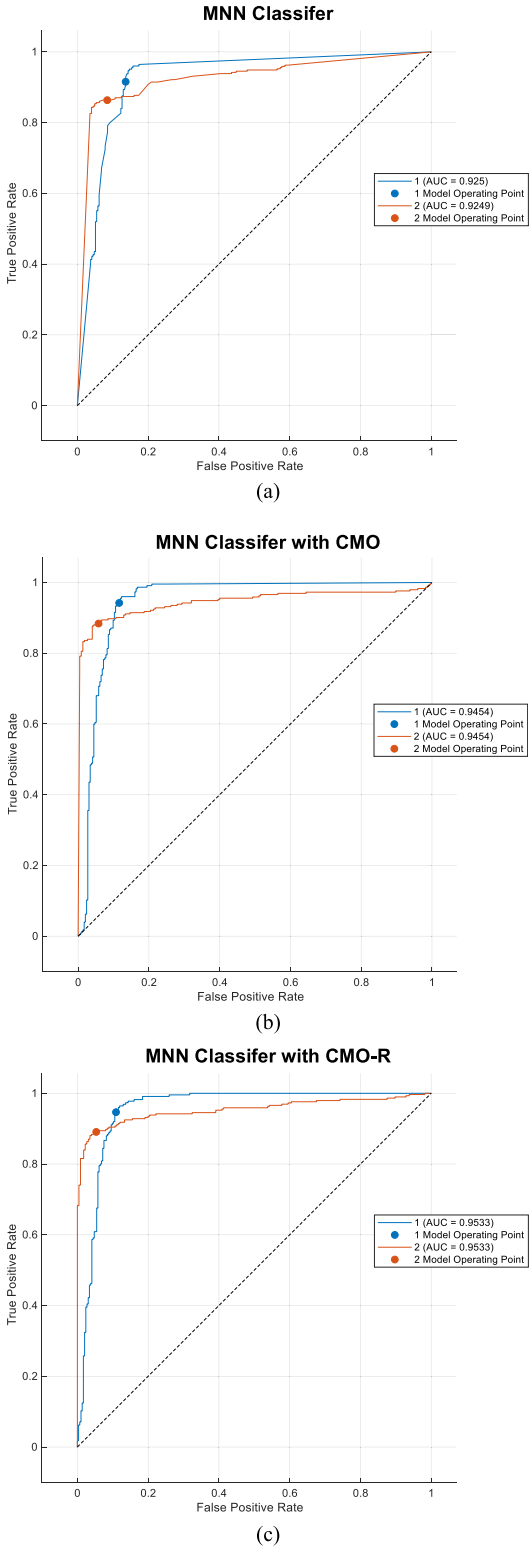


FIGURE 12. ROC curves and AUC values for MNN classifier according to feature selection cases:(a) case 1 (no feature selection), (b) case 2 (CMO feature selection), (c) case 3 (CMO-R feature selection).

achieved with the AlexNet model. The test accuracy of the VGG16 model used in the proposed approach was 0.65% lower than the test accuracy of the ResNet50 model. However, as seen in Figure 8, the VGG16 model was chosen for the proposed approach because of the overfitting problem in the ResNet50 model. In addition, overfitting problems occurred in models other than AlexNet and VGG16. Loss values of AlexNet, MobileNetV2, ResNet50, and SqzeeNet models were 0.6745, 1.2425, 0.4328, and 0.4280, respectively.

TABLE 2. Performance metric results according to the confusion matrix values in Fig. 11.

Cases	Classes	TP	TN	FP	FN	Sn	Sn	Pr	F-Score
1	Minor	198	256	37	27	0.88	0.87	0.84	0.86
	Major	256	198	27	37	0.87	0.88	0.9	0.88
2	Minor	206	261	32	19	0.92	0.89	0.87	0.89
	Major	261	206	19	32	0.89	0.92	0.93	0.91
3	Minor	213	268	25	12	0.95	0.91	0.9	0.92
	Major	268	213	12	25	0.91	0.95	0.96	0.94

Figure 10 shows the classification process using deep features extracted from the VGG16 model. To achieve the best performance in the classification process, 6 popular classifiers in machine learning were used, including MNN, SVM, DT, KNN, LD, and NB.

Table 1 shows the performance metrics generated using the values in the confusion matrices in Figure 10. Thus, sensitivity (Sn), specificity (Sp), precision (Pr), and F-score values were calculated for each classifier and class. Accuracy (ACC) scores were also calculated for each classifier. As seen in Table 1, the best accuracy value was obtained with the MNN classifier (88.2%), while the worst accuracy value was obtained with the SVM classifier (%85.5). The best average Sn (0.9), Sp (0.9), Pr (0.89), and F-score (0.88) values were obtained with the LD classifier. However, the performance metric scores between the two classes of the MNN classifier are more balanced.

Figure 11 shows the MNN classifier confusion matrix results according to the feature selection cases. MNN classifier accuracy was 88.22% without feature selection (case 1), 90.15% with CMO (case 2), and 92.85% with CMO-R (case 3).

Table 2 shows the performance metric results calculated based on the confusion matrix results in Figure 11. The best performance values in Sn, Sp, Pr, and F-score values for both classes were achieved with case 3 (MNN classifier + CMO-R feature selection).

Figure 12 shows the ROC curves and AUC values for the 3 different feature selection cases mentioned in Figure 11. As seen in Figure 12, the average AUC values for cases 1, 2, and 3 were 0.925, 0.9454, and 0.9533.

VIII. CONCLUSION

In this study, a two-class dataset is created to represent the condition of asphalt cracks on highways after the earthquakes in Turkey on February 6, 2023. First, a transfer learning approach was performed with 5 different popular pre-trained CNN models. The most optimal performance (80.32% ACC) with no overfitting was achieved by the VGG16 model. Deep features were extracted from the fully connected layers of the VGG16 model to improve the classification performance. These features were tested on 6 different popular classifiers. The best classification performance was obtained with the MNN classifier. In the next step, a new feature selection algorithm is used to further improve the classification performance. This algorithm improved the classification performance by 4.63% and achieved 92.85% classification accuracy.

The proposed methodology faces several limitations. Firstly, its reliance on a relatively small dataset raises concerns about its generalizability to larger and more diverse datasets, as the transfer learning approach, specifically utilizing the VGG16 model, was necessitated by this constraint. Additionally, the dependency on pre-trained models like VGG16 introduces a potential limitation in adaptability to different network architectures. The incorporation of the Combined Metaheuristic Optimization-Relieff (CMO-R) algorithm, involving 10 different metaheuristic optimization algorithms, adds a significant level of algorithmic complexity that may result in increased computational costs, limiting its practicality for real-time or resource-constrained applications. The extensive feature selection process, relying on CMO-R and ReliefF algorithms, raises concerns about overfitting to the training data. The methodology's reliance on a single classifier, the Medium Neural Network (MNN), based on its performance, may restrict adaptability to diverse classification scenarios where other classifiers might perform better. Furthermore, the limited explanation of algorithm choices within CMO-R hinders reproducibility and understanding. The evaluation based on data from specific earthquake events may limit the generalizability of the methodology to other disaster types or scenarios. Lastly, the application of a thresholding value in the ReliefF algorithm for feature reduction introduces subjectivity, impacting the final feature set and overall model performance. In conclusion, while the proposed methodology shows promise in addressing specific challenges, careful consideration of these limitations is essential for a nuanced interpretation of its applicability and performance across various contexts.

ACKNOWLEDGMENT

Also, the central concept of this research originated from the master's thesis entitled "Analysis of Asphalt Deformations on Highways Following Severe Earthquakes and Automatic Classification of Asphalt Cracks Using Deep Learning Approaches." The study was conducted under the guidance of Prof. Dr. Mehmet Yılmaz and Assoc. Prof. Dr. Fatih Demir at the Faculty of Technology, Department of Mechatronics Engineering, Fırat University.

REFERENCES

[1] J. S. Miller and W. Y. Bellinger, *Distress Identification Manual for the Long-Term Pavement Performance Program*. Washington, DC, USA: Federal Highway Administration. Office of Infrastructure, 2003.

[2] *Standard Practice for Roads and Parking Lots Pavement Condition Index Surveys*, document ASTM D6433, ASTM International, West Conshohocken, PA, USA, 2011, p. 49, no. 11.

[3] Q. Yang, W. Shi, J. Chen, and W. Lin, "Deep convolution neural network-based transfer learning method for civil infrastructure crack detection," *Autom. Construct.*, vol. 116, Aug. 2020, Art. no. 103199, doi: 10.1016/j.autcon.2020.103199.

[4] D. Dais, I. E. Bal, E. Smyrou, and V. Sarhosis, "Automatic crack classification and segmentation on masonry surfaces using convolutional neural networks and transfer learning," *Autom. Construct.*, vol. 125, May 2021, Art. no. 103606, doi: 10.1016/j.autcon.2021.103606.

[5] F. Liu, J. Liu, and L. Wang, "Asphalt pavement crack detection based on convolutional neural network and infrared thermography," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 11, pp. 22145–22155, Nov. 2022.

[6] F. Liu and L. Wang, "UNet-based model for crack detection integrating visual explanations," *Construction Building Mater.*, vol. 322, Mar. 2022, Art. no. 126265.

[7] Z. Liu, Y. Cao, Y. Wang, and W. Wang, "Computer vision-based concrete crack detection using U-Net fully convolutional networks," *Autom. Construct.*, vol. 104, pp. 129–139, Aug. 2019.

[8] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, "Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection," *Construct. Building Mater.*, vol. 157, pp. 322–330, Dec. 2017.

[9] J. Huyan, W. Li, S. Tighe, Z. Xu, and J. Zhai, "CrackU-Net: A novel deep convolutional neural network for pixelwise pavement crack detection," *Struct. Control Health Monitor.*, vol. 27, no. 8, Aug. 2020, Art. no. e2551.

[10] V. Mandal, L. Uong, and Y. Adu-Gyamfi, "Automated road crack detection using deep convolutional neural networks," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2018, pp. 5212–5215.

[11] H. Majidifard, Y. Adu-Gyamfi, and W. G. Buttlar, "Deep machine learning approach to develop a new asphalt pavement condition index," *Construct. Building Mater.*, vol. 247, Jun. 2020, Art. no. 118513.

[12] J. Guan, X. Yang, L. Ding, X. Cheng, V. C. S. Lee, and C. Jin, "Automated pixel-level pavement distress detection based on stereo vision and deep learning," *Autom. Construct.*, vol. 129, Sep. 2021, Art. no. 103788.

[13] B. Li, K. C. P. Wang, A. Zhang, E. Yang, and G. Wang, "Automatic classification of pavement crack using deep convolutional neural network," *Int. J. Pavement Eng.*, vol. 21, no. 4, pp. 457–463, Mar. 2020.

[14] J. Zhu, J. Zhong, T. Ma, X. Huang, W. Zhang, and Y. Zhou, "Pavement distress detection using convolutional neural networks with images captured via UAV," *Autom. Construct.*, vol. 133, Jan. 2022, Art. no. 103991.

[15] J. Liu, X. Yang, S. Lau, X. Wang, S. Luo, V. C. Lee, and L. Ding, "Automated pavement crack detection and segmentation based on two-step convolutional neural network," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 35, no. 11, pp. 1291–1305, Nov. 2020.

[16] F. Liu, J. Liu, and L. Wang, "Asphalt pavement fatigue crack severity classification by infrared thermography and deep learning," *Autom. Construct.*, vol. 143, Nov. 2022, Art. no. 104575.

[17] T. S. Tran, V. P. Tran, H. J. Lee, J. M. Flores, and V. P. Le, "A two-step sequential automated crack detection and severity classification process for asphalt pavements," *Int. J. Pavement Eng.*, vol. 23, no. 6, pp. 2019–2033, May 2022.

[18] R. Ribani and M. Marengoni, "A survey of transfer learning for convolutional neural networks," in *Proc. 32nd SIBGRAPI Conf. Graph., Patterns Images Tuts. (SIBGRAPI-T)*, Oct. 2019, pp. 47–57.

[19] C. Tan, "A survey on deep transfer learning," in *Proc. Int. Conf. Artif. neural Netw.*, 2018, pp. 270–279.

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

[21] T. Dokeroglu, E. Sevinc, T. Kucukyilmaz, and A. Cosar, "A survey on new generation metaheuristic algorithms," *Comput. Ind. Eng.*, vol. 137, Nov. 2019, Art. no. 106040.

[22] M. Abdel-Basset, L. Abdel-Fatah, and A. K. Sangaiah, "Metaheuristic algorithms: A comprehensive review," in *Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications* (Intelligent Data-Centric Systems). Elsevier, 2018, pp. 185–231.

[23] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Future Gener. Comput. Syst.*, vol. 97, pp. 849–872, Aug. 2019.

[24] A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, "Marine predators algorithm: A nature-inspired metaheuristic," *Expert Syst. Appl.*, vol. 152, Aug. 2020, Art. no. 113377.

[25] F. A. Hashim, E. H. Houssein, M. S. Mabrouk, W. Al-Atabany, and S. Mirjalili, "Henry gas solubility optimization: A novel physics-based algorithm," *Future Gener. Comput. Syst.*, vol. 101, pp. 646–667, Dec. 2019.

[26] Y. Zhang, Z. Jin, and S. Mirjalili, "Generalized normal distribution optimization and its applications in parameter extraction of photovoltaic models," *Energy Convers. Manage.*, vol. 224, Nov. 2020, Art. no. 113301.

[27] A. Faramarzi, M. Heidarinejad, B. Stephens, and S. Mirjalili, "Equilibrium optimizer: A novel optimization algorithm," *Knowl.-Based Syst.*, vol. 191, Mar. 2020, Art. no. 105190.

[28] W. Zhao, L. Wang, and Z. Zhang, "Atom search optimization and its application to solve a hydrogeologic parameter estimation problem," *Knowl.-Based Syst.*, vol. 163, pp. 283–304, Jan. 2019.

[29] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, Feb. 2016.

[30] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization," *Future Gener. Comput. Syst.*, vol. 111, pp. 300–323, Oct. 2020.

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

[32] W. Zhao, Z. Zhang, and L. Wang, "Manta ray foraging optimization: An effective bio-inspired optimizer for engineering applications," *Eng. Appl. Artif. Intell.*, vol. 87, Jan. 2020, Art. no. 103300.

[33] K. Demir, B. Ari, and F. Demir, "Detection of brain tumor with a pre-trained deep learning model based on feature selection using MR images," *FIRAT Univ. J. Experim. Comput. Eng.*, vol. 2, no. 1, pp. 23–31, 2023.

[34] K. Demir, M. Ay, M. Cavas, and F. Demir, "Automated steel surface defect detection and classification using a new deep learning-based approach," *Neural Comput. Appl.*, vol. 35, no. 11, pp. 8389–8406, Apr. 2023.

[35] F. Demir, Y. Akbulut, B. Taşcı, and K. Demir, "Improving brain tumor classification performance with an effective approach based on new deep learning model named 3ACL from 3D MRI data," *Biomed. Signal Process. Control*, vol. 81, Mar. 2023, Art. no. 104424.

[36] F. Demir, K. Demir, and A. Şengür, "DeepCov19Net: Automated COVID-19 disease detection with a robust and effective technique deep learning approach," *New Gener. Comput.*, vol. 40, no. 4, pp. 1053–1075, Dec. 2022, doi: 10.1007/s00354-021-00152-0.

[37] F. Demir, K. Siddique, M. Alswaitti, K. Demir, and A. Sengur, "A simple and effective approach based on a multi-level feature selection for automated Parkinson's disease detection," *J. Personalized Med.*, vol. 12, no. 1, p. 55, Jan. 2022, doi: 10.3390/jpm12010055.

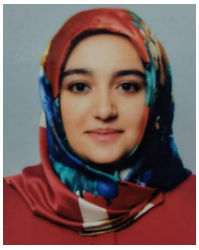
MEHMET YILMAZ received the B.Sc. and M.Sc. degrees in civil engineering and the Ph.D. degree in transportation engineering from Fırat University, Turkey, in 2003, 2005, and 2011, respectively. His research interests include flexible pavements and asphalt materials within the field of transportation engineering.

ERKUT YALÇIN received the B.Sc. and M.Sc. degrees in civil engineering from Fırat University, Turkey, in 2012 and 2014, respectively, and the Ph.D. degree from Fırat University, in 2018, with a focus on transportation engineering. His research interests include self-healing asphalt, flexible pavements, and asphalt materials.

SAIF KIFAH received the master's degree in information technology from the University of Kebangsaan Malaysia, in 2011, with a focus on implementing a decision support system by filtering data based on rough set theory, and the Ph.D. degree in 2017. His master's thesis was on the implementation of a meta-heuristic optimization approach to enhance the pre-processing phase in the data mining field. He was with the Xiamen University Malaysia. He is currently an Assistant Professor who teaches computer science-related courses with the School of Computing and Data Science, Xiamen University Malaysia. He also holds the position of a program coordinator of the data science program with Xiamen University Malaysia. His published works deal with the application of the meta-heuristic algorithm to handle the scheduling problem. His research interests include the application of artificial intelligence methods to handle real-world problems.

FATIH DEMIR received the B.Sc., M.Sc., and Ph.D. degrees in electrical and electronics engineering from Fırat University, Turkey, in 2007, 2010, and 2020, respectively. His research interests include automatic classification, image processing, and deep learning.

ABDULKADIR ŞENGÜR received the B.Sc. degree in electronics and computer education, the M.Sc. degree in electronics education, and the Ph.D. degree in electrical and electronics engineering from Fırat University, Turkey, in 1999, 2003, and 2006, respectively. He was a Research Assistant with the Technical Education Faculty, Fırat University, in February 2001. He is currently a Professor with the Technology Faculty, Fırat University. His research interests include signal processing, image segmentation, pattern recognition, medical image processing, and computer vision.



ROZERIN DEMIR received the bachelor's degree from the Department of Civil Engineering, Fırat University, Turkey, in 2020, and the M.Sc. degree from the Department of Geodesy and Geographic Information Technologies, Fırat University, in 2023.

RAJA MAJID MEHMOOD (Senior Member, IEEE) received the B.S. degree in computer science from Gomal University, Pakistan, the M.S. degree in software technology from Linnaeus University, Sweden, and the Ph.D. degree in computer engineering from the Division of Computer Science and Engineering, Chonbuk National University, South Korea.

He was a Research Professor with the Department of Brain and Cognitive Engineering, Korea University, Seoul, South Korea. He was a Lecturer with the Software Engineering Department, King Saud University, Saudi Arabia. He is currently an Assistant Professor with the School of Computing and Data Science, Xiamen University Malaysia, Sepang, Malaysia. He is the world's top 2% scientist, according to Stanford University. He is the Founder of the High-Tech Research and Development Lab EJAD. He has authored more than 50 research articles and is supervising or co-supervising several graduate (M.S./Ph.D.) students. His research interests include affective computing, brain–computer interfaces, information visualization, image processing, pattern recognition, multi-task scheduling, and software engineering.

Dr. Mehmood has served as a Reviewer for various international journals, including *IEEE Communications Magazine*, IEEE TRANSACTIONS ON AFFECTIVE COMPUTING, *International Journal of Information Technology and Decision Making*, IEEE ACCESS, *Multimedia Tools and Applications*, *ACM Transactions on Multimedia Computing, Communications, and Applications*, *Journal of King Saud University*, *Computer and Information Sciences*, *Biomedical Engineering* (BME), IEEE INTERNET OF THINGS JOURNAL, and IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS.