Electronic version of an article published as International Journal of Information Technology and Decision Making,Vol. 20, No. 06, (2021), pp. 1561-1614, https://doi.org/10.1142/S0219622021500425 © World Scientific Publishing Company https://www.worldscientific.com/worldscinet/ijitdm.
The following publication Abdelkader, E. M., Moselhi, O., Marzouk, M., & Zayed, T. (2021). A grey wolf optimization-based method for segmentation and evaluation of scaling in reinforced concrete bridges. International Journal of Information Technology & Decision Making, 20(06), 1561-1614 is available at https://www.worldscientific.com/doi/abs/10.1142/S0219622021500425.

A Grey Wolf Optimization-Based Method for Segmentation and Evaluation of Scaling in Reinforced Concrete Bridges

ABSTRACT

Bridges are prone to severe deterioration agents which promote their degradation over the course of their lifetime. Furthermore, maintenance budgets are being trimmed. This state of circumstances entails the development of a computer vision-based method for the condition assessment of bridge elements in an attempt to circumvent the drawbacks of visual inspection-based models. Scaling is progressive local flaking or loss in the surface portion of concrete that affects the functional and structural integrity of reinforced concrete bridges. As such, the present research study proposes a self-adaptive three-tier method for the automated detection and assessment of scaling severity levels in reinforced concrete bridges. The first tier relies on the integration of cross entropy function and grey wolf optimization algorithm for the segmentation of scaling pixels. The second tier is designated for the autonomous interpretation of scaling area. In this model, a hybrid feature extraction algorithm is proposed based on the fusion of singular value decomposition and discrete wavelet transform for the efficient and robust extraction of the most dominant features in scaling images. Then, an integration of Elman neural network and grey wolf optimization algorithm is proposed for the sake of improving the prediction accuracies of scaling area though optimization of both structure and parameters of Elman neural network. The third tier aims at establishing a unified scaling severity index to assess the extent of severities of scaling according to its area and depth. The developed method is validated through multi-layered comparative analysis that involved performance evaluation comparisons, statistical comparisons and box plots. Results demonstrated that the developed scaling detection model significantly outperformed a set of widely-utilized classical segmentation models achieving mean squared error, mean absolute error, peak signal to noise ratio and cross entropy of 0.175, 0.407, 55.754 and 26011.019, respectively. With regards to the developed scaling evaluation model, it accomplished remarkable better and more robust performance that other meta-heuristic-based Elman neural network models and conventional prediction models. In this context, it obtained mean absolute percentage error, root-mean squared error and mean absolute error 1.513%, 29.836 and 12.066, respectively as per split validation. It is anticipated that the developed integrated computer vision-based method could serve as the basis of automated, reliable and cost-effective inspection platform of reinforced concrete bridges which can assist departments of transportation in taking effective preventive maintenance and rehabilitation actions.

Keywords: Bridges; computer vision; scaling; cross entropy; grey wolf optimization; discrete wavelet transform; Elman neural network

1. INTRODUCTION

Bridges are critical elements of civil infrastructure systems that are vital for economic developments and public welfare. As such, they should be continuously monitored to preserve their performance condition statuses and prevent them from further degradation despite encountered severe operating and environmental conditions. Deterioration of bridges is one of the major concerns for transportation agencies over the last few years. Recently, the number of bridges has increased drastically. Additionally, there are large numbers of existing bridges in transportation networks. On top of that, there are limited funds available for maintenance, repair and rehabilitation actions. This state of affairs motivated researchers to give particular attention to the condition assessment of bridges, which paves the way for the efficient planning and prioritization of their maintenance activities. In Canada, the instantaneous and serious economic and environmental impacts of bridge collapses besides the high owner and user costs have drawn the attention to the importance of bridge management systems. Bridges experience accelerated aging and extensive deterioration and larger portion of them require urgent rehabilitation or replacement. The deterioration agents encompass variable traffic overloading, chloride ingress, cycles of the freeze and thaw, poor construction practices, earthquakes, etc.

According to the Canadian infrastructure report card, 26% of the bridges are either in a “Fair”, “Severe” or “Very Severe” conditions. One-third of Canada’s bridges were reported to experience structural or functional deficiencies with short remaining service life, such that 15,000 public transits, 750,000 trucks and 20 million light vehicles utilize the Canadian bridges annually. It was reported that average age of bridges reached 24.5 years in 2007 while their average age was 43.3 years. Thus, fifty seven percentage of the estimated service life has already been consumed. It is worth mentioning that the average age of bridges is considered as the second highest among the five main assets, namely roads, bridges, water supply systems, wastewater treatment facilities and sewer systems. Bridges in Quebec reached higher levels of deterioration such that they consumed 72% of their useful lifetime, which is regarded as a higher average age than any other province in Canada. On the other side, Prince Edward Island had the smallest average age of 15.6 years. This can be explained by the fact that approximately 70% of Quebec’s bridges were constructed between the 1960s and 1980s.[1,2,3,4]

Visual inspection is considered as the common practice adopted by transportation agencies to monitor condition of the bridge decks. These inspections are carried out at equal time intervals by engineers to evaluate the severities of the bridge defects. The condition assessment models established based on the visual inspection are error-prone because of the inherent subjectivity arising from being highly dependent on the skills and experience of inspectors, which creates wide variations among the evaluations of the inspectors. Furthermore, routine visual inspection is criticized for being laborious, costly, time-consuming and dangerous. As such, these circumstances necessitate the development of computer vision-based method that aids in enhancing the condition assessment accuracy and minimizing the costs elicited from visual inspection.[5,6,7]

Surface defects are the most observable indication of possible structural deterioration. They are viewed as the cornerstone of most inspection manuals since they implicate an accurate reflection of the condition of the structural member. Scaling is a surface deterioration mechanism which can be defined as flaking or peeling of finished hardened concrete surface more often due to the exposure to cycles of freezing and thawing and the utilization of de-icer chemicals, whereas concrete pores near the surface thaws and freezes as a result of the temperature fluctuations. The freezing of water in saturated concrete establishes substantial expansive forces that cause the concrete surfaces to be scaled off mainly when it is not well-protected with entrained air. It starts with small patches and it progresses with time to extend to large areas, whereas its severity varies from light scaling to severe scaling. Light scaling doesn’t involve exposure of coarse aggregate. Severe scaling involves loss of mortar and coarse aggregate particles up to a depth greater than 20 mm. Scaling affects the functional performance of the structural element because it influences the riding quality and safety of traffic. Moreover, it has been reported that scaling can expose the concrete to ready ingress of moisture and aggressive salts, which accelerates the deterioration of concrete and may influence the structural durability at the latter stages. Therefore, it can be interpreted that efficient mapping and evaluation of scaling can eventually lead to the accurate assessment of the condition of the bridge element.[8,9]

2. LITERATURE REVIEW

Several previous machine learning-based and deep learning-based models were developed for the detection and recognition of surface defects in reinforced concrete bridges. Wang *et al.*[10] proposed an integrated approach for the detection of cracks in reinforced concrete bridges. In it, non-linear median filter was applied in the form of three continuous groups for removing noises from the original gray images. Then, an improved threshold segmentation model was proposed based on the integration of Otsu algorithm and modified Sobel operator. It was found that the developed approach managed to achieve an absolute error of 0.02 mm in the crack width. Yu *et al.*[11] presented a crack detection algorithm for the detection of cracks in bridges based on a set of digital image processing techniques. In the developed model, a visual comparison between a set of filtering techniques was carried out, and it was inferred that orientation filtering was able to efficiently remove the noise to the maximum extent while retaining cracks' edge information. Gamma transformation was used to correct the image and improve its contrast. Otsu algorithm was utilized to detect the crack through computing local thresholds for the different regions of the image. Zhang and Suen algorithm was implemented to extract the crack's skeleton and Hough line algorithm was used to identify the different fragments and trends of multiple cracks. It was highlighted that the developed algorithm was able to provide engineers with a platform to measure widths of irregular cracks.

Lei *et al.*[6] developed a method for the crack detection based on the crack central point (CCPM) algorithm. Gaussian filter was applied to remove noise from images and restore them. They highlighted that the designated crack can be distinguished capitalizing on the existing minimum gray value in the row of crack area, which usually takes the form of parabolic distribution. They also urged that the developed method can accurately compute a separating threshold that can efficiently extract the crack from the images collected using the unmanned aerial vehicle. Zhang and Wang[12] introduced a combined model for the classification of bridge cracks. Median filter and contrast enhancement were applied to reduce the corrupting noises and lessen the effects of background texture and details. Morphological analysis was then carried out to segment the crack details and eliminate noise bocks. Visual geometry group network (VGG16) was used to extract and realize features of the input segmented images. Finally, the extracted features were fed into support vector machines model to classify the images to either horizontal cracks, vertical cracks, slope cracks and block cracks. It was concluded that the developed model outperformed support vector machines, convolutional neural network and projection algorithm providing detection accuracies above 90% for the four types of cracks.

Noh *et al.*[13] proposed a method for the automated detection of fine concrete cracks in bridges. Segmentation was performed using fuzzy C-means clustering to differentiate between cracked and non-cracked regions according to the average brightness of the pixels. Dilation morphological operation, grassfire search algorithm and connected component labeling were used for noise removal and better representation of the cracks' information. Experimental results indicated that the developed model accomplished higher precision and recall when compared against edge detection algorithms. Xu *et al*.[14] presented a deep learning-based model for concrete bridge crack detection. In the proposed model, the output feature map of the convolutional layers was fed into Atrous Spatial Pyramid Pooling (ASPP) module to identify the multi-scale crack feature information. The ASPP module encompassed depthwise separable convolution for reducing the computational complexity of the model while maintaining an appropriate prediction performance. The proposed model outperformed a set of transfer learning-based deep neural networks such as Resnet18, Resnet34, Resnet50, VGG16 and VGG19 providing accuracy, precision, sensitivity, specificity and $\mathrm{F1-score}$ of 96.37%, 78.11%, 100%, 95.83% and 0.8771, respectively.

Kim *et al.*[5] identified bridge cracks using unmanned aerial vehicles equipped with a high resolution vision sensor. A 3D point cloud-based background model was generated in the preliminary flight to create the damage map. Region-based convolutional neural network (RCNN) was then implemented for the localization of cracks and quantification of their sizes. The developed crack quantification algorithm was able to achieve a relative error ranging from 1% to 2%. Li *et al.*[15] presented automated bridge crack detection and evaluation model based on the fusion of fully convolutional neural network and naïve Bayes algorithm. Label-maintaining transformation was applied to augment the original dataset using sliding window, rotation and slipping. In this regard, the output feature map of the convolutional neural network model was passed into naïve Bayes model to determine whether concrete images contain cracks. The developed detection model surpassed crack tree algorithm, random structured forest, relatively competitive convolutional neural network and deep fusion convolutional neural network obtaining error rate of 1.28%. Furthermore, the developed quantification model yielded accuracy rates of 93.2% and 92.8% in interpreting cracks width and length, respectively.

Yan *et al.*[16] developed an encoder-decoder model for the autonomous detection of bridge cracks. The encoder comprised convolutional layers convolutional layers and pooling layers to generate the input feature map and reduce the computational burden, respectively. The decoder utilized the de-convolution method to restore the feature map. A residual module is added to the $\mathrm{Res}-\mathrm{Unet}$ network to better realize crack pixels and background pixels. It was found that the developed model succeeded in localizing and identifying the different fragments of cracks. Zhang *et al.*[17] introduced computer vision-based model for bridge crack classification. Wavelet transform filter was used for de-noising and enhancing the quality of the input images. Otsu algorithm was applied for threshold segmentation based on the brightness and color of pixels in the images. Convolutional neural network was then applied to classify the types of cracks. Simulation results demonstrated that the developed model attained a correct classification rate of 92%, 95% and 90% in detecting small cracks, larger cracks and serious cracks, respectively.

Xie and Ming[18]developed a machine vision-based method for the detection of bridge cracks. Histogram equalization was carried out to obtain more uniform and continuous gray histogram. Median filtering, Gaussian filtering and Laplacian edge enhancement were deployed to remove salt and pepper noises as well as Gaussian noises. Otsu algorithm was then used to determine the optimum threshold value and detect the cracking features. Alexnet and Caffenet were utilized to identify the bridge cracks. It was concluded that the developed model could achieve a detection accuracy of 91.7%. Zhang *et al.*[19] proposed an improved $\mathrm{YOLO}-\mathrm{V3}$ algorithm for bridge surface crack detection. In the developed model, depthwise separable convolution replaced the standard convolution to reduce the network's parameters and computational effort. A convolutional block attention module was presented to create more effective adaptive learning of the features through multiplying the attention map with input feature map. The improved $\mathrm{YOLO}-\mathrm{V3}$ algorithm achieved precision and recall of 89.16% and 91.16%, respectively.

Ye *et al.*[20] presented a fully convolutional network model for the automated detection of structural cracks in bridges. The architecture of the fully convolutional network model comprised seven convolution layers, two up-sampling layers, two max-pooling layers, six deconvolution layers and a softmax layer. Pixel-level labeled images were used for training and testing the developed model. Results showed that the developed deep learning model yielded precision, recall, intersection over union and F-measure of 0.84, 0.82, 0.73 and 0.6, respectively. Droguett *et al.*[21] introduced a modified Densenet-based model for the semantic segmentation of crack images of concrete bridges. The modified topology of Densenet consisted of thirteen layers, one stage of feature extraction, and two stages of upsampling and downsampling. In the feature extraction module, each dense block represented a concatenation of convolutional units, whereas each convolutional unit dealt with a concatenation of all the previous units and the block unit. The modified architecture performed very well in crack segmentation attaining intersection over union of 94.51%.

Vignesh *et al.*[22] established a convolutional neural network-based model for the detection of bridge cracks. Gabor filter was employed to remove the noises, and enhanced adaptive thresholding module based on Otsu algorithm was proposed to better identify the crack edges. The convolutional neural network module included arous space pyramid pool to obtain the multi-dimensional context data, and depthwise separable convolution to reduce the number of model parameters and computational loss without adversely affecting the prediction results. The developed crack detection model outperformed the standard Resnet50 obtaining recall rate, accuracy rate, false alarm rate and missed alarm rate of 99.55%, 96.69%, 3.31% and 94.64%, respectively. Tian *et al.*[23] presented a concrete crack detection and evaluation model based on image processing techniques. Estimation of crack size was performed using scale algorithm and object-based algorithm Double edge statistical algorithm was then used to compute the cracking length. An improved scale invariant feature transform algorithm was introduced to deal with crack image mosaic. Results highlighted that the object-based algorithm performed better than the scale algorithm yielding accuracy above 90%. Also, the developed crack estimation algorithm was able to achieve 92% accuracy.

It can be inferred that most of the previous studies focused on the recognition and assessment of cracking, which reveals that there is lack of investigation of other surface defects such as scaling and spalling. The absence of spalling and scaling assessment models which can look at the evaluation of their severity levels, create incomprehensive and unreliable condition assessment models that in return can substantially influence the maintenance planning and prioritization models in the different managerial levels. Furthermore, most of the previous publications relied on Otsu algorithm followed by K-means clustering and then fuzzy C-means clustering and watershed algorithm for the segmentation of the defects from the background. Detection of scaling in images is an exhaustive task due to their complex texture patterns, presence of noise, uneven illumination and existence of low contrast between scaling and the background. These encountered conditions create multi-modal histograms which are difficult to be explored by the classical segmentation models and causes them to fail in segmenting scaling images efficiently. In this regard, it should be noted that searching multimodal histograms to find the optimum threshold is more exhaustive and sophisticated task to be achieved more than the unimodal histograms. It can be also inferred that there is lack of investigation of the optimization-based methods which are less invariant to the noise that may corrupt the images and yield more accurate results when compared against the classical segmentation methods mostly in the complex images.[24,25]

Some previous models utilized back propagation artificial neural networks for the detection and classification of cracks. In this context, their training process is normally carried out using gradient descent algorithm, which is based on finding the partial derivative of the error function with respect to each weight in an attempt to minimize the distances between the predicted and actual values. However, learning using gradient descent algorithm is highly vulnerable to poor convergence rate, local minima stagnation, over-fitting issues and low global search abilities. This restrains the neural network model from obtaining the most optimum configuration of weights.[26,27]

3. PROPOSED METHOD

The ultimate objective of the present study is to provide transportation agencies with an automated decision-making platform that aids in evaluation of scaling severities in reinforced concrete bridges capitalizing on a unified scaling severity index. This is addressed through establishing a three-tier paradigm for the detection and synthetic analysis of scaling. The framework of the proposed method is depicted in Figure 1. As can be seen, it encompasses three main models, namely scaling detection, scaling evaluation and scaling severity index. The first model is designed for the purpose of detection of scaling in reinforced concrete bridges. In it, the first step is to standardize the images to size of 200×200 in order to facilitate the further processing stages. The next stage involves conversion of true-color image RGB to the grayscale image to minimizing the computational effort meanwhile preserving the important features in the image. In the true-color RGB image; R G and B stand for red, green and blue, respectively. The lowest possible intensity value of R, G and B is zero while the highest possible value is 255. The conversion to grayscale image is performed through weighted average of the R, G and B colors as follows.[28]

$$\mathrm{G(i,j)} = 0.299 \times \mathrm{R(i,j)} + 0.587 \times \mathrm{G(i,j)} + 0.114 \times \mathrm{B(i,j)} \quad (1)$$

Where;

$\mathrm{G(i,j)}$ stands for the grayscale image.

Noise represents unwanted information that degrades the quality of the image and has adverse implication on the further stage of detection and evaluation of scaling. Image restoration can be performed in several domains such as spatial domain and frequency domain. Spatial domain techniques deal directly with the pixel intensities present in the image. Nevertheless, frequency domain filters are based on the Fourier transform of the image. The proposed method utilizes Wiener filer as a frequency domain filter to remove the maximum noise from the corrupted images while maintaining the significant characteristics and important features of the image. In this regard, the frequency domain filter is preferred over the spatial domain filter because it demonstrated higher restoration capacity with respect to other spatial domain filters in dealing with different types of noises.[29,30,31]

Bridges are complex structures due to the substantial and exhaustive amount of details and information present in images. Furthermore, they experience low contrast, color distortion, irregular texture pattern and inhomogeneous illumination conditions. Thus, min-max gray level discrimination approach is applied to magnify the differences between the scaling and the background. The min-max gray level discrimination approach increases the gray level intensities of the scaling pixels causing them to become darker, and it increases gray level intensities of non-scaling pixels causing them to become lighter. The enhanced image capitalizing on the min-max gray level discrimination approach can be obtained as follows.[32]

$$\mathrm{I_a(x,y)} = \begin{cases} \min(\mathrm{M,T}) \ \text{if} \ \mathrm{I_o(x,y)} > \mathrm{I_o}\min + \tau \times (\mathrm{S}) \\ \max(\mathrm{N,F}) \ \text{if} \ \mathrm{I_o(x,y)} \leq \mathrm{I_o}\min + \tau \times (\mathrm{S}) \end{cases} \quad (2)$$

Such that;

$$T = I_o(x, y) \times R_a,$$
$$F = I_o(x, y) \times R_a^{-1}$$

$$S = I_o max - I_o min$$

$$M = I_o max$$

$$N = I_o min$$

Where;

$I_a(x, y)$ and $I_o(x, y)$ represent the adjusted image and original image, respectively. $I_o min$ and $I_o max$ represent the minimum and maximum gray level intensities in the original image. τ and R_a denote the margin parameter and adjusted ratio, respectively. In this context, τ and R_a are set as 0.5 and 1.1, respectively.

The next stage is the image segmentation, whereas in it bi-level thresholding is performed to generate a single threshold T that classifies the image pixels into two classes, namely foreground (scaling) and background (surface). The bi-level thresholding function can be defined as follows.

$$B(x, y) = \begin{cases} 1, \text{if } F(x, y) \geq T \\ 0 \text{ , otherwise} \end{cases} \tag{3}$$

Where;

$B(x, y)$ represents the binary image. $F(x, y)$ represents the gray image. T denotes the threshold that separates the foreground scaling pixels from the background non-scaling pixels, whereas if the image pixels are above the threshold, they are appended to the foreground scaling pixels otherwise, they are appended to the background

Image segmentation methods can be categorized into five main clusters, namely edge detection-based methods, clustering-based methods, region-based methods, histogram-based methods, and optimization-based methods. In the recent few years, optimization-based methods have received considerable attention by researchers, and they demonstrated their superior segmentation capacities against other image segmentation methods. In this regard, image segmentation is formulated as optimization problems, whereas the optimum threshold is computed based on a predefined objective function such as maximizing of the between-class variance, maximization of the Kapur entropy, maximization of the Renyi's entropy and minimization of the Bayesian error.[33,34,35]

The proposed scaling segmentation model ($\mathrm{MCE} - \mathrm{GWO}$) utilizes grey wolf optimization (GWO) algorithm to search for the optimum threshold capitalizing on minimization of the cross entropy (MCE) between the original image and segmented image. Minimum cross entropy approach proved its higher segmentation capacity in different bi-level thresholding and multi-level thresholding applications. Furthermore, it requires less parameters to be calibrated when compared against other image segmentation approaches.[36,37,38] Grey wolf optimization algorithm is a newly-developed and efficient meta-heuristic in the area of swarm intelligence that exhibits proper trade-off between exploration and exploitation process which enables it to avoid local minima entrapment and this leads to improved convergence rate.[39,40] It has been previously-utilized in exploring complex and multi-local search spaces in diversified real-world applications such as designing reinforced concrete cantilever retaining wall[41], vehicular ad-hoc networks[42], water resources allocation[43], and economic load dispatch problems.[44] Furthermore, it outperformed some of the well-performing state of art meta-heuristics including genetic algorithm, particle swarm optimization algorithm, artificial bee colony algorithm, cuckoo search algorithm, bat algorithm, improved bat algorithm and gravitational search algorithm.[45,46,47,48] Another competitive advantage of grey wolf optimization algorithm is that it requires fewer control parameters to be calibrated which leads to less effort in the tuning process. This also implies that grey wolf optimization algorithm experiences less perturbations and more robust search performance.[49,50] Furthermore, grey wolf optimization is characterized by its simplicity, less memory requirements and ease of implementation.[51,52] As such, the characteristics of grey wolf optimizer make it suitable for the implementation in scaling detection and evaluation. Several improvements in the literature have been proposed to the classical grey wolf optimization algorithm in order to search for the global optimum solution in a faster and more efficient manner through integration with other meta-heuristics[53,54], adding new search strategies[55,56,57,58] and using chaotic operators.[59] Thus, the present study relies on the classical grey wolf optimization algorithm for the automated segmentation of scaling pixels and interpretation of scaling area.

The validation process of the scaling detection model is three-folded. The first fold is to substantiate the deployment of grey wolf optimization algorithm through comparison with high-performing state of art meta-heuristics, namely genetic algorithm (GA), particle swarm optimization (PSO) algorithm, harmony search (HS) algorithm, differential evolution (DE) algorithm and shuffled frog-leaping (SFL) algorithm. The second fold is conducted to justify the employment of the proposed optimization-based method. This encompasses its comparison against other types of segmentation models including: Otsu, K-means clustering (KM), fuzzy C-means clustering (FCM) and expectation maximization (EM). K-means aims at dividing the data observations into homogenous groups through minimizing the sum of squared error between the data points and their respective cluster's centroid over all clusters. Expectation maximization algorithm involves two stages, namely expectation and maximization. The expectation stage encompasses computing the cluster probability for every data instance. The maximization stage obtains the parameters of the distribution based on the clusters' probabilities through maximizing the likelihood of the distribution given the data instances.[60] The third fold is for the purpose of evaluating the statistical significance of the output of the proposed $\mathrm{MCE} - \mathrm{GWO}$ model against the afore-mentioned segmentation models. In this regard, Shapiro-Wilk test is applied first to study the normality of the data at significance level (α) of 0.05. Afterwards, parametric or non-parametric tests are performed relying on the assessment of normality of the data. Assessment of the performances of the optimization-based models is a complicated task because it usually comprises several conflicting performance indicators that need to be considered.[61] In this regard, the scaling segmentation models are evaluated as per three performance indicators which are: mean-squared error, mean absolute error, peak signal to noise ratio and cross entropy.

The second model aims at evaluation of scaling based on area whereas feature extraction plays a very fundamental role in it. Feature extraction is a dimensionality reduction algorithm that transforms a higher dimension dataset into a lower one meanwhile preserving the model prediction performance through eliminating redundant and uninformative attributes.[62] Feature extraction can be performed based on spatial domain analysis or frequency domain analysis. Spatial domain approach deals with physical parameters such that spatial domain features include texture, size, color, shape and edge intensity. Frequency domain approach relies on measuring parameters from an image, and the frequency domain features encompasses the coefficients of fast Fourier transform, discrete cosine transform (DCT) and discrete wavelet transform (DWT). Frequency domain represents a space in which each image value at a certain position F constitutes the amount that the intensity values in spatial domain image I vary over a specific distance with respect to position F. Thus, frequency domain demonstrates the rate at which image intensity values are changed in the spatial domain image I. High frequency components correspond to pixel values that transit rapidly across the image such as text and edges. Strong low frequency components correspond to large scale features in the images such as smooth regions, homogenous objects that dominate the image, and slow-varying character. It is worth mentioning that the DCT and DWT transformation algorithms enable the transition from the spatial domain to frequency domain, and the inverse transformation enables returning back to the original spatial space.[63,64]

The present study proposes a novel feature extraction method that capitalizes on cascading the higher efficiency capabilities of singular value decomposition (SVD) in capturing the intrinsic information and the robustness of discrete wavelet transform against proportion variance and rotation variance. In this context, singular value decomposition and discrete wavelet transform are adopted to model the spatial domain features and frequency domain features, respectively. This concatenation of features ($SVD - \mathrm{DWT}$) is expected to establish a trade-off that minimizes the complexity of the training process and computational time alongside enhancing the computational capacity of the machine learning model elicited from its ability in providing an accurate representation for the information in images.

The fusion of the singular value decomposition and discrete wavelet transform creates a feature vector that is then utilized to feed the machine learning model. In this regard, a hybrid Elman neural network-grey wolf optimization ($\mathrm{ENN} - \mathrm{GWO}$) model is established to autonomously evaluate the scaling area in reinforced concrete bridges. This model can be deployed by transportation agencies without domain knowledge in machine learning and meta-heuristics. Training Elman neural networks with meta-heuristic optimization algorithms is a powerful mechanism to improve the search engine of Elman recurrent neural networks through addressing the exploration– exploitation trade-off dilemma, which is expected to yield a significant enhancement in the prediction accuracy of scaling area. The proposed method utilizes grey wolf optimization algorithm for both parametric and structural learning, i.e., to automatically optimize the weights and define the best possible architecture of the Elman recurrent neural network. The

Elman neural network is trained by designing a variable-length single-objective optimization problem which encompasses a fitness function of minimization of the mean absolute percentage error of the scaling area. The steps of the grey wolf optimization algorithm are repeated until satisfying the convergence criteria, i.e., reaching maximum designated number of iterations. The optimized Elman neural network is saved and utilized to simulate the testing dataset.

The validation of the proposed scaling area evaluation model is conducted through three phases of comparisons. The first phase constitutes validating the employment of grey wolf optimization algorithm. This is achieved through comparing the proposed $\mathrm{ENN-GWO}$ model against hybrid Elman neural network-genetic algorithm ($\mathrm{ENN-GA}$) model, hybrid Elman neural network-particle swarm optimization algorithm ($\mathrm{ENN-PSO}$) model, hybrid Elman neural network-harmony search algorithm ($\mathrm{ENN-HS}$) model, hybrid Elman neural network-differential evolution algorithm ($\mathrm{ENN-DE}$) model and hybrid Elman neural network-shuffled frog-leaping algorithm ($\mathrm{ENN-SFL}$) model. The second phase involves comparing $\mathrm{ENN-GWO}$ model against seven state of art machine learning and deep learning models reported for their higher accuracies, namely back-propagation artificial neural network (ANN), Elman neural network, radial basis neural network (RBNN), generalized regression neural network (GRNN), convolutional neural network (CNN), support vector machines (SVM) and decision tree (DT). Evaluation of prediction models is one of the crucial issues in any data mining process, and it needs to be carried out based on multiple performance measures to create a comprehensive assessment of the prediction models.[65] Thus, the performances of the prediction models are assessed as per mean absolute percentage error (MAPE), root-mean squared error and mean absolute error (MAE). It is worth mentioning that the performances were assessed using split validation and 10-fold cross validation. The K-fold cross validation is used to ensure the training and testing of the entire dataset, which truncates the possibility of encountering over-fitting or over-learning in the scaling evaluation phase. The second phase aims at evaluating the robustness and stability of the scaling evaluation models using box plot analysis. The third phase is designated for the evaluation of the statistical significance levels of the outcome of prediction models using the performances of the different folds at a significance level of 0.05.

The third model is developed for the purpose of establishing a unified scaling severity index to evaluate scaling in reinforced concrete bridges based on its area and depth. It should be mentioned also that the proposed index can aid transportation agencies in prioritizing bridge decks for maintenance. The scaling area is interpreted from the previous model while the scaling depth is adopted from the third-order polynomial regression function developed by Dawood *et al.*[66]. The severity of scaling area is expressed in the form of percentage of the scaling area with respect to the whole zone area. Sufficient amount of record needs to be available for the sake of creating accurate severity rating systems of both area and depth. In this context, both scaling depth and area are simulated as random variables that follow certain probability distributions relying on the available dataset. Then, the best-fit probability distribution is determined according to Anderson Darling test goodness of fit test. Afterwards, Latin hypercube sampling is adopted to create numerous scenarios in order to establish the rating system for both scaling area and depth. Latin hypercube is stratified sampling scheme that offers better space coverage and scanning of the domain of the multi-dimensional design space. It is preferred over Monte Carlo sampling because it exhibits faster convergence rate within less number of samples and less variance.[67,68]

Fuzzy C-means clustering is adopted to compute the thresholds of the severity levels for both scaling area and depth. In this context, fuzzy C-means clustering is preferred over other clustering algorithms due to its ability to deal with inherent uncertainties encountered during the capturing and processing of the scaling images. Additionally, it produces more compact and better-separated clusters than hard K-means clustering algorithm.[69,70,71] The input images are then evaluated based on area and depth to determine the degree of scaling severity based on the obtained thresholds. The worst case scenario yielded from both scaling area and depth is selected and appended to create a more conservative model. Finally, the unified scaling severity index can be computed based on the weighted average computation of the different condition categories using Equation (4). The previous models are automated using a computerized platform that encompasses a hybridization of visual C#.net and Matlab programming languages. It is expected that the automated platform is capable of exploiting the compatibility and versatility capabilities of C#.net and the superior computational capacity of the Matlab.

$$\mathrm{USSI} = \frac{\sum_{c=1}^{4} Q_c \times W_c}{\sum_{c=1}^{4} Q_c} \tag{4}$$

Where;

Q_c represents the number of zones in condition category c. W_c represents the weighting factors for a bridge element in condition category c. The weighting factors for the "good", "medium", "severe", and "very severe" condition categories are assumed 100%, 70%, 50%, and 20%, respectively.

INSERT FIGURE 1

4. SCALING DETECTION MODEL

The proposed segmentation model $(\mathrm{MCE}-\mathrm{GWO})$ capitalizes on accommodation the minimum cross entropy approach and grey wolf optimization algorithm for the discrimination of scaling from background pixels. This section describes the basic theories of minimum cross entropy approach and grey wolf optimization algorithm.

4.1 Minimum cross entropy approach

Cross entropy is known as "Divergence", which is information metric that is used to measure the distance between two probability distributions. Assume $\mathrm{A}=\{\mathrm{A}_1, \mathrm{A}_2, \mathrm{A}_3, \mathrm{A}_4, \dots \dots .. \mathrm{A}_\mathrm{N}\}$ and $\mathrm{B}=\{\mathrm{B}, \mathrm{B}_2, \mathrm{B}_3, \mathrm{B}_4, \dots \dots .. \mathrm{B}_\mathrm{N}\}$, which represent two probability distributions. The cross entropy between A and B can be computed using Equation (5).[72,73] $\mathrm{D}(\mathrm{A}, \mathrm{B}) =$

$$\log \left(\sum_{i=1}^{N} \mathrm{A}_\mathrm{i} \left(\frac{\mathrm{A}_\mathrm{i}}{\mathrm{B}_\mathrm{i}}\right)\right. \tag{5}$$

Where;

$\mathrm{D}(\mathrm{A}, \mathrm{B})$ represents the cross entropy between the two probability distributions.

The minimum cross entropy thresholding algorithm is based on finding the optimum threshold T between the original image the segmented image. Assume an image I that contains L gray-levels $\{0, 1, 2, 3, \dots \dots .. \mathrm{L}-1\}$. Then the segmented image I_t obtained based on the threshold T can be defined using the following Equation.

$$\mathrm{I}_\mathrm{t}\,(\mathrm{x}, \mathrm{y}) = \begin{cases} \mu(0, \mathrm{T}-1), \text{if } \mathrm{I}(\mathrm{x}, \mathrm{y}) < \mathrm{T} \\ \mu(\mathrm{T}, \mathrm{L}-1), \text{if } \mathrm{I}(\mathrm{x}, \mathrm{y}) \geq \mathrm{T} \end{cases} \tag{6}$$

The normalized value of the cross entropy between the ranges c and d can be computed using Equation (7).

$$\mu(c,d) = \frac{\sum_{i=c}^{d-1} iH(i)}{\sum_{i=c}^{d-1} H(i)}, \qquad i = 0, 1, 2, 3, \dots \dots .. L-1 \tag{7}$$

The minimum cross entropy thresholding algorithm finds the optimum threshold by minimizing the cross entropy of the image (objective function) as shown in Equation (8).

$$D(T) = \min \left[\sum_{i=1}^{L} ih(i) \times \log(i) - \sum_{i=1}^{t-1} ih(i) \times \log\big(\mu(1,\ t)\big) - \sum_{i=t}^{L} ih(i) \times \log\big(\mu(t,\ L)\big) \right] \tag{8}$$

Since the first term is constant for a given digital image, the objective function can be re-formulated as follows.

$$D(T) = \min \left[-\sum_{i=1}^{t-1} ih(i) \times \log\big(\mu(1,\ t)\big) - \sum_{i=t}^{L} ih(i) \times \log\big(\mu(t,\ L)\big) \right] \tag{9}$$

Where;

$$D(T) = \min \left[-\sum_{i=0}^{T-1} i \times h(i) \times \log\left(\frac{\sum_{i=0}^{T-1} i \times h(i)}{\sum_{i=0}^{T-1} i \times h(i)}\right) - \sum_{i=T}^{L-1} i \times h(i) \times \log\left(\frac{\sum_{i=T}^{L-1} i \times h(i)}{\sum_{i=T}^{L-1} i \times h(i)}\right) \right] \tag{10}$$

4.2 Basic theory of grey wolf optimization algorithm

Grey wolf optimization algorithm is a recently-developed nature-inspired algorithm that was proposed by Mirjalili et al. in 2014.[40] The GWO algorithm is characterized by its capability to offer a proper trade-off between exploration and exploitation. This algorithm is based on simulation of the behavior of a pack of grey wolves, which follow distinct steps while hunting in nature. Each pack hierarchy consists of four levels of grey wolves which are: alpha (α), beta (β), delta (δ) and omega (ω). Alpha wolves are the leaders of the pack and the ones responsible for making decisions. The next level in the hierarchy is the beta grey wolves, whereas they act as the subordinates of the alpha grey wolves and they support them in the decision-making process. Delta grey wolves follow the dictated orders of both alpha and beta grey wolves but they dominate the omega grey wolves. Delta grey wolves can be scouts, hunters, elders, sentinels or caretakers. Omega grey wolves are the least prioritized wolves in the hierarchy, whereas they have to submit to all other dominant wolves. They play the role of scapegoat and they are the last ones allowed to eat.

The hunting mechanism of the GWO algorithm is discussed in the following sections[40,74]:

The social hierarchy of the GWO algorithm is as follows: a specific number of grey wolves in the pack explore the multi-dimensional search space to hunt a prey. The positions of the grey wolves are deemed as different position variables such that the distances of the prey from the grey wolves determine the fitness function values. Alpha is considered as the fittest solution while beta is the second best solution. Finally, delta is the third fittest solution. The individual grey wolf adjusts its position and moves towards a better position over the course of iterations in order to reach the prey with the shortest possible route.

Encircling prey is one of the main operators in the GWO algorithm, whereas grey wolves encircle prey during the hunting process. Equation (11) and Equation (12) are used to update the position of the grey wolf from the current location to a new location. The encircling behaviour can be mathematically expressed as follows.

$$\vec{D} = [\vec{C}.\overrightarrow{X_p}(t) - \vec{X}(t)] \tag{11}$$

$$\vec{X}(t+1) = [\overrightarrow{X_p}(t) - \vec{A}.\vec{D}] \tag{12}$$

Where;

t indicates the current iteration. $\overrightarrow{X_p}$ and $\vec{X}$ are the position vectors of the prey and grey wolf, respectively. $\vec{C}$ and $\vec{A}$ are the coefficient vectors and they can be expressed as follows.

$$\vec{A} = 2\vec{a}.\overrightarrow{r_1} - \vec{a} \tag{13}$$

$$\vec{C} = 2.\overrightarrow{r_2} \tag{14}$$

Where;

$\vec{a}$ is a motion vector linearly decreasing from 2 to 0 over the course of iterations in order to model approaching the prey. $\overrightarrow{r_1}$ and $\overrightarrow{r_2}$ are two random vectors within the interval [0, 1]. The vector $\vec{C}$ simulates the effect of the obstacles close to the prey in nature.

The hunting is a main operator in the GWO algorithm, whereas grey wolves are capable of determining the position of the prey and encircle it for hunting. Alpha grey wolf guides the pack during the hunting process. Beta and delta wolves might also participate in the hunting process. The alpha wolf which represents the best candidate solution in addition to the beta and delta wolves are assumed to have better knowledge about the position of the prey. Then, the best three positions obtained so far are appended and other search agents including the omega wolves are enforced to update their positions as per the positions of the best search agents. The mathematical formulation of the hunting behavior and positions of various categories of grey wolves can be expressed as follows.

$$\overrightarrow{D_\alpha} = |\overrightarrow{C_1}.\overrightarrow{X_\alpha} - \vec{X}| \tag{15}$$

$$\overrightarrow{D_\beta} = |\overrightarrow{C_2}.\overrightarrow{X_\beta} - \vec{X}| \tag{16}$$

$$\overrightarrow{D_\delta} = |\overrightarrow{C_3}.\overrightarrow{X_\delta} - \vec{X}| \tag{17}$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha} - \overrightarrow{A_1}.(\overrightarrow{D_\alpha}) \tag{18}$$

$$\overrightarrow{X_2} = \overrightarrow{X_\beta} - \overrightarrow{A_2}.(\overrightarrow{D_\beta}) \tag{19}$$

$$\overrightarrow{X_3} = \overrightarrow{X_\delta} - \overrightarrow{A_3}.(\overrightarrow{D_\delta}) \tag{20}$$

$$\vec{X}(t+1) = \frac{\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_3}}{3} \tag{21}$$

Where;

$\overrightarrow{X_\alpha}$, $\overrightarrow{X_\beta}$ and $\overrightarrow{X_\delta}$ represent the position vectors of the alpha, beta and delta fray wolves, respectively. The coefficient vectors $\overrightarrow{C_1}$, $\overrightarrow{C_2}$, $\overrightarrow{C_3}$, $\overrightarrow{A_1}$, $\overrightarrow{A_2}$ and $\overrightarrow{A_3}$ are computed using Equation (13) and Equation (14). $\vec{X}(t+1)$ represents the updated position of the grey wolf.

Attacking the prey is an important operator in the grey wolf optimization algorithm. The hunting process of the grey wolf optimization algorithm is terminated by attacking the prey when it stops moving. The fluctuation range of $\vec{A}$ is assumed to decrease by $\vec{a}$, which implies that $\vec{A}$ is a random value in the interval [-2a, 2a]. It was found that when $|\vec{A}|<1$, the grey wolves are forced to attack the prey which implies exploitation or local search of the grey wolf optimization algorithm. Search of prey is an essential operator in the hunting strategy of the GWO algorithm, whereas based on the positions of the alpha, beta and delta wolves, the grey wolves diverge from each other to search for the prey. It was noticed that when $|\vec{A}|>1$, the grey wolves diverge to search for a better prey in the search space which implies exploration or the global search of the grey wolf optimization algorithm. This avoids stagnation of the GWO algorithm in local solutions. It is worth mentioning that the coefficient vector in Equation (14) enables the GWO algorithm to behave more randomly and emphasize efficient exploration of the solution space by avoiding local optima. Finally, the GWO algorithm is terminated when the convergence criteria is satisfied.

5. SCALING EVALUATION MODEL

The scaling evaluation model is divided into two main sections, namely hybrid feature extraction model and Hybrid $\mathrm{ENN}-\mathrm{GWO}$ model for scaling area interpretation.

5.1 Hybrid feature extraction model

The framework of the proposed $\mathrm{SVD}-\mathrm{DWT}$ model for feature extraction is depicted in Figure 2. As shown in Figure 2, the spatial and frequency domain features are extracted using singular value decomposition and discrete wavelet transform, respectively. With respect to singular value decomposition, an input image of size 200×200 is reduced to a feature vector of size 1×200 that represents the singular values of the image. For discrete wavelet transform, the size of the input image is modelled using a sub-band energy vector of size 1×10 that is obtained from the wavelet decomposed scaling image. The two feature vectors are fused to create a resultant feature vector of size 1×210 to speed up the computational process by eliminating the insignificant features and appending the most dominant information in scaling images. An overview of the use of discrete wavelet transform in feature extraction of scaling images is presented in as follows.

INSERT FIGURE 2

Discrete wavelet transform is a multi-resolution representation that splits the input image into multiple frequency sub-bands that carry coarse approximation and detailed information of the image by convolving its rows and columns through a set of band pass filters.[75] The detailed information of the image is represented in three directions, namely vertical, horizontal and diagonal. In the present study, two dimensional discrete wavelet transform ($\mathrm{2D}-\mathrm{DWT}$) is applied to decompose the input scaling image, whereas a single level of decomposition using $\mathrm{2D}-\mathrm{DWT}$ results in four frequency sub-band images.[76] These frequency sub-bands are referred to as low-low (LL), low-high (LH), high-low (HL) and high-high (HH). In this regard, each of these frequency sub-bands denotes different characteristics of the image. The frequency sub-images can be derived through the implementation of scaling functions and wavelet functions as shown in Equations (22), (23), (24) and (25).[77]

$$\psi_{LH}(x, y) = \Phi(x)\psi(y) \tag{22}$$

$$\psi_{HL}(x, y) = \psi(x)\Phi(y) \tag{23}$$

$$\psi_{HH}(x, y) = \psi(x)\psi(y) \tag{24}$$

$$\Phi_{LL}(x, y) = \Phi(x)\Phi(y) \tag{25}$$

Where;

$\Phi(.)$ and $\psi(.)$ stand for scaling function (low-pass filter) and wavelet function (high-pass filter), respectively. $\Phi(x)$ and $\Phi(y)$ represent 1D scaling functions in row direction and column direction, respectively. $\psi(x)$ and $\psi(y)$ signify 1D wavelet functions in row direction and column direction, respectively. The high-pass filter extracts the high frequency components which contain detailed image's coefficients while the low pass filter induces the low frequency information which involve most of the image's energy (approximation coefficients). The sub-band image LH preserves the horizontal details of the image and it is obtained by implementing low-pass filter for rows and high-pass filter for columns. The sub-band image HL exhibits the vertical knowledge of the image and it is generated by passing the columns through low-pass filter and the rows by high-pass filter. The diagonal information of the image is stored in the sub-band image HH and it is produced by passing both rows and columns through low pass and high pass filters. LL is a coarse approximation of the image and it is created by passing both rows and columns through low pass filtering.

In each level of decomposition, the $2D - DWT$ generates the previously-mentioned four sub-band frequency images of LH, HL, HH and LL. In this context, at the primary level of decomposition, the four sub-band images are referred to as LH_1, HL_1, HH_1 and LL_1. The coefficient of the sub-image of LL_1 are used as an input for the subsequent second level of decomposition, whereas the sub-image is further decomposed into LH_2, HL_2, HH_2 and LL_2.In the third level of decomposition, the frequency sub-image $LL_2$is further divided into LH_3, HL_3, HH_3 and LL_3. The process continues until reaching the desired number of wavelet decomposition. At L levels of decomposition of $2D - DWT$, there are $(3 \times L)$+1 sub-bands. In the developed method, the scaling images undergo three levels of decomposition. This, results in obtaining ten frequency sub-band images.[78] Three levels of decomposition are found to provide an optimal trade-off between computational accuracy and computational efficiency.[79,80] Figure 3 illustrates a visualization of three level decomposition of the input scaling image using $2\mathrm{D}-\mathrm{DWT}$. As can be seen, the dimensions of the host scaling images are decreased by half during each level of decomposition. The frequency domain feature vector is composed of the energy values of the ten sub-images obtained from the three levels of decomposition. Thus, the size of the frequency domain feature vector is 1×10. The energy of each sub-band frequency image can be computed using Equation (26).[78,81]

INSERT FIGURE 3

$$\mathrm{ENE_s} = \frac{1}{\mathrm{W} \times \mathrm{H}} \sum_{\mathrm{i}=1}^{\mathrm{W}} \sum_{\mathrm{j}=1}^{\mathrm{H}} |\, \mathrm{P_s}(\mathrm{i}, \mathrm{j})| \qquad (26)$$

Where;

$\mathrm{ENE_s}$ stands for the energy of the $\mathrm{s}-\mathrm{th}$ sub-band image. $\mathrm{P_s}(\mathrm{i},\mathrm{j})$ denotes the pixel value of coordinates i and j in the $\mathrm{s}-\mathrm{th}$ sub-band image. W and H are the width and height of the sub-band image, respectively.

The developed method gives the user the flexibility to select between Haar, Daubechies 3, Symlet 2and Coiflet 5 wavelet transforms to build the frequency domain feature vector set. In the current case study, Haar wavelet transform is selected over other wavelets because it is one of the most important and frequently utilized wavelets that has been successfully implemented in several and diverse applications that demand high levels of accuracies. This encompasses its use in fingerprint recognition in document images[82], cloud detection from satellite images[83], monitoring driver fatigue[84] and image compression.[85] Haar wavelet is also characterized by being symmetric, orthogonal, and conceptually simple to compute and understand.[86,87] Furthermore, it is compactly supported, a computationally efficient wavelet and able to extract features well in the presence of noises.[88,89,90]

5.2 Hybrid ENN − GWO model for scaling area interpretation

The present study introduces a self-adaptive hybrid ENN − GWO model for the sake of automated evaluation of scaling present in images retrieved from reinforced concrete bridges. Elman neural network is a recurrent neural network that was proposed by Jeffrey Locke Elman in 1990. Elman neural network is known by its additional context layers, which helps in providing a memory about the results of the computations done so far in the network. The main distinct feature between the conventional feed-forward neural networks and recurrent neural networks is that in the case of RNNs, the output at each time step depends on memorizing previous inputs and computations while in the feed-forward neural network, outputs are independent of each other and the network output depends only on the current time step.[91,92]

In the proposed method, grey wolf optimization algorithm is applied to train the Elman neural network for the purpose of circumventing the slow learning rate, inferior accuracy of the gradient descent algorithm and the manual tuning of hyper parameters of Elman neural network. This is expected to achieve the global convergence accurately efficiently through amplifying the exploration and exploitation of the search space. The convergence of the gradient descent algorithm is highly sensitive to the initial setting of weight values, and this may yield significant different performances from one setting to the other. Additionally, starting with incorrect values results in slow learning progress. As such, the training process based on the gradient descent algorithm usually gets the network to converge to local minima. The convergence of the network before global minima often hinders its exploration of the whole search space and training set, which results in inferior network performance.[26,27]

With regards to the manual tuning of the hyper parameters of Elman neural network, Elman neural networks are characterized by the presence of large number of hyper parameters such as number of context layers, number of hidden layers, number of context neurons, number of hidden neurons and type of activation transfer function. Manual tuning is based on trial and error iterations to come up with the optimum configuration of the hyper parameters. In this context, manual tuning of these hyper parameters is error-prone, tedious, and highly reliable on engineers' expertise and understanding of the features of the underlying problem. Moreover, it is incapable of capturing non-linear hyper parameters' interactions and it is impractical in complex large-scale problems.[93,94]

In view of the above, the developed method relies on grey wolf optimization algorithm for the simultaneous tuning of parameters and hyper parameters of Elman neural network. With respect to the parametric learning, the developed method aims at optimizing the weight values of the connections between neurons. At the level of structural learning, the developed method automatically optimizes number of context layers, number of context neurons, number of hidden layers, number of hidden neurons and type of transfer function. In this regard, the developed method explores the implementation of eight different types of activation transfer function. This encompasses triangular basis transfer function, normalized radial basis transfer function, radial basis transfer function, positive linear transfer function, linear transfer function, Elliot symmetric sigmoid transfer function, log-sigmoid transfer function and hyperbolic tangent sigmoid transfer function. As a result of the hyper parameter optimization feature of the developed method, the number of weighted connection changes during each training iteration according to the numbers of context layers, hidden layers, context neurons and hidden neurons. Therefore, an estimator is created for the automated computation of the number of weighted connections according to the explored hyper parameters in this iteration. The designed estimator can be mathematically expressed using Equation (27).

$$\begin{aligned}\mathrm{Num\_W} = &\big((\mathrm{IN}+1)\times \mathrm{HN}\big) + ((\mathrm{HN}\times \mathrm{CN}\times \mathrm{HL} + \big((\mathrm{HN}+1)\times \mathrm{HN}\times(\mathrm{HL}-1)\big) \\ &+ \big((\mathrm{HN}+1)\times \mathrm{ON}\big)\end{aligned} \tag{27}$$

Where;

$\mathrm{Num\_W}$ is the number of weighted connections between neurons. IN and ON represent numbers of input neurons and output neurons, respectively. HN and CN denote numbers of hidden neurons and context neurons, respectively. HL stands for number of hidden layers. In this regard, the number of hidden layers is assumed to be equal to the number of context layers. It is worth mentioning that the developed automated platform enables to set the maximum numbers of hyper parameters according to the desired user's preference.

The automated parameter and hyper parameter optimization is carried out based on designing a single-objective optimization function that minimizes the mean absolute percentage error of scaling area. In this context, the automated calibration of Elman neural network is triggered by the spatial domain and frequency domain features of the input scaling images. Mean absolute percentage error is chosen as training function because it is a widely-utilized performance indicator for the evaluation of prediction models. Furthermore, it is more robust and practical performance metric.[95,96] The mean absolute percentage error of scaling area can be defined using Equation (28).

$$\mathrm{MAPE} = \frac{100}{\mathrm{N}} \times \sum_{\mathrm{i}=1}^{\mathrm{K}} \frac{|\mathrm{P_i} - \mathrm{A_i}|}{\mathrm{A_i}} \tag{28}$$

Where;

K denotes number of input images. $\mathrm{A_i}$ and $\mathrm{P_i}$ stand for actual and predicted scaling areas of $\mathrm{i}-\mathrm{th}$ image , respectively.

6. METHOD IMPLEMENTATION

The images utilized to train and test the proposed scaling detection and evaluation method are captured from three bridge decks in Montreal and Laval, Canada using Sony DSC-H300 digital camera of 20.1 megapixel resolution. The dimensions of the captured images are of size 5152×3864, and the resolution of the image is 350 ppi (pixels per inch). The dataset is comprised of 60 images such that 50 images were used for training and the remaining 10 images were utilized for testing purpose. All the computations of the machine learning and optimization algorithms are carried out on a laptop with an Intel Core i7 CPU, 2.2 GHz and 16 GB of memory. The images are standardized to 200×200 to speed up the computation process and enhance the learning capacity of the developed machine learning model. Sample of the scaling images is shown in Figure 4. Different positions and orientations of scaling are considered for the sake of investigating the robustness of the proposed $\mathrm{MCE}-\mathrm{GWO}$ model. Wiener filter of size 3×3 is applied to restore the images by removing noises present in images. Then, the min-max gray level discrimination approach is applied for the purpose of contrast enhancement of scaling.

INSERT FIGURE 4

The proposed scaling segmentation model relies on the integration of minimum cross entropy approach and grey wolf optimization algorithm for the purpose of segmentation of scaling in reinforced concrete bridges. The cross entropy is the objective function that GWO algorithm seeks its minimization to explore the search space for the optimum threshold. The number of iterations and search agents of the GWO algorithm are assumed 10 and 40, respectively. The convergence curves of the proposed $\mathrm{MCE-GWO}$ model for image “A”, image “B”, image “C”, image “D” and image “E” are presented in Figures 5, 6 and 7. As can be seen, the cross entropy function stabilizes after iteration 14, 3, 16, 3 and 11 for image “A”, image “B”, image “C”, image “D” and image “E”, respectively. The optimum threshold values obtained using the $\mathrm{MCE-GWO}$ model for image “A”, image “B”, image “C”, image “D” and image “E” are 153, 111, 118, 153 and 127, respectively. This manifests the superior capability of the grey wolf optimization algorithm in exploring the histogram-based search space to find the optimum threshold. The segmented images using the proposed $\mathrm{MCE-GWO}$ model for images “A”, “B”, “C”, “D” and “E” are depicted in Figure 8. Any pixel that has a value more than the optimum threshold, it is appended as a scaling (blue mask). Otherwise, it is considered as a non-distress pixel in the background. These images provide a visual understanding and evaluation of the qualities of the proposed segmentation model. As shown in the Figure 8, the proposed $\mathrm{MCE-GWO}$ successfully recognized the scaling in the images, such that the scaling pixels are very well-discriminated from the background.

INSERT FIGURE 5

INSERT FIGURE 6

INSERT FIGURE 7

INSERT FIGURE 8

The proposed $\mathrm{MCE-GWO}$ is compared against $\mathrm{MCE-GA}$, $\mathrm{MCE-PSO}$, $\mathrm{MCE-HS}$, $\mathrm{MCE-DE}$ and $\mathrm{MCE-SFL}$ to test the performance of the grey wolf optimization algorithm. Different initializations of parameters of the meta-heuristics were experimented in order to search for their optimum setting Each meta-heuristic was run ten times independently in order to avoid unstable solutions due to random initialization of population. In order to establish a fair comparison between the different meta-heuristic optimization algorithms, the population size and number of iterations are assumed 10 and 40, respectively. For the genetic algorithm, tournament selection is the parent selection strategy. Two-point crossover is utilized, and the crossover rate is assumed 0.8. Mutation rate is assumed 0.1. With respect to the particle swarm optimization algorithm, the cognitive learning and social parameters are assumed two, and the inertia weight is assumed 0.5. With respect to the harmony search algorithm, the harmony search consideration rate and the pitch adjustment rate are assumed 0.9 and 0.1, respectively. For the differential evolution algorithm, the crossover probability is assumed 0.2, and the mutation is assumed to follow a uniform distribution between 0.2 and 0.8. For the shuffled frog-leaping algorithm, the number of memeplexes is assumed 2 and there are 5 frogs per each memeplex.

The convergence curves of the meta-heuristic-based scaling segmentation models of image "C" are depicted in Figure 9. It can be observed that the proposed $\mathrm{MCE}-\mathrm{GWO}$ model achieved the lowest cross entropy followed by $\mathrm{MCE}-\mathrm{DE}$. On the other hand, $\mathrm{MCE}-\mathrm{HS}$ provided the least performance among the meta-heuristic-based segmentation models for the detection of scaling in image "C". The least minimum cross entropy achieved by $\mathrm{MCE}-\mathrm{GWO}$, $\mathrm{MCE}-\mathrm{DE}$ and $\mathrm{MCE}-\mathrm{PSO}$ are 59379.5593, 59382.1686 and 59475.5606, respectively. As such, it can be concluded that the proposed $\mathrm{MCE}-\mathrm{GWO}$ model outperformed other meta-heuristic-based segmentation models in the detection of scaling in image "C". A quantitative comparative analysis between the different meta-heuristic-based segmentation models for scaling detection of the fifty images is shown in Table 1. The performances of the meta-heuristic-based scaling segmentation models are evaluated capitalizing on average mean-squared error (AMSE), average mean absolute error (AMAE), average peak signal to noise ratio (APSNR) and average cross entropy (ACE). The cross entropy represents the fitness function of the meta-heuristic-based scaling segmentation models.

INSERT FIGURE 9

As shown in Table 1, $\mathrm{MCE}-\mathrm{GWO}$ performed better than other meta-heuristic-based scaling segmentation models achieving AMSE, AMAE, APSNR and ACE of 0.175, 0.407, 55.754 and 26011.019, respectively. Nevertheless, $\mathrm{MCE}-\mathrm{GA}$ showed the lowest segmentation performance attaining AMSE, AMAE, APSNR and ACE of 0.191, 0.43, 55.364 and 27309.737, respectively. It can be also inferred that $\mathrm{MCE}-\mathrm{HS}$ and $\mathrm{MCE}-\mathrm{PSO}$ accomplished an acceptable segmentation performance. In this regard, $\mathrm{MCE}-\mathrm{HS}$ achieved AMSE, AMAE, APSNR and ACE of 0.181, 0.42, 55.555 and 30353.82, respectively. Moreover, $\mathrm{MCE}-\mathrm{PSO}$ generated AMSE, AMAE, APSNR and ACE of 0.187, 0.426, 55.426 and 26120.03, respectively . In the light of forging, it can be derived that $\mathrm{MCE}-\mathrm{GWO}$ achieved the best segmentation performance in terms of the four performance indicators.

INSERT TABLE 1

The proposed $\mathrm{MCE-GWO}$ is further validated through its comparison against some of the state of art well-performing image segmentation models, namely Otsu, K-means clustering, fuzzy C-means clustering and expectation maximization. The segmented images using Otsu, K-means clustering, fuzzy C-means clustering and expectation maximization for images "A", "B" and "C" are shown in Figures 10, 11 and 12. It can be projected that the classical segmentation models failed to establish clear and well-separated scaling pixels. Their segmented images encompass noises and unwanted pixels in the foreground elicited form the failure of the classical segmentation models to search for the optimum thresholds. Furthermore, it can be concluded that the proposed $\mathrm{MCE-GWO}$ model yielded a consistent superior segmentation capacity demonstrated in the form of homogenous and well-separated histogram corresponding to each class of the image, i.e., scaling and background. Table 2 provides a comprehensive performance comparison of the proposed scaling segmentation models against the conventional models. As shown in Table 2, the proposed $\mathrm{MCE-GWO}$ model provided better segmentation capacity when compared against other models in the literature. K-means clustering generated the second highest performance achieving AMSE, AMAE, APSNR and ACE of 0.232, 0.471, 54.495 and 35876.343, respectively. It can be also derived that Otsu generated the lowest segmentation performance achieving AMSE, AMAE, APSNR and ACE of 0.273, 0.512, 53.813 and 40621.781, respectively.

INSERT FIGURE 10

INSERT FIGURE 11

INSERT FIGURE 12

INSERT TABLE 2

Figures 13 and 14 provide a visualization of the meta-heuristic-based segmentation models and classical segmentation models based on their average mean squared error, average mean absolute error, average peak signal to noise ratio and average cross entropy. It can be noticed that meta-heuristic-based segmentation models obtained lower AMSE, AMAE, ACE and higher APSNR than classical segmentation models. At the level of meta-heuristic-based segmentation models, the developed $\mathrm{MCE-GWO}$ model accomplished the lowest AMSE, AMAE, ACE and the highest APSNR. $\mathrm{MCE-GA}$ attained the highest AMSE, AMAE and the lowest APSNR. Additionally,

$\mathrm{MCE-HS}$ yielded the highest ACE. With regards to the classical segmentation models, the smallest values of AMSE, AMAE, ACE and the highest value of APSNR were obtained by K-means clustering. Furthermore, Otsu and fuzzy c-means clustering algorithms failed to detect scaling appropriately such that Otsu provided the highest AMSE, AMAE and the lowest APSNR. Also, fuzzy c-means clustering algorithm exhibited the highest value of ACE.

INSERT FIGURE 13

INSERT FIGURE 14

A third comparative analysis is carried out to evaluate the significance levels of the output of the scaling segmentation models. Shapiro-Wilk test is applied to study the normality of the data at significance level of 0.05. It examines the null hypothesis ($\mathrm{H_0}$), which implies that the random variable follows a normal distribution. On the other hand, the alternative hypothesis ($\mathrm{H_1}$) assumes that the random variable doesn't follow a normal distribution. Hence, if the $\mathrm{P-value}$ is less than the significance level, then the outcome of the segmentation models don't follow normal distribution. Nonetheless, if the $\mathrm{P-value}$ is more than the significance level, then the outcome of the segmentation models follow normal distribution. Table 3 describes the $\mathrm{P-values}$ of the mean absolute error and peak signal to noise ratio. As shown in Table 3, all the $\mathrm{P-values}$ are more than 0.05, which imply that the null hypothesis is accepted and therefore the performance indicators of the scaling segmentation models follow normal distributions.

In the view of the above, a parametric student's t-test is applied to examine the statistical significance levels of the outcome of the scaling segmentation models at significance level of 0.05. The performed student's t-tests examine the null hypothesis ($\mathrm{H_0}$), which is that there is no significant difference between the segmentation capacities of the scaling detection models. On the other hand, the alternative hypothesis ($\mathrm{H_1}$) assumes that there is a significant difference between the segmentation capacities of the scaling detection models. Tables 4 and 5 describe the student's t-test based on MAE and PSNR, respectively. As can be seen, the $\mathrm{P-values}$ of the proposed $\mathrm{MCE-GWO}$ model against other models are less than 0.05. For instance, the $\mathrm{P-value}$ of the pair ($\mathrm{MCE-GWO}$, Otsu) based on is 0 while the $\mathrm{P-value}$ of the pair ($\mathrm{MCE-GWO}$, K-means clustering) is 1×10^{-5}. This implies that the null hypothesis is rejected. Thus, there is significant difference between the proposed scaling segmentation model and other models. In the light of forgoing analysis, it can be derived that the developed $\mathrm{MCE} - \mathrm{GWO}$ model successfully investigated the solution space of the multimodal histogram of scaling images globally and it was able to find the optimum threshold values that distinguished scaling from the background, whereas the developed model exemplified comprehensive and significant superior segmentation capabilities against other scaling detection models. This results from the higher exploration and exploitation abilities of the integration of cross entropy function grey wolf optimization algorithm, such that other meta-heuristics were not able to maintain same level of segmentation accuracies due to local minima trapping effect. Also, conventional segmentation models failed to deal with the uneven illumination and low contrast nature of scaling images, and thus they were unable to find the optimum threshold.

INSERT TABLE 3

INSERT TABLE 4

INSERT TABLE 5

The second model is the hybrid $\mathrm{ENN} - \mathrm{GWO}$ which is designed for the purpose of evaluating scaling area in reinforced concrete bridges. The proposed $\mathrm{SVD} - \mathrm{DWT}$ model is formulated to generate the feature vector set by mapping the most dominant features and information present in images. The interface of the feature extraction module in the automated platform is shown in Figure 15. In this regard, the user is asked to identify the type of wavelet function, which is selected to be Haar function for the case in-hand. By clicking "View" button, the computerized platform computes the spatial domain features and frequency domain features of the images relying on the singular values and the energies of all Haar discrete wavelet transform sub-bands.

INSERT FIGURE 15

The performance capacity of the Elman neural network is heavily influenced by its setting hyper-parameters which include: number of hidden layers, number of context layers, number of hidden neurons, number of context neurons, type of transfer functions and weights of the connections between neurons. Thus, the present study adopts the grey wolf optimization algorithm to autonomously optimize the parameters and topography of the Elman neural network. In the automated platform, the user is asked to specify the optimization parameters of the proposed self-adaptive scaling assessment. The interface of the developed scaling evaluation model is presented in Figure 16. As can be seen, the maximum number of hidden and context layers are eight. Also, the maximum number of hidden and context neurons are eight. Eight transfer functions are investigated and the values of weights are real numbers between -1 and 1. Therefore, the maximum length of the decision variables is 2716, which is regarded as an exhaustive search space that substantiates the employment of extensive training mechanism. For the parameters of the grey wolf optimization algorithm, number of search agents and number of iterations are assumed 100 and 200, respectively. The output of this model is obtained by pressing the "View" button. This constitutes the maximum length of the variable-length optimization model, minimum mean absolute percentage error, and optimum parameters and configuration of the Elman neural network. The lowest MAPE achieved by the $\mathrm{ENN-GWO}$ model is 0.7144%. Moreover, the optimum numbers of hidden and context layers are three while the optimum numbers of hidden and context neurons are eight. The optimum transfer function is the hyperbolic tangent sigmoid function. The optimum transfer function is the hyperbolic tangent sigmoid function.

INSERT FIGURE 16

A three-fold comparison is carried out for the validation of the proposed $\mathrm{ENN-GWO}$ model. The first fold involves its comparison against a set of meta-heuristic-based Elman neural network models, namely $\mathrm{ENN-GA}$, $\mathrm{ENN-PSO}$, $\mathrm{ENN-HS}$, $\mathrm{ENN-DE}$, and $\mathrm{ENN-SFL}$. The MAPE is the fitness function adopted to train the Elman neural network models. The convergence curves of the meta-heuristic-based Elman neural network models are depicted in Figure 17. It can be derived that $\mathrm{ENN-GWO}$ achieved the lowest MAPE followed by $\mathrm{ENN-PSO}$ while $\mathrm{ENN-GA}$ generated the highest MAPE among the meta-heuristic-based Elman neural network models. The lowest MAPE obtained by $\mathrm{ENN-GA}$, $\mathrm{ENN-PSO}$, $\mathrm{ENN-HS}$, $\mathrm{ENN-DE}$, and $\mathrm{ENN-SFL}$ are 8.2207%, 1.9253%, 6.9765%, 4.2292% and 4.663%, respectively. This demonstrates that the proposed $\mathrm{ENN-GWO}$ model accomplished the lowest training error among the other meta-heuristic-based Elman neural network models.

INSERT FIGURE 17

The performances of the thirteen prediction models as per split validation and 10-fold cross validation are shown in Tables 6 and 7, respectively. The mean absolute error and root-mean squared error are measured in terms of cm^2. As shown in Tables 6 and 7, the hybrid meta-heuristic-based Elman neural network models outperformed the well-performing state of art machine learning models. It can be derived that the proposed $\mathrm{ENN}-\mathrm{GWO}$ model outperformed other machine learning models as per split validation and 10-fold cross validation. $\mathrm{ENN}-\mathrm{PSO}$ obtained the second lowest prediction error. On the other hand, $\mathrm{ENN}-\mathrm{GA}$ model obtained the least prediction performance among the meta-heuristic-based Elman neural network models. This state of affairs substantiates the use of grey wolf optimization algorithm in exploring the relationships between the input images and scaling area. With respect to the conventional machine learning models, ENN generated the least prediction error while ANN model obtained the highest prediction error. For instance, as per the cross-validation model, the proposed $\mathrm{ENN}-\mathrm{GWO}$ model achieved MAPE, RMSE and MAE of 1.513%, 29.836 and 12.066, respectively. Nevertheless, MAPE, RMSE and MAE of ANN model are 23.306%, 232.823 and 194.135, respectively.

INSERT TABLE 6

INSERT TABLE 7

An illustration of the obtained MAPE, RMSE and MAE by the meta-heuristic-based Elman neural network models and the conventional prediction models based on split validation are depicted in Figures 18, 19 and 20. In the grand scheme of things, the meta-heuristic-based Elman neural network models performed better than the conventional prediction models in terms of MAPE, RMSE and MAE. At the level of the meta-heuristic-based Elman neural network models, it is found that the developed $\mathrm{ENN}-\mathrm{GWO}$ accomplished considerable lower MAPE, RMSE and MAE than the remainder of the meta-heuristic-based Elman neural network models. Furthermore, $\mathrm{ENN}-\mathrm{PSO}$ and $\mathrm{ENN}-\mathrm{SFL}$ attained satisfactory MAPE, RMSE and MAE. It is also observed that $\mathrm{ENN}-\mathrm{GA}$ had the highest MAPE and RMSE while $\mathrm{ENN}-\mathrm{HS}$ provided the highest MAE. At the level of conventional prediction models, it can be derived that they were unable to accurately interpret scaling area. In this regard, ENN had the lowest MAPE and RMSE while SVM generated the lowest MAE, On the contrary, ANN generated the highest MAPE and MAE while CNN yielded the highest RMSE.

INSERT FIGURE 18

INSERT FIGURE 19

INSERT FIGURE 20

In order to synthesize the performances of the meta-heuristic-based Elman neural network models, the box plot of the mean absolute percentage error is presented in Figure 21. The box plots facilitate analysing the robustness of the different meta-heuristics through mapping the distribution and skewness of the numerical data. It displays the minimum, first quartile, third quartile and maximum values of the multiple runs. The solid line in the box encodes the second quartile or the median value. The height of the box (space between the first and third quartiles) delineates the robustness of the algorithm, which is regarded as one of the main aspects to evaluate their performance. Lower spread in the box plot signifies more robustness performance of the model. Figure 21 demonstrates that the developed $\mathrm{ENN-GWO}$ model exhibited more stable and consistent results compared to the reminder meta-heuristic-based Elman neural network models, On the contrary, $\mathrm{ENN-PSO}$ provided unstable results, such that it experiences large perturbations in the different runs. It can be also projected that $\mathrm{ENN-GWO}$ model sustains the lowest mean absolute percentage with respect to other models over the course of the different runs.

INSERT FIGURE 21

A further comparison is carried out to investigate the significance levels of the prediction capabilities of the machine learning models. Shapiro-Wilk test is deployed to study the normality of the mean absolute percentage generated from the different folds of the machine learning models (see Table 8). As presented in Table 8, the $\mathrm{P-values}$ of the output variable (MAPE) is less than 0.05, which implicates that the MAPE doesn't follow normal distribution. In this context, non-parametric tests are employed to evaluate the statistical significant levels of the performances of the machine learning models. These tests include Wilcoxn test, Mann-Whitney-U test, Kruskal–Wallis test, binomial sign test and Mood's median test. The non-parametric tests of the machine learning models are shown in Table 9. Results indicate that the $\mathrm{P-values}$ of the $\mathrm{ENN-GWO}$ model against other models are less than 0.05 for all tests. For example, the $\mathrm{P-values}$ of the pairs $(\mathrm{ENN-GWO, ENN-GA})$ and $(\mathrm{ENN-GWO, ENN})$ are 8.5×10^{-4} and 4.12×10^{-2}, respectively. This reveals that the proposed $\mathrm{ENN-GWO}$ model significantly outperformed the meta-heuristic-based Elman neural network models and state of prediction models.

INSERT TABLE 8

INSERT TABLE 9

In view of the above multi-layered comparative analysis, it can be concluded that the developed $\mathrm{ENN}-\mathrm{GWO}$ model exhibited significant better and more robust performance than other meta-heuristic-based Elman neural network models and conventional prediction models. In this regard, the developed $\mathrm{SVD}-\mathrm{DWT}$ managed to extract the most essential underlying features in the scaling images. Furthermore, the developed hybrid $\mathrm{ENN}-\mathrm{GWO}$ model sustained adequate trade-off between exploration and exploitation search abilities which allowed it to efficiently sweep the entire design space of parameters and hyper parameters of Elman neural network globally while avoiding being trapped in local minima and premature convergence. It was also noticed that the remainder of the meta-heuristic-based Elman neural network models provided higher prediction error than the developed $\mathrm{ENN}-\mathrm{GWO}$ model as a result of the low exploration capabilities of GA, PSO algorithm, HS algorithm, DE algorithm and SFL algorithm. Conventional prediction models did not perform well also because of the absence of the automated hyper parameter optimization and their low training efficiency in scaling area prediction.

The third model focuses on establishing a unified scaling severity index. A sample of the cluster memberships of scaling obtained from the FCM algorithm is shown in Table 10. In the fuzzy C-means clustering algorithm, the data point is assigned to the cluster that has the maximum degree of membership. For instance, the data point of scaling area 71.013 is assigned to "Cluster 1" because it has the maximum degree of membership of 0.557. Furthermore, the data point of scaling area 1059.227 is assigned to "Cluster 3" since it is accompanied with the maximum degree of membership of 0.81. The thresholds used to describe the severity levels of scaling area and depth are shown in Table 11. It can be interpreted that if the scaling area is between 40% and 50% of the zone area, this implies that the bridge deck is in a "Poor" Condition from an area perspective. Furthermore, if the scaling depth is more than 4 mm, this means that the bridge deck is in a "Very Poor" condition from a depth perspective. As such, the percentages of the "Good", "Medium", "Poor", and "Very Poor" categories are: 10%, 75%, 15%, and 0%, respectively. The unified scaling severity index based on Equation (4) is 73.75%, which indicates that the bridge deck is in a "Medium" category based on scaling. In the view of the afore-conducted comparative analysis, it is expected that the developed method can provide an efficient decision-making platform that aids transportation agencies in evaluating scaling in reinforced concrete bridges.

INSERT TABLE 10

INSERT TABLE 11

7. Conclusion

Routine inspections are diagnostic methods that are executed on equal time intervals to monitor the deterioration of reinforced concrete bridges. Nevertheless, visual inspection-based models are biased resulting from being subjective, labour-intensive, time-consuming and hazardous in some circumstances. This state of affairs adversary affect the quality of the decision-making process exemplified in the form of error-prone condition assessment models and inefficient maintenance prioritization plans at the various managerial levels. In this regard, the present study proposes a novel three-tier platform for the automated detection and evaluation of scaling in reinforced concrete bridges. The first model $(\mathrm{MCE}-\mathrm{GWO})$ is envisioned on the accommodation of the minimum cross entropy approach coupled with grey wolf optimization algorithm the segmentation of scaling. In this regard, Wiener filter and, min-max gray level discrimination approach are applied for the purpose of restoration of the degraded images noise and contrast enhancement.

The second model is a newly-developed $\mathrm{ENN}-\mathrm{GWO}$ hybridization for the automated evaluation of scaling area present in images. In this context, a variable-length grey wolf optimization model is formulated for both parametric and structural learning of Elman neural network that amplified the exploration and exploitation of its training mechanism. The second model comprises a hybrid $\mathrm{SVD}-\mathrm{DWT}$ model to create the feature vector set by mapping the most dominant features in images. In this model, singular value decomposition and discrete wavelet transform are utilized to capture the spatial domain features and frequency domain features, respectively. The third model is designated for establishing a unified scaling severity index to evaluate scaling in reinforced concrete bridges capitalizing on its area and depth. In it, Anderson Darling test is employed to identify the best-fit distribution of scaling area and depth. Additionally, fuzzy C-means clustering is applied to establish their severity levels. A computerized platform is designed to facilitate the implementation of the developed method by the users.

The first model is validated through both performance evaluation and statistical significance comparison against image segmentation models reported for their higher accuracies. It was inferred that the developed scaling segmentation model significantly outperformed the afore-mentioned models such that it achieved AMSE, AMAE, APSNR and CE of 0.175, 0.407, 55.754 and 26011.019, respectively. In this regard, it was found that the developed $\mathrm{MCE}-\mathrm{GWO}$ model managed to improve the scaling detection accuracies by 24.01%, 10.25% and 8.1% when compared against the widely-utilized Otsu, K-means clustering and fuzzy c-means clustering algorithms, respectively. This evinces that the developed $\mathrm{MCE}-\mathrm{GWO}$ model successfully investigated the solution space-based histogram of thresholds globally while conventional segmentation models failed to deal with the uneven illumination and low contrast nature of scaling images.

With regards to the developed scaling evaluation model, it was projected that the developed model achieved notable superior and more robust prediction accuracies than a set of meta-heuristic-based Elman neural network models and state of art conventional prediction models as per split validation and ten-fold cross validation. In this context, the developed $\mathrm{ENN}-\mathrm{GWO}$ model obtained MAPE, RMSE and MAE of 1.513%, 29.836 and 12.066, respectively as per ten-fold cross validation. As such, the developed $\mathrm{ENN}-\mathrm{GWO}$ model was able to accomplish a reduction in the prediction error of scaling area by 81.38% and 45.65% with respect to $\mathrm{ENN}-\mathrm{GA}$ and $\mathrm{ENN}-\mathrm{PSO}$, respectively. Furthermore, it obtained results that are 91.23% and 89.34% better than highly acknowledged artificial neural network and support vector machines, respectively. This demonstrates the higher efficiently of grey wolf optimizer in exploring the entire design space of parameters and hyper parameters without getting trapped in local minima. The third model interpreted that the severity levels of the scaling area are 30%, 40% and 50% of the zone area. Furthermore, it clarified that the USSI of the bridge deck is 73.75%, which implies that the severity level of scaling of the bridge deck is in a “Medium” category.

In the light of forgoing, it can be argued that the developed automated method performed competently in detection and evaluation of scaling in reinforced concrete bridges, which enables it to establish a reliable decision-making paradigm for assessing and prioritizing bridge decks based on the extent of severities of scaling. This in return can pave the way for less labor intensive, more accurate and cost-efficient inspection process of reinforced concrete bridges. However, the developed method has some limitations. It could misclassify water marks, expansion joints, dirt, oil stains and as scaling pixels. Thus, further processing needs to be conducted in order to remove these noises from the input images. Another limitation in the current developed method is the long processing time resulting from the high computational effort required for optimizing the architecture of the Elman neural network. Future research directions include exploring of some remote sensing technologies such as unmanned aerial vehicles equipped with light detection and ranging (LIDAR) sensors and global positioning systems (GPS) for evaluating scaling severities of the different zones of the bridge and comparing it against image-based models. This also comprises studying the possibility of fusing the heterogeneous data of LIDAR and digital camera for more accurate interpretation of scaling severities. Future work may also include developing a mobile application that is able to provide practical and real time analysis of scaling images.

Conflict of interest: The authors declare that they have no conflict of interest.

REFERENCES

1. A. Farzam, M.-J. Nollet and A. Khaled, Integration of site conditions information using geographic information system for the seismic evaluation of bridges, *Canadian Society of Civil Engineering Annual Conference: Resilient Infrastructure*, 1-4 June 2016, London, Canada, pp. 1-10.
2. G. Felio, *Canadian Infrastructure Report Card* (Canadian Construction Association, Canadian Public Works Association, Canadian Society for Civil Engineering, and Federation of Canadian Municipalities, Canada, 2016).
3. Viami International Inc. and the Technology Strategies Group, *Market Study for Aluminium Use in Roadway Bridges* (Aluminium Association in Canda, Montreal, Canada, 2013).
4. Statistics Canada, Age of Public Infrastructure: A Provincial Perspective (2009), http://www.statcan.gc.ca/pub/11-621-m/11-621-m2008067-eng.htm.

5. I. H. Kim, H. Jeon, S. C. Baek, W. H. Hong and H. J. Jung, Application of crack identification techniques for an aging concrete bridge inspection using an unmanned aerial vehicle, *Sensors*, **18**(6) (2018) 1–14.
6. B. Lei, N. Wang, P. Xu and G. Song, New Crack Detection Method for Bridge Inspection Using UAV Incorporating Image Processing, *Journal of Aerospace Engineering*, **31**(5) (2018) 1–13.
7. R. S. Lim, H. M. La and W. Sheng, A robotic crack inspection and mapping system for bridge deck maintenance, *IEEE Transactions on Automation Science and Engineering*, **11**(2) (2014) 367–378.
8. Y. Dang, N. Xie, A. Kessel, E. McVey, A. Pace and X. Shi, Accelerated laboratory evaluation of surface treatments for protecting concrete bridge decks from salt scaling, *Construction and Building Materials*, **55** (2014) 128–135.
9. X. Shi, M. Akin, T. Pan, L. Fay, Y. Liu and Z. Yang, Deicer Impacts on Pavement Materials: Introduction and Recent Developments, *The Open Civil Engineering Journal*, **3**(1) (2009) 16–27.
10. Y. Wang, J. Y. Zhang, J. X. Liu, Y. Zhang, Z. P. Chen, C. G. Li, K. He and R. B. Yan, Research on Crack Detection Algorithm of the Concrete Bridge Based on Image Processing. *Procedia Computer Science*, **154** (2019) 610–616.
11. X. Yu, X. Wang, X. Da and J. Zhao, Crack Detection Algorithm of Complex Bridge Based on Image Process, *20th COTA International Conference of Transportation*, 14-16 August 2020, Xi’an, China, pp. 3144–3155.
12. X. Zhang and X. Wang, An Effective Bridge Cracks Classification Method Based on Machine Learning, *Proceedings of the 2020 4th International Conference on Electronic Information Technology and Computer Engineering*, 6-8 November 2020, Xiamen, China, pp.790–794.
13. Y. Noh, D. Koo, Y. M. Kang, D. G. Park and D. H. Lee, Automatic crack detection on concrete images using segmentation via fuzzy C-means clustering, *2017 International Conference on Applied System Innovation (ICASI)*, 13-17 May 2017, Sapporo, Japan, pp. 877–880.
14. H. Xu, X. Su, Y. Wang, H. Cai, K. Cui and X. Chen, Automatic Bridge Crack Detection Using a Convolutional Neural Network, *Applied Sciences*, **9**(14) (2019) 1-14.

15. G. Li, Q. Liu, S. Zhao, W. Qiao and X. Ren, Automatic crack recognition for concrete bridges using a fully convolutional neural network and naive Bayes data fusion based on a visual detection system, *Measurement Science and Technology*, **31**(7) (2020), 1-17.
16. W. Yan, Y. Junjie, M. Junteng, C. Yong and W. Kai, Automatic detection method of bridge cracks based on residual network, *2020 6th International Conference on Hydraulic and Civil Engineering*, 11-13 December 2020, Xi'an, China, pp. 1-5.
17. L. Zhang, G. Zhou, Y. Han, H. Lin and Y. Wu, Application of Internet of Things Technology and Convolutional Neural Network Model in Bridge Crack Detection, *IEEE Access*, **6** (2018) 39442–39451.
18. Y. Xie and S. Ming, Bridge defect detection technology based on machine vision and embedded. *Journal of Physics: Conference Series*, **1856**(1) (2021), 1-7.
19. Y. Zhang, J. Huang and F. Cai, Bridge Crack Detection Based On Bridge Crack Detection Based On Bridge Crack Detection Based On Bridge Crack Detection Based on Improved Algorithm On Bridge Crack Detection Based on Algorithm on Algorithm on an Improved Improved, *IFAC PapersOnLine*, **53**(2) (2020), 8205–8210.
20. X. W. Ye, T. Jin and P. Y. Chen, Structural crack detection using deep learning–based fully convolutional networks, *Advances in Structural Engineering*, **22**(16) (2019), 341-3419.
21. E. L. Droguett, J. Tapia, C. Yanez and R. Boroschek, Semantic segmentation model for crack images from concrete bridges for mobile devices, *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, (2020), 1-14.
22. R. Vignesh, B. Narenthiran, S. Manivannan, R. Arul Murugan and V. RajKumar, *Concrete Bridge Crack Detection Using Convolutional Neural Network, Materials, Design, and Manufacturing for Sustainable Environment*, eds. S. Mohan, S, Shankar and G. Rajeshkumar (Springer Singapore, 2021), pp. 797-812.
23. F. Tian, Y. Zhao, X. Che, Y. Zhao and D. Xin, Concrete crack identification and image mosaic based on image processing, *Applied Sciences*, **9**(22) (2019), 1-19.
24. H. Liang and J. Zou, Rock Image Segmentation of Improved Semi-supervised SVM–FCM Algorithm Based on Chaos, *Circuits, Systems, and Signal Processing*, **39**(2) (2020) 571–585.
25. Q. Xie and K. Zhou, An Optimized Ant Colony Algorithm for Text Edge Extraction, *2019 International Conference on Artificial Intelligence and Advanced Manufacturing*, 16-18 October 2019, Dublin, Ireland, pp. 289–292.

26. A. M. Hemeida, S. A. Hassan, A. A. A. Mohamed, S. Alkhalaf, M. M. Mahmoud, T. Senjyu and A. Bahaa El-Din, Nature-inspired algorithms for feed-forward neural network classifiers: A survey of one decade of research, *Ain Shams Engineering Journal*, **11**(3) (2020) 659–675.
27. J. N. D. Gupta, A. Majumder and D. Laha, Flowshop scheduling with artificial neural networks, *Journal of the Operational Research Society*, **71**(10) (2020), 1619–1637.
28. S. Yan, L. Jin, S. Duan, L. Zhao, C. Yao and W. Zhang, Power line image segmentation and extra matter recognition based on improved Otsu algorithm, *2013 2nd International Conference on Electric Power Equipment - Switching Technology (ICEPE-ST)*, 20-23 October 2013, Matsue, Japan, pp. 1-4.
29. E. Mohammed Abdelkader, M. Marzouk and T. Zayed, A self-adaptive exhaustive search optimization-based method for restoration of bridge defects images, *International Journal of Machine Learning and Cybernetics*, **11** (2020), 1659–1716.
30. R. J. Hemalatha, B. Babu, A. J. A. Dhivya, T. R. Thamizhvani, J. Elsa and J. R. Chandrasekaran, A Comparison of Filtering and Enhancement Methods in Malignant Melanoma Images, *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, 21-22 September 2017, Chennai, India, pp. 2704-2710.
31. J. Sivakumar, K. Thangavel and P. Saravanan, Computed radiography skull image enhancement using Wiener filter, *International Conference on Pattern Recognition, Informatics and Medical Engineering (PRIME-2012)*, 21-23 March 2012, Tamilnadu, India, pp. 307-311.
32. N. Hoang, Detection of Surface Crack in Building Structures Using Image Processing Technique with an Improved Otsu Method for Image Thresholding, *Advances in Civil Engineering*, Article ID 3924120 (2018) 1-10.
33. E. Mohammed Abdelkader, O. Moselhi, M. Marzouk and T. Zayed, A Multi-objective Invasive Weed Optimization Method for Segmentation of Distress Images, *Intelligent automation and soft computing*, **26**(4) (2020) 643-661.
34. S. Mishra and M. Panda, Bat Algorithm for Multilevel Colour Image Segmentation Using Entropy-Based Thresholding, *Arabian Journal for Science and Engineering*, **43** (2018) 7285–7314.
35. M. A. El. Aziz, A. A. Ewees and A. E. Hassanien, Whale Optimization Algorithm and Moth-

Flame Optimization for Multilevel Thresholding Image Segmentation, *Expert Systems With Applications*, **83** (2017) 242–256.

36. D. Oliva, S. Hinojosa, V. Osuna-Enciso, E. Cuevas, M. Pérez-Cisneros and G. Sanchez-Ante, Image Segmentation by Minimum Cross Entropy using Evolutionary Methods, *Soft Computing*, **23**(2) (2019) 431-450.
37. L. He and S. Huang, Modified firefly algorithm based multilevel thresholding for color image segmentation, *Neurocomputing*, **240** (2017) 152–174.
38. S. Sarkar, S. Das and S. S. Chaudhuri, A multilevel color image thresholding scheme based on minimum cross entropy and differential evolution, *Pattern Recognition Letters*, **54** (2015) 27–35.
39. X. Song, L. Tang, S. Zhao, X. Zhang, L. Li, J. Huang and W. Cai, Grey Wolf Optimizer for parameter estimation in surface waves, *Soil Dynamics and Earthquake Engineering*, **75** (2015) 147–157.
40. S. Mirjalili, S. Mohammad and A. Lewis, Grey Wolf Optimizer, *Advances in Engineering Software*, **69** (2014) 46–61.
41. E. N. Kalemci, , S. B. İkizler, T. Dede and Z. Angın, Design of reinforced concrete cantilever retaining wall using Grey wolf optimization algorithm, *Structures*, **23** (2020) 245–253.
42. M. Fahad, F. Aadil, Z. Rehman, S. Khan, P. A. Shah, K. Muhammad, C. Lloret, X. Wang, J. Lee and I. Mehmood, Grey wolf optimization based clustering algorithm for vehicular ad-hoc networks, *Computers and Electrical Engineering*, **70** (2018) 853–870.
43. S. Yu and H. Lu, An integrated model of water resources optimization allocation based on projection pursuit model – Grey wolf optimization method in a transboundary river basin, *Journal of Hydrology*, **559** *(2018)* 156–165.
44. M. Pradhan, P. K. Roy and T. Pal, Grey wolf optimization applied to economic load dispatch problems, *Electrical Power and Energy Systems*, **83** (2016) 325–334.
45. A. Korashy, S. Kamel, A. R. Youssef and F. Jurado, Solving Optimal Coordination of Direction Overcurrent Relays Problem Using Grey Wolf Optimization (GWO) Algorithm, *2018 20th International Middle East Power Systems Conference*, 18-20 December 2018, Cairo, Egypt, pp. 621–625.
46. K. S. Nimma, M. D. A. Al-Falahi, H. D. Nguyen, S. D. G. Jayasinghe, T. S. Mahmoud and M. Negnevitsky, Grey Wolf optimization-based optimum energy-management and battery-

sizing method for grid-connected microgrids, *Energies*, **11**(4) (2018) 1–27.

47. S. Sukumar, M. Marsadek, A. Ramasamy and H. Mokhlis, Grey Wolf Optimizer Based Battery Energy Storage System Sizing for Economic Operation of Microgrid, *2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe)*, 12-15 June 2018, Palermo, Italy, pp. 1–5.
48. A. Dogan, A, Load Frequency Control of Two Area and Multi Source Power System Using Grey Wolf Optimization Algorithm, *2019 11th International Conference on Electrical and Electronics Engineering (ELECO)*, 28-30 November 2019, Bursa, Turkey, pp. 81-84.
49. Z. M. Gao and J. Zhao, An improved grey Wolf optimization algorithm with variable weights, *Computational Intelligence and Neuroscience*, Article ID 2981282 (2019) 1-13.
50. H. Faris, I. Aljarah, M. A. Al-Betar and S. Mirjalili, Grey wolf optimizer: a review of recent variants and applications, *Neural Computing and Applications*, **30**(2) (2018) 413–435.
51. S. M. M. H. Daneshvar, P. Alikhah Ahari Mohajer and S. M. Mazinani, Energy-Efficient Routing in WSN: A Centralized Cluster-Based Approach via Grey Wolf Optimizer, *IEEE Access*, **7** (2019) 170019–170031.
52. W. Long, J. Jiao, X. Liang and M. Tang, Inspired grey wolf optimizer for solving large-scale function optimization problems, *Applied Mathematical Modelling*, **60** (2018) 112–126.
53. S. Gupta, K. Deep, H. Moayedi, L. K. Foong and A. Assad, Sine cosine grey wolf optimizer to solve engineering design problems, *Engineering with Computers*, (2020) 1-27.
54. S. Singh and J. C. Bansal, Grey Wolf Optimizer with Crossover and Opposition-Based Learning, *Proceedings of 6th International Conference on Harmony Search, Soft Computing and Applications*, eds. S.M. Nigdeli, J.H. Kim, G. Bekdaş and A. Yadav (Springer, Singapore, 2021), pp. 401–410.
55. J. C. Bansal and S. Singh, A better exploration strategy in Grey Wolf Optimizer. *Journal of Ambient Intelligence and Humanized Computing*, **12**(1) (2021) 1099–1118.
56. S. Gupta and K. Deep, A memory-based Grey Wolf Optimizer for global optimization tasks, *Applied Soft Computing*, **93** (2020) 1-31
57. S. Gupta and K. Deep, A novel Random Walk Grey Wolf Optimizer, *Swarm and Evolutionary Computation*, **44** (2019), 101–112.

58. S. Gupta and K. Deep, Improved Grey Wolf Optimizer Based on Opposition-Based Learning, *Advances in Intelligent Systems and Computing*, eds. J. Bansal, K. Das, A. Nagar and K. Deep (Springer, Singapore, 2019), pp. 327-388.
59. S. Gupta and K. Deep, An opposition-based chaotic Grey Wolf Optimizer for global optimisation tasks, *Journal of Experimental and Theoretical Artificial Intelligence*, **31**(5) (2019) 751–779.
60. G. Kou, Y. Peng and G. Wang, Evaluation of clustering algorithms for financial risk analysis using MCDM methods, *Information Sciences*, **275** (2014), 1–12.
61. G. Kou, H. Xiao, M. Cao and L. H. Lee, Optimal computing budget allocation for the vector evaluated genetic algorithm in multi-objective simulation optimization, *Automatica*, **129** (2021), 1-14.
62. G. Kou, Y. Xu, Y. Peng, F. Shen, Y. Chen, K.Chang and S. Kou, Bankruptcy prediction for SMEs using transactional data and two-stage multiobjective feature selection, *Decision Support Systems*, **140** (2021), 1-14.
63. M. Khan and T. Shah, A Literature Review on Image Encryption Techniques, *3D Research*, **5**(4) (2014) 1-25.
64. M. Kaushik, R. Sharma and A. Vidhyarthi, Analysis of Spatial Features in CBIR System, *International Journal of Computer Applications*, **54**(17) (2012) 11–15.
65. G. Kou, Y. Lu, Y. Peng and Y. Shi, Evaluation of classification algorithms using MCDM and rank correlation, *International Journal of Information Technology and Decision Making*, **11**(1) (2012), 197–225.
66. T. Dawood, Z. Zhu and T. Zayed, Machine vision-based model for spalling detection and quantification in subway networks, *Automation in Construction*, **81** (2017) 149–160.
67. Y. Zhang, J. Li, F. Gao, X. Song, Y. Li and S. Zhao, Research on Risk Assessment Method of Distribution Network with Distributed Generation Based on Latin Hypercube Sampling, *2019 IEEE 8th International Conference on Advanced Power System Automation and Protection (APAP)*, 21-24 October 2019, Xi'an, China, pp. 1125–1130.
68. F. Zou, D. Chen, R. Lu, and P. Wang, Hierarchical multi-swarm cooperative teaching–learning-based optimization for global optimization, *Soft Computing*, **21**(23) (2017), 6983–7004.
69. R. S. Pratama, A. F. Huda, A. Wahana, W. Darmalaksana, Q. U., Safitri and A. Rahman,

Analysis of Fuzzy C-Means Algorithm on Indonesian Translation of Hadits Text, *2019 IEEE 5th International Conference on Wireless and Telematics (ICWT)*, 25-26 July 2019, Yogyakarta, Indonesia, pp. 1-5.

70. A. Sheshasayee and P. Sharmila, Comparative study of fuzzy C means and K means algorithm for requirements clustering, *Indian Journal of Science and Technology*, **7**(6) (2014) 853–857.
71. A. B. Goktepe, S. Altun and A. Sezer, Soil clustering by fuzzy c-means algorithm, *Advances in Engineering Software*, **36** (2005) 691–698.
72. M. Horng, M and R. Liou, Multilevel minimum cross entropy threshold selection based on the firefly algorithm, *Expert Systems With Applications*, **38**(12) (2011) 14805–14811.
73. D. Oliva, S. Hinojosa, V. Osuna-Enciso, E. Cuevas, M. Pérez-Cisneros and G. Sanchez-Ante, Image Segmentation by Minimum Cross Entropy using Evolutionary Methods, *Soft Computing*, **23**(2) (2017) 431-450.
74. A. K. Khairuzzaman and S. Chaudhury, Multilevel Thresholding using Grey Wolf Optimizer for Image Segmentation, *Expert Systems With Applications*, **86** (2017) 64–76.
75. E.Najafi, A robust embedding and blind extraction of image watermarking based on discrete wavelet transform, *Mathematical Sciences*, **11**(4) (2017) 307–318.
76. M. A.Qureshi and M. Deriche, A new wavelet based efficient image compression algorithm using compressive sensing, *Multimedia Tools and Applications*, **75**(12) (2016) 6737–6754.
77. D. R. Nayak, R. Dash and B. Majhi, Brain MR image classification using two-dimensional discrete wavelet transform and AdaBoost with random forests, *Neurocomputing*, **177** (2016) 188–197.
78. S. Tarare, A. Anjikar and H. Turkar, Fingerprint based gender classification using DWT transform, *2015 International Conference on Computing Communication Control and Automation*, 26-27 February 2015, Pune, India, pp. 689–693.
79. K. T. Shanavaz and P. Mythili, A fingerprint-based hybrid gender classification system using genetic algorithm, *International Journal of Computational Vision and Robotics*, **6**(4) (2016) 399-413.
80. M. M. Ibrahim, N. S. Abdel Kader and M. Zorkany, Video multiple watermarking technique based on image interlacing using dwt, *Scientific World Journal*, Article ID 634828 (2014) 1-13.

81. S. Sahu, A. P. Rao and S. T. Mishra, Fingerprints based gender classification using Adaptive Neuro Fuzzy Inference System, *2015 International Conference on Communications and Signal Processing (ICCSP)*, 2-4 April 2015, Melmaruvathur, India, pp. 1218–1222.

82. U. D. Dixit and M. S. Shirdhonkar, Fingerprint-Based Document Image Retrieval, *International Journal of Image and Graphics*, **19**(2) (2019) 1–17.

83. S. R. Surya and M. Abdul Rahiman, Cloud detection from satellite images based on Haar wavelet and clustering, *2017 International Conference on Nextgen Electronic Technologies: Silicon to Software (ICNETS2)*, 23-25 March 2017, Chennai, India, pp. 163–167.

84. S. A. Khan, S. Hussain, S. Xiaoming and S. Yang, An Effective Framework for Driver Fatigue Recognition Based on Intelligent Facial Expressions Analysis, *IEEE Access*, **6** (2018) 67459–67468.

85. H. Kanagaraj and V. Muneeswaran, Image compression using HAAR discrete wavelet transform, *2020 5th International Conference on Devices, Circuits and Systems (ICDCS)*, 5-6 March 2020, Coimbatore, India, pp. 271–274.

86. G. R. N. Kumari, S. S. Jeeru and S. Maruthuperumal, Color Image Watermarking Using Wavelet Transform Based on HVS Channel, *ICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of Computer Society of India- Vol I,* eds. S. Satapathy, P. Avadhani, S. Udgata and S. Lakshminarayana (Springer, Cham, 2014), pp. 59-67.

87. B. L. Gunjal and S. N. Mali, Comparative performance analysis of digital image watermarking scheme in DWT and DWT-FWHT-SVD domains, Comparative performance analysis of digital image watermarking scheme in DWT and DWT-FWHT-SVD domains, *2014 Annual IEEE India Conference (INDICON)*, 11-13 December 2014, Pune, India, pp. 1-6.

88. A.Sharma, S. S. Bhadauria and R. Gupta, Image Compression and Sparsity Measurement by using Multilevel and Different Wavelet Functions, *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 12-14 December 2019, Palladam, India, pp. 517–520.

89. F. N. Thakkar and V. K. Srivastava, A fast watermarking algorithm with enhanced security using compressive sensing and principle components and its performance analysis against a set of standard attacks, *Multimedia Tools and Applications*, **76**(14) (2017) 15191–15219.

90. S. Kumar and Y. K. Jain, Performance Evaluation and Analysis of Image Restoration

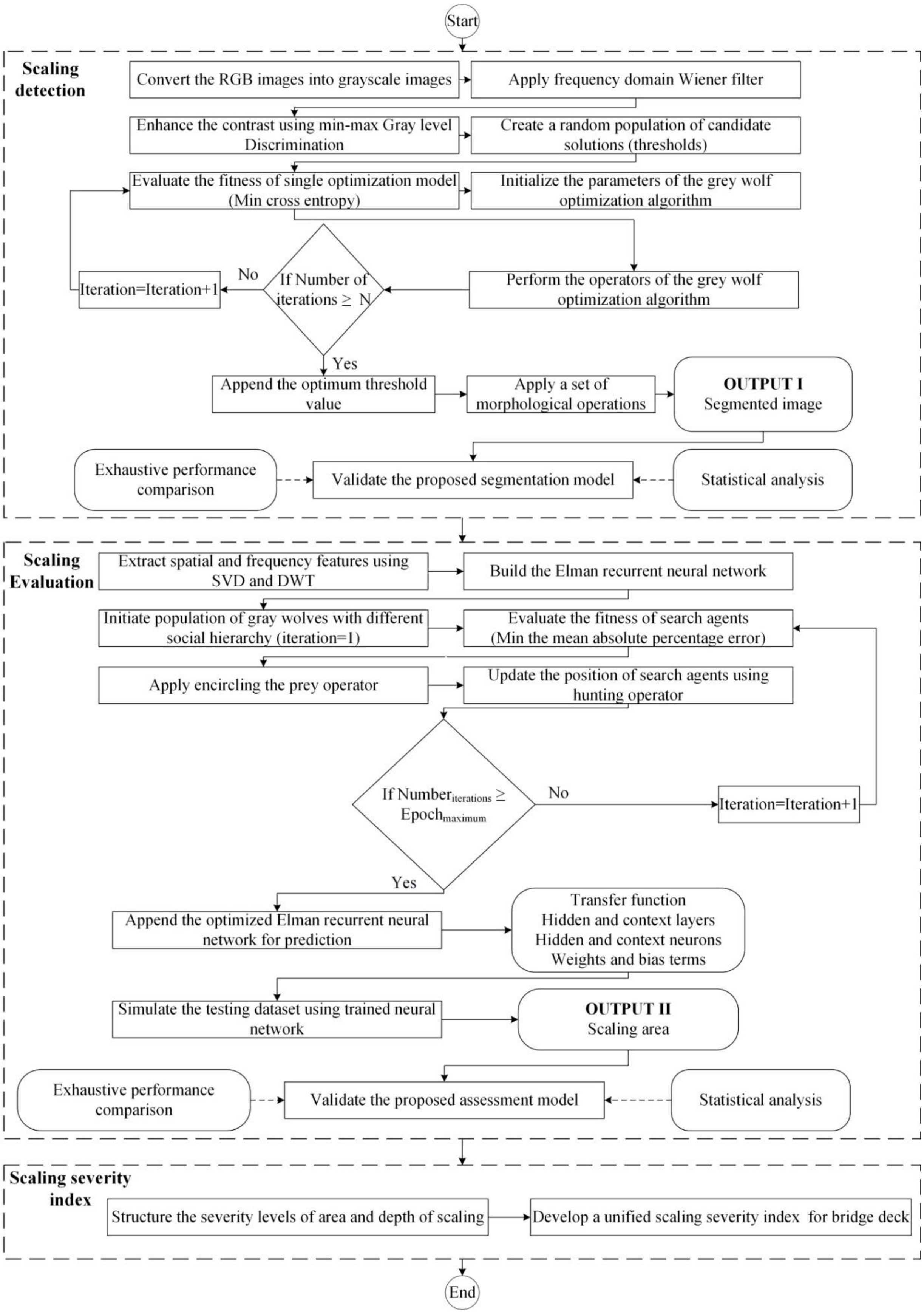
Technique using DWT. *International Journal of Computer Applications*, **72**(18) (2013) 11–20.

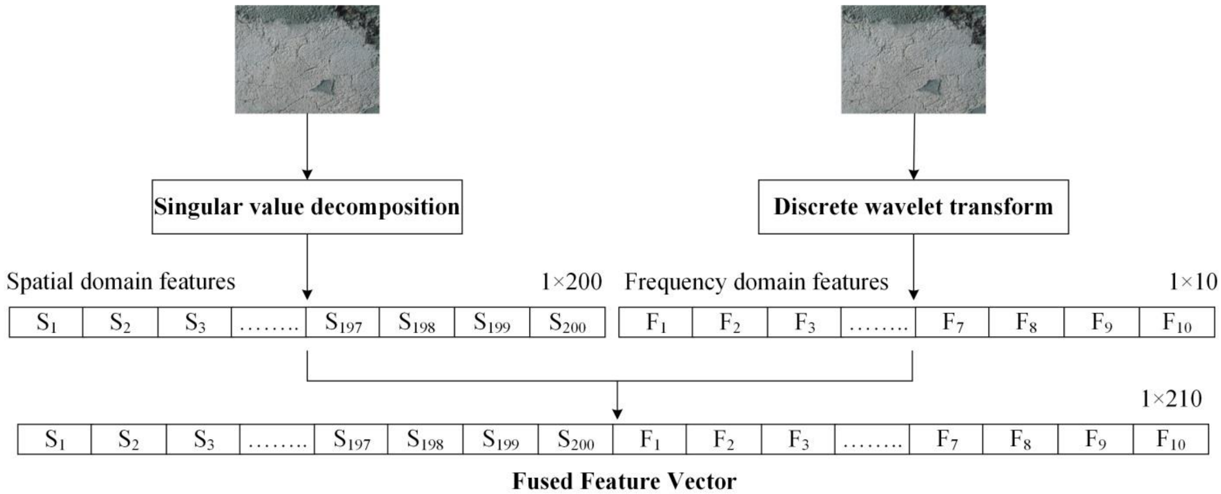
91. A. Lauraitis, R. Maskeli and R. Damaševičius, ANN and Fuzzy Logic Based Model to Evaluate Huntington Disease Symptoms, *Journal of Healthcare Engineering*, Article ID 4581272 (2018) 1-10.
92. K. Kurach and K. Pawlowski, Predicting Dangerous Seismic Activity with Recurrent Neural Networks, *Proceedings of the Federated Conference on Computer Science*, 11-14 September 2016, Gdansk, Poland, pp. 239–243.
93. L. Yang, and A. Shami, On hyperparameter optimization of machine learning algorithms: Theory and practice, *Neurocomputing*, **415** (2020) 295–316.
94. M. Tsiakmaki, G. Kostopoulos, S. Kotsiantis and O. Ragos, Implementing autoML in educational data mining for prediction tasks, *Applied Sciences*, **10**(1) (2020) 1–27.
95. M. Juszczyk, On the search of models for early cost estimates of bridges: An SVM-based approach, *Buildings*, **10**(1) (2020) 1-17
96. S. S. Roy, P. Samui, I. Nagtode, H. Jain, V. Shivaramakrishnan and B. Mohammadi-ivatloo, Forecasting heating and cooling loads of buildings: a comparative performance analysis, *Journal of Ambient Intelligence and Humanized Computing*, **11**(3) (2020) 1253–1264.

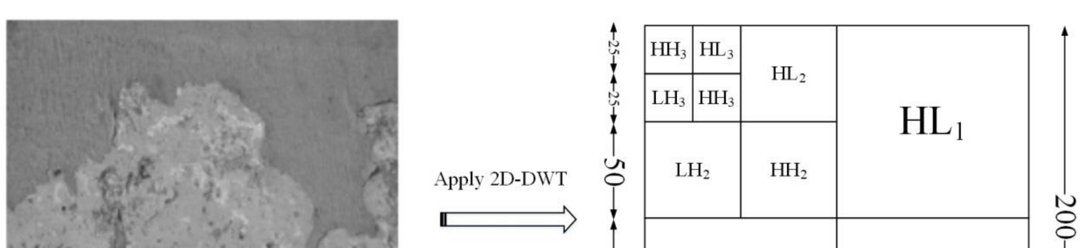
List of Figures

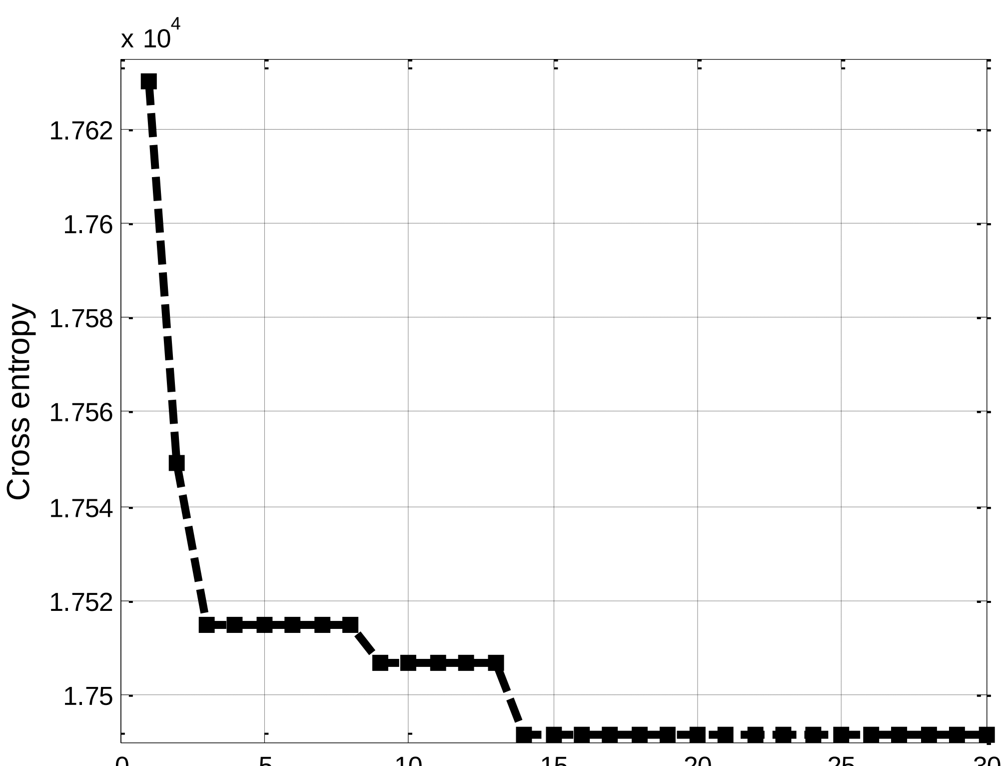
Figure 20: Visualization of the performances of the meta-heuristic-based Elman neural network and conventional prediction models based on mean absolute error

Figure 21: Box plots of the mean absolute percentage error obtained by the hybrid meta-heuristic-based Elman neural network models

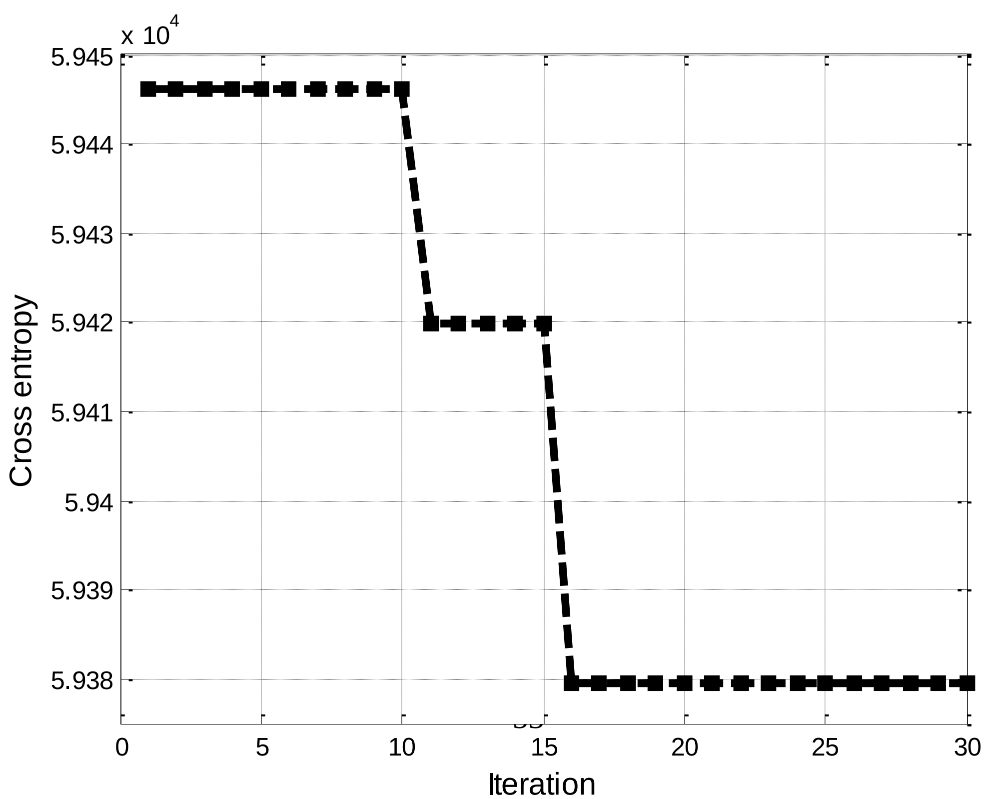

Start
Scaling detection
Convert the RGB images into grayscale images
Apply frequency domain Wiener filter
Enhance the contrast using min-max Gray level Discrimination
Create a random population of candidate solutions (thresholds)
Evaluate the fitness of single optimization model (Min cross entropy)
Initialize the parameters of the grey wolf optimization algorithm
Iteration=Iteration+1
No
If Number of iterations ≥ N
Perform the operators of the grey wolf optimization algorithm
Yes
Append the optimum threshold value
Apply a set of morphological operations
OUTPUT I
Segmented image
Exhaustive performance comparison
Validate the proposed segmentation model
Statistical analysis
Scaling Evaluation
Extract spatial and frequency features using SVD and DWT
Build the Elman recurrent neural network
Initiate population of gray wolves with different social hierarchy (iteration=1)
Evaluate the fitness of search agents (Min the mean absolute percentage error)
Apply encircling the prey operator
Update the position of search agents using hunting operator
If Number$_{iterations}$ ≥ Epoch$_{maximum}$
No
Iteration=Iteration+1
Yes
Append the optimized Elman recurrent neural network for prediction
Transfer function
Hidden and context layers
Hidden and context neurons
Weights and bias terms
Simulate the testing dataset using trained neural network
OUTPUT II
Scaling area
Exhaustive performance comparison
Validate the proposed assessment model
Statistical analysis
Scaling severity index
Structure the severity levels of area and depth of scaling
Develop a unified scaling severity index for bridge deck
End


Singular value decomposition
Discrete wavelet transform
Spatial domain features
1×200
Frequency domain features
1×10
S1 S2 S3 …….. S197 S198 S199 S200
F1 F2 F3 …….. F7 F8 F9 F10
1×210
S1 S2 S3 …….. S197 S198 S199 S200 F1 F2 F3 …….. F7 F8 F9 F10
Fused Feature Vector


Apply 2D-DWT
HH3
HL3
LH3
HH3
HL2
LH2
HH2
HL1
25
25
50
200

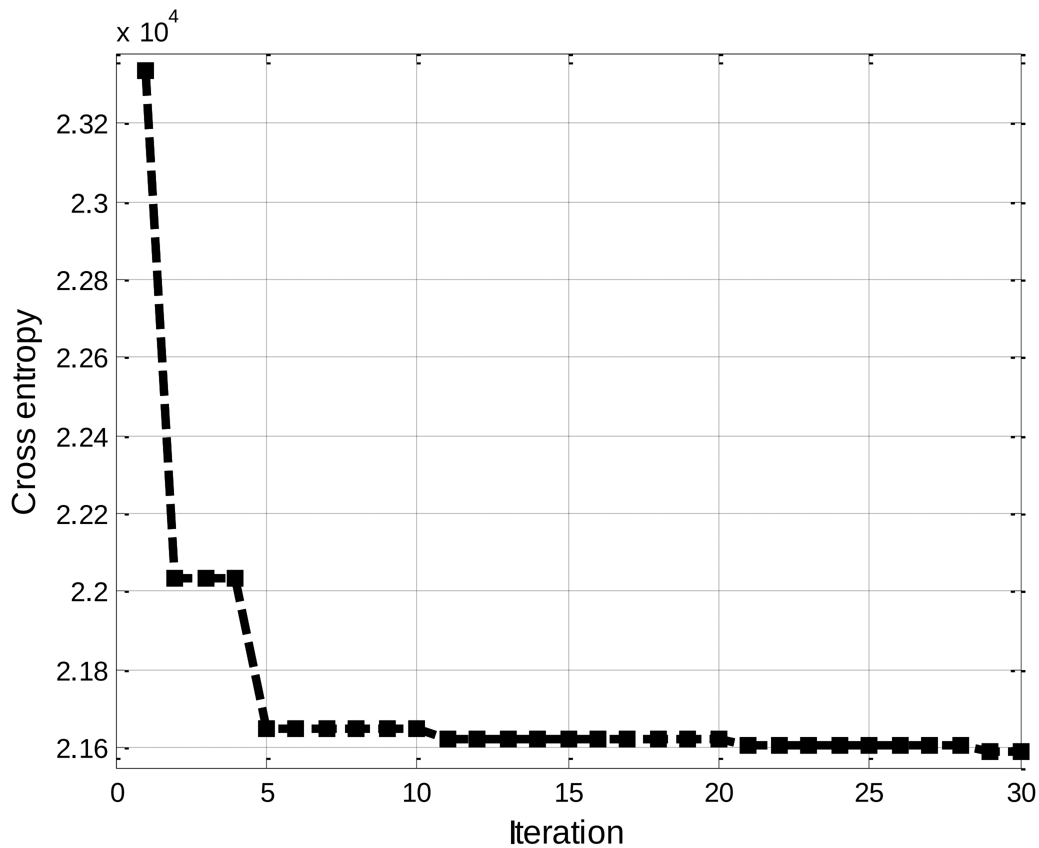


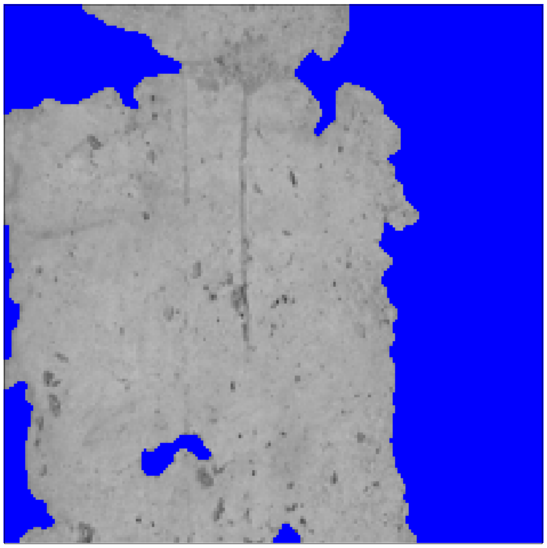
x 10^4
1.762
1.76
1.758
1.756
1.754
1.752
1.75
Cross entropy
0
5
10
15
20
25
30



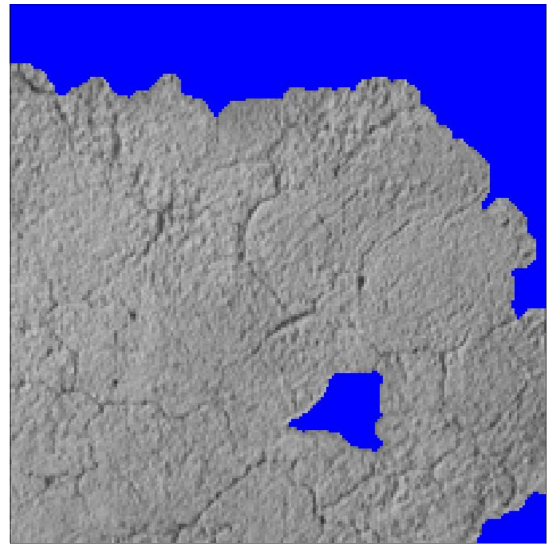
(a) Image “C”

x 10^4

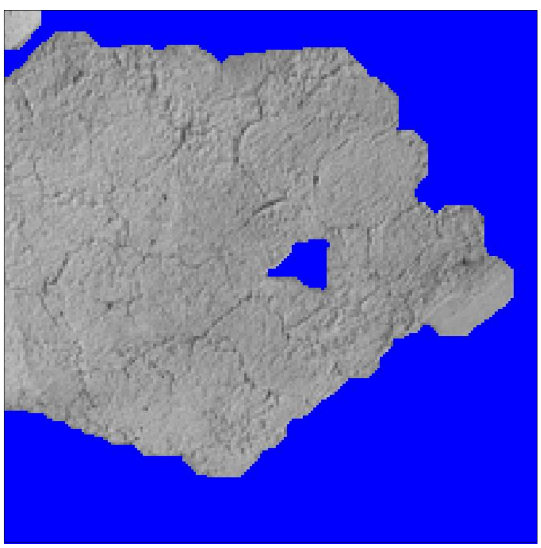

x 10^4
2.32
2.3
2.28
2.26
2.24
2.22
2.2
2.18
2.16
Cross entropy
0
5
10
15
20
25
30
Iteration



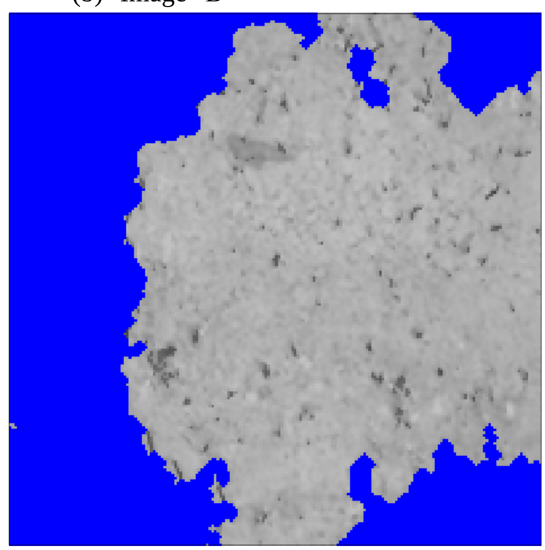
(a) Image “A”



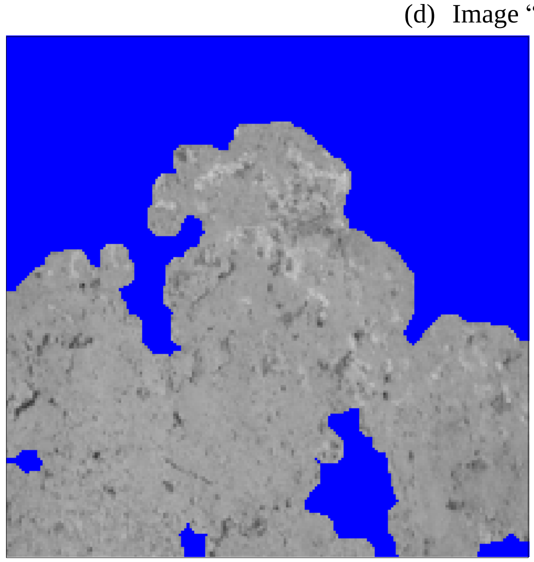
(b) Image “B”



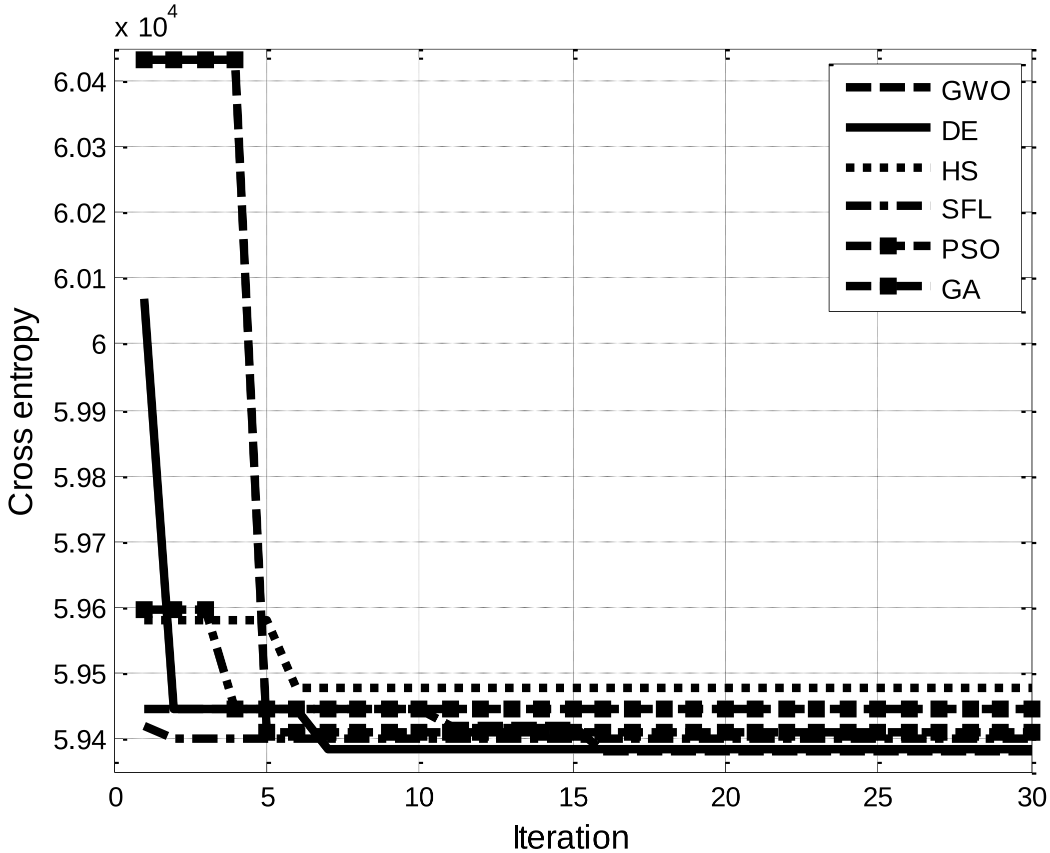
(c) Image “C”

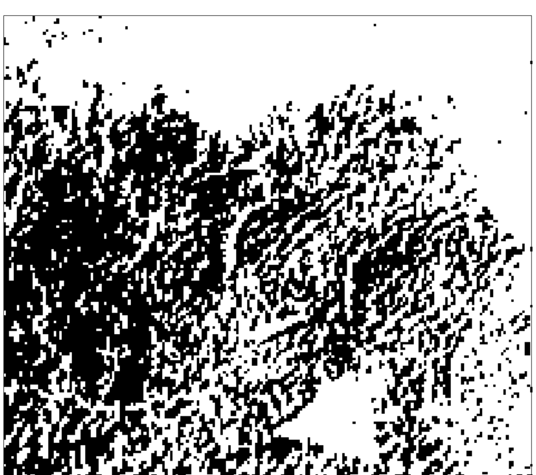


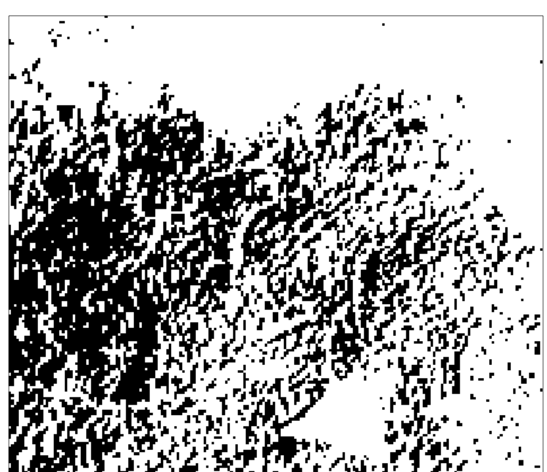
(d) Image “D”

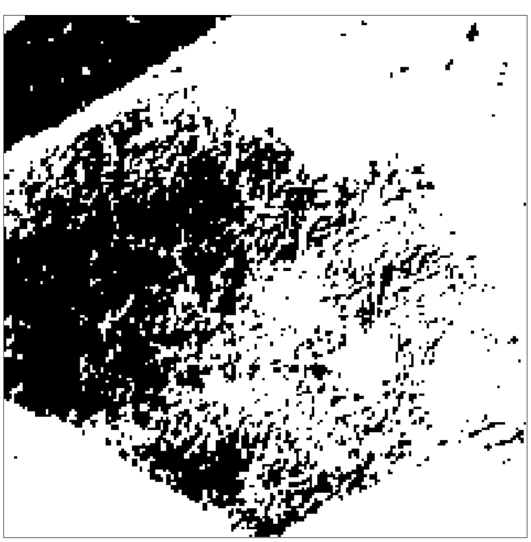


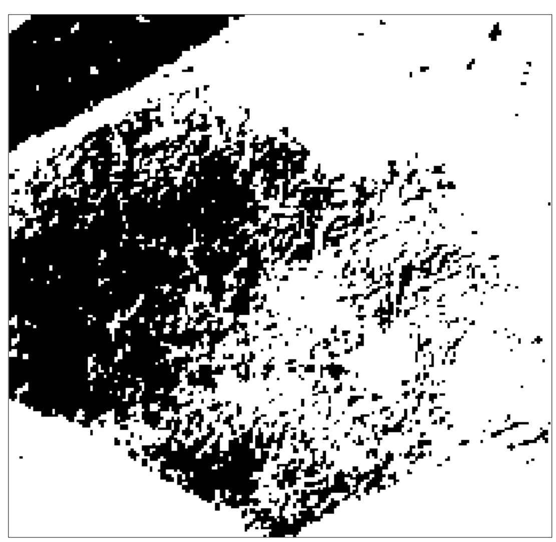
(e) Image “E”


x 10^4
Cross entropy
6.04
6.03
6.02
6.01
6
5.99
5.98
5.97
5.96
5.95
5.94
0
5
10
15
20
25
30
Iteration
GWO
DE
HS
SFL
PSO
GA



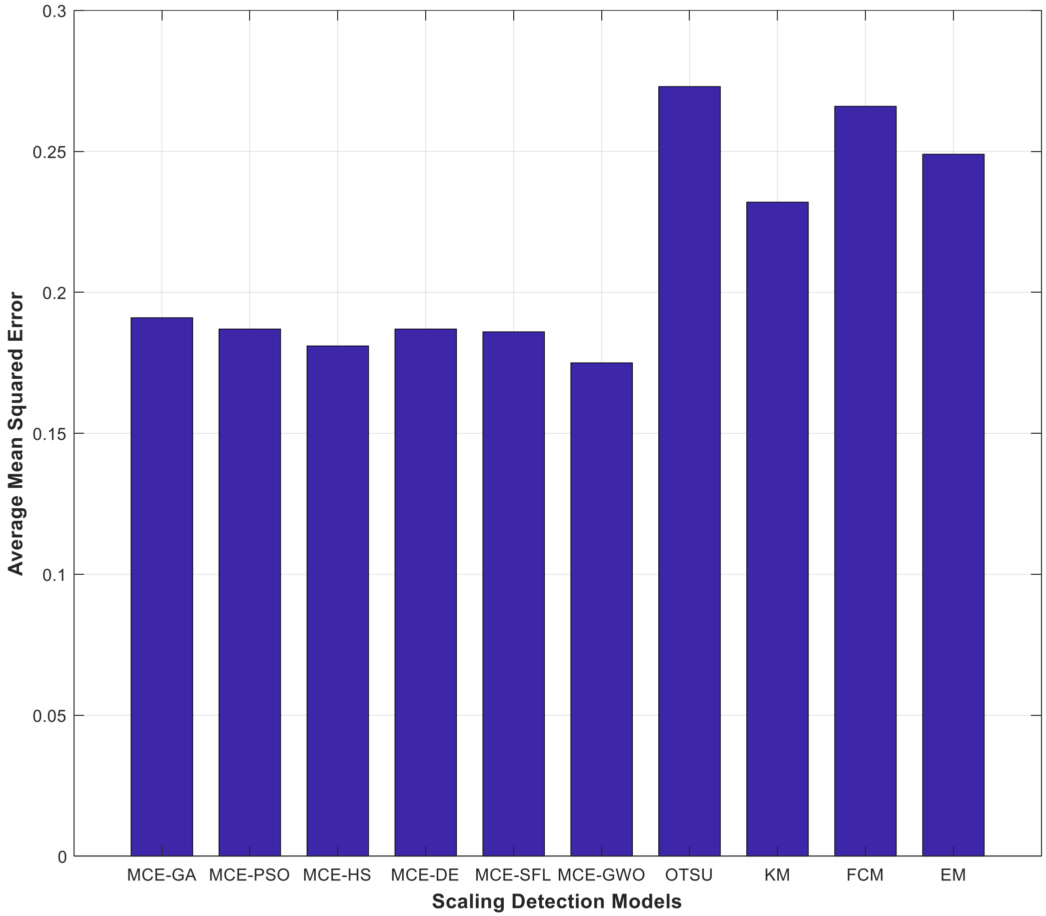




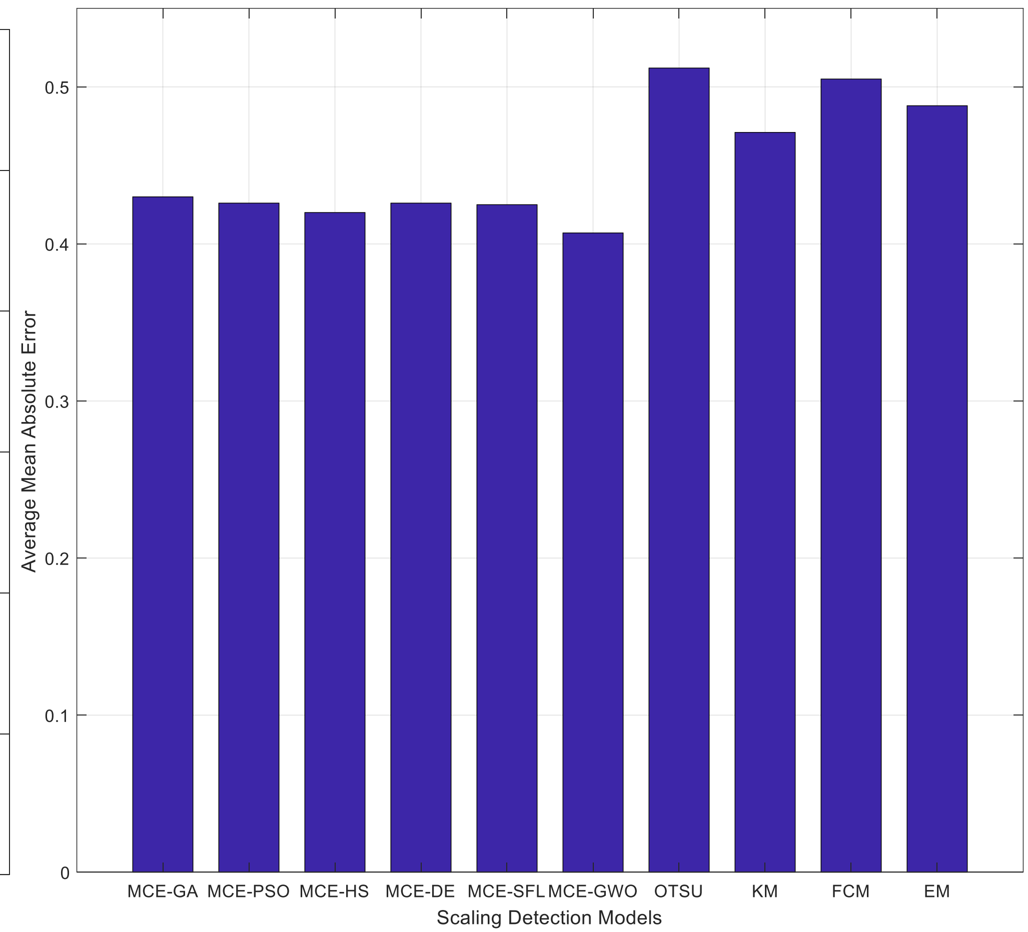


(a) Otsu

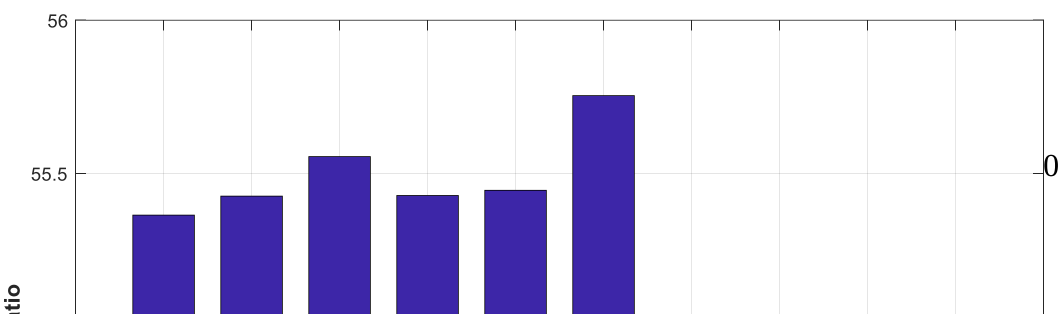
(b) K-means clustering

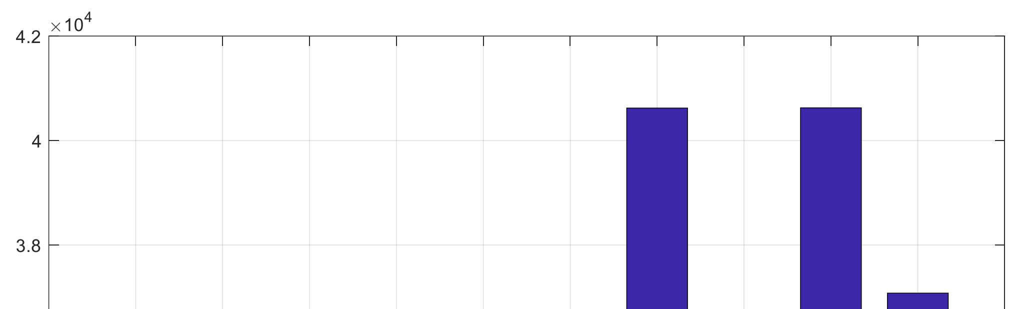


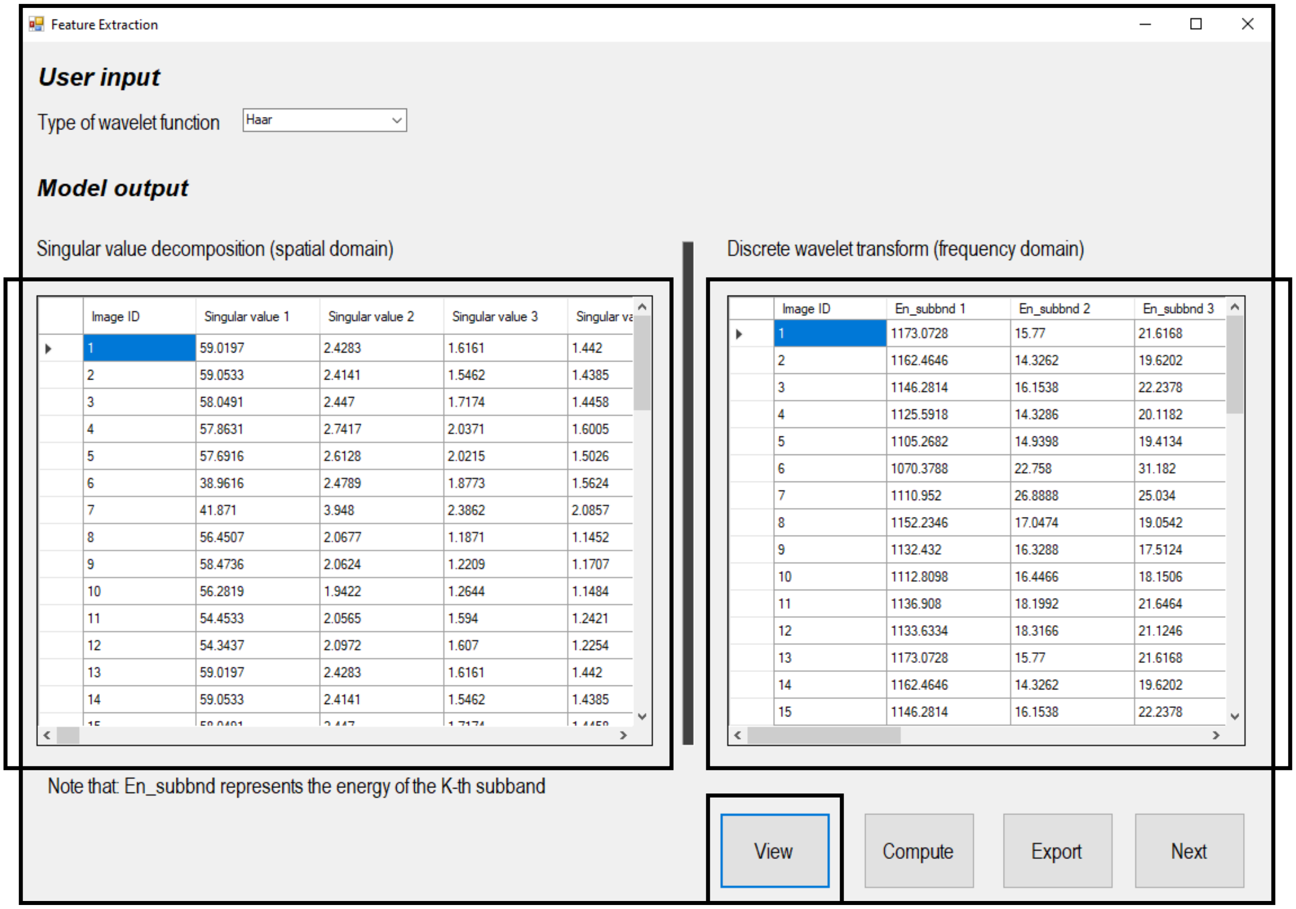
(a) Illustration of the obtained average mean squared error

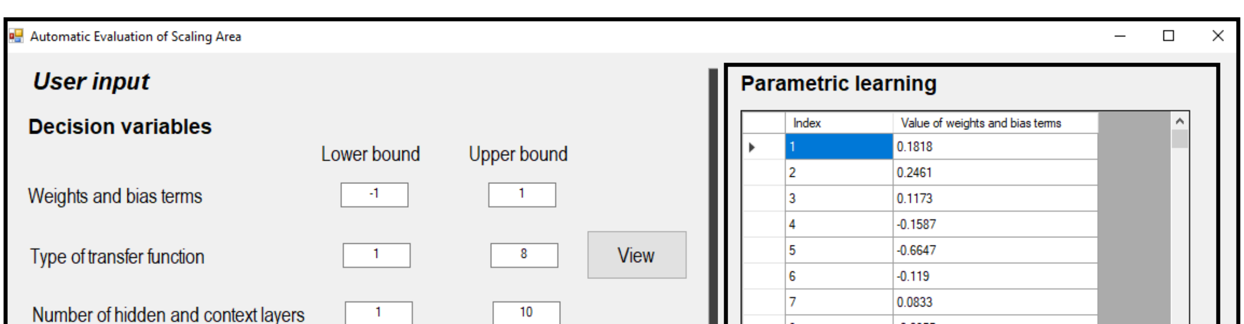


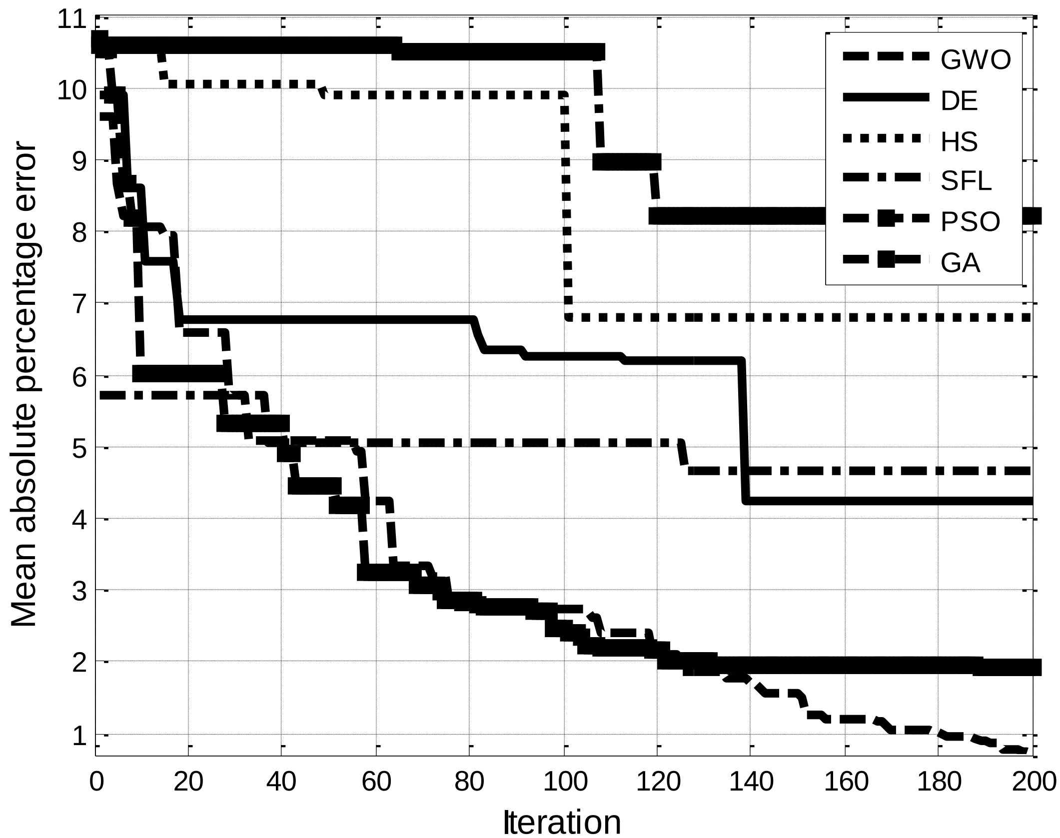
(b) Illustration of the obtained average mean absolute error



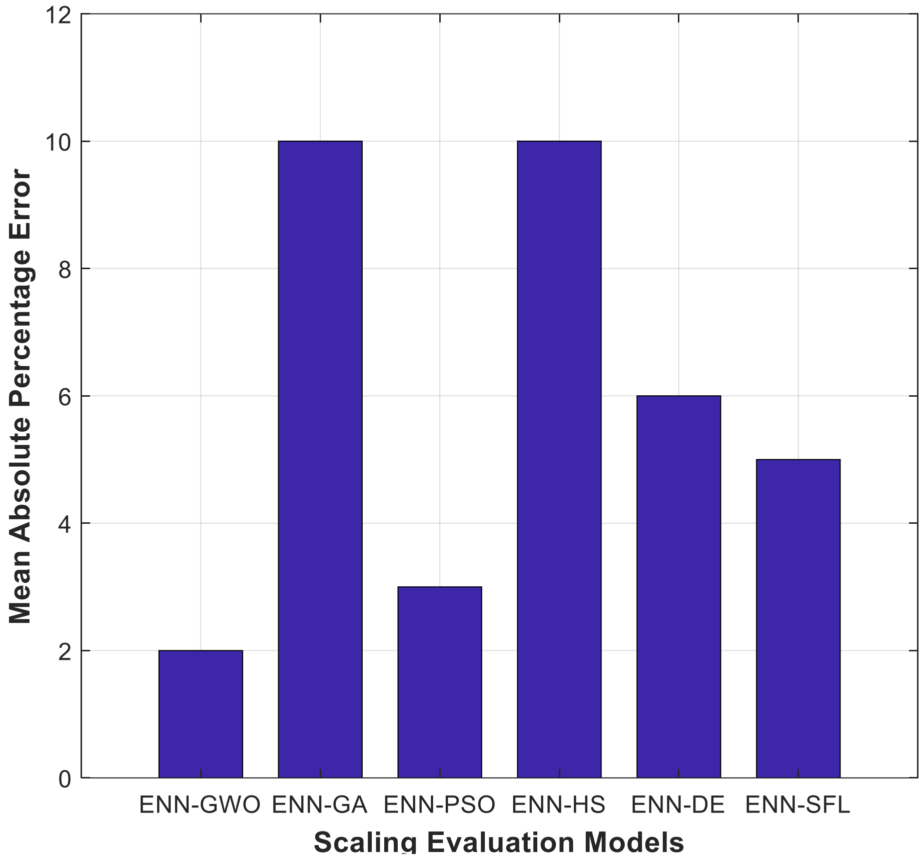



Feature Extraction
User input
Type of wavelet function
Haar
Model output
Singular value decomposition (spatial domain)
Discrete wavelet transform (frequency domain)
Image ID | Singular value 1 | Singular value 2 | Singular value 3
1 | 59.0197 | 2.4283 | 1.6161 | 1.442
2 | 59.0533 | 2.4141 | 1.5462 | 1.4385
3 | 58.0491 | 2.447 | 1.7174 | 1.4458
4 | 57.8631 | 2.7417 | 2.0371 | 1.6005
5 | 57.6916 | 2.6128 | 2.0215 | 1.5026
6 | 38.9616 | 2.4789 | 1.8773 | 1.5624
7 | 41.871 | 3.948 | 2.3862 | 2.0857
8 | 56.4507 | 2.0677 | 1.1871 | 1.1452
9 | 58.4736 | 2.0624 | 1.2209 | 1.1707
10 | 56.2819 | 1.9422 | 1.2644 | 1.1484
11 | 54.4533 | 2.0565 | 1.594 | 1.2421
12 | 54.3437 | 2.0972 | 1.607 | 1.2254
13 | 59.0197 | 2.4283 | 1.6161 | 1.442
14 | 59.0533 | 2.4141 | 1.5462 | 1.4385
Image ID | En_subbnd 1 | En_subbnd 2 | En_subbnd 3
1 | 1173.0728 | 15.77 | 21.6168
2 | 1162.4646 | 14.3262 | 19.6202
3 | 1146.2814 | 16.1538 | 22.2378
4 | 1125.5918 | 14.3286 | 20.1182
5 | 1105.2682 | 14.9398 | 19.4134
6 | 1070.3788 | 22.758 | 31.182
7 | 1110.952 | 26.8888 | 25.034
8 | 1152.2346 | 17.0474 | 19.0542
9 | 1132.432 | 16.3288 | 17.5124
10 | 1112.8098 | 16.4466 | 18.1506
11 | 1136.908 | 18.1992 | 21.6464
12 | 1133.6334 | 18.3166 | 21.1246
13 | 1173.0728 | 15.77 | 21.6168
14 | 1162.4646 | 14.3262 | 19.6202
15 | 1146.2814 | 16.1538 | 22.2378
Note that: En_subbnd represents the energy of the K-th subband
View
Compute
Export
Next

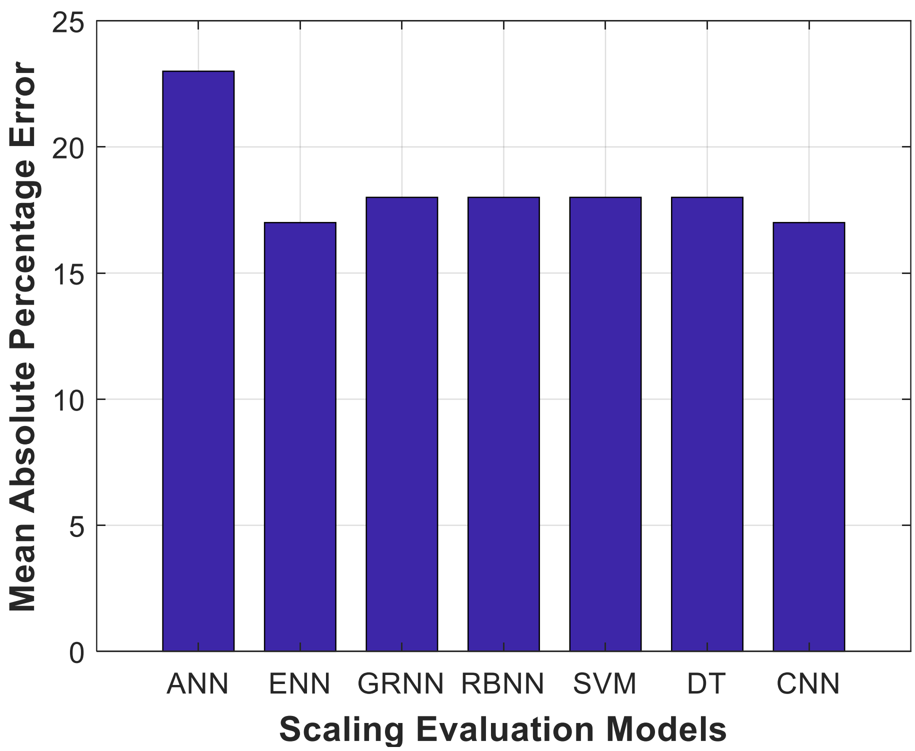

Automatic Evaluation of Scaling Area
User input
Decision variables
Lower bound
Upper bound
Weights and bias terms
-1
1
Type of transfer function
1
8
View
Number of hidden and context layers
1
10
Parametric learning
Index | Value of weights and bias terms
1 | 0.1818
2 | 0.2461
3 | 0.1173
4 | -0.1587
5 | -0.6647
6 | -0.119
7 | 0.0833



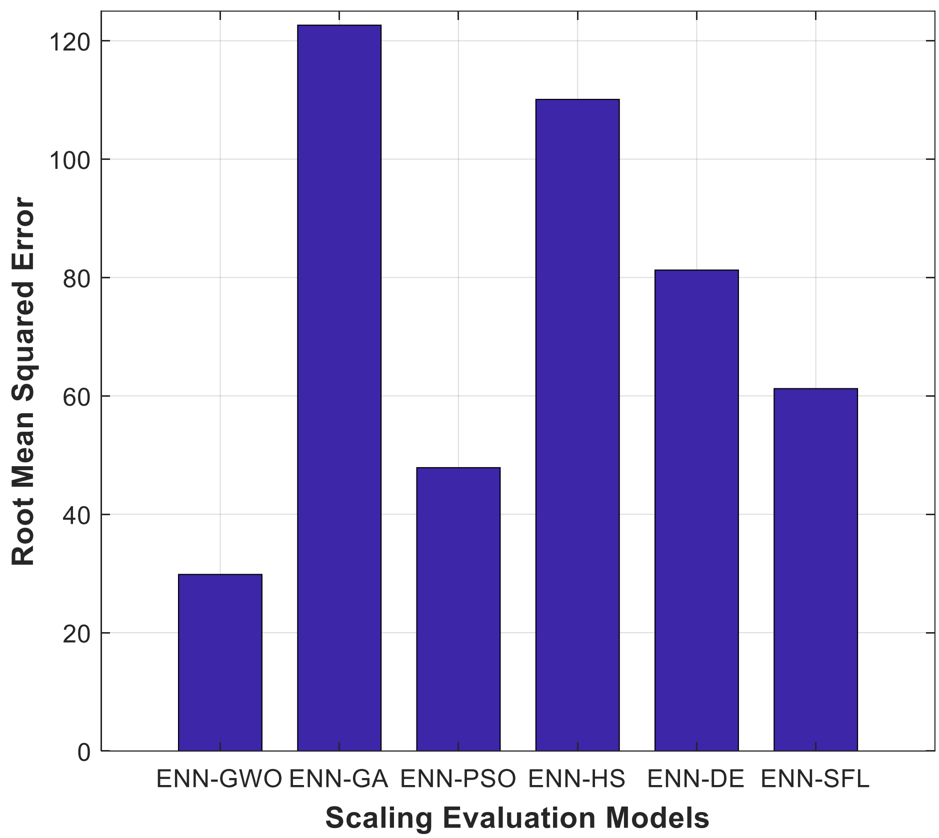
Mean absolute percentage error
Iteration
GWO
DE
HS
SFL
PSO
GA
0
20
40
60
80
100
120
140
160
180
200
1
2
3
4
5
6
7
8
9
10
11



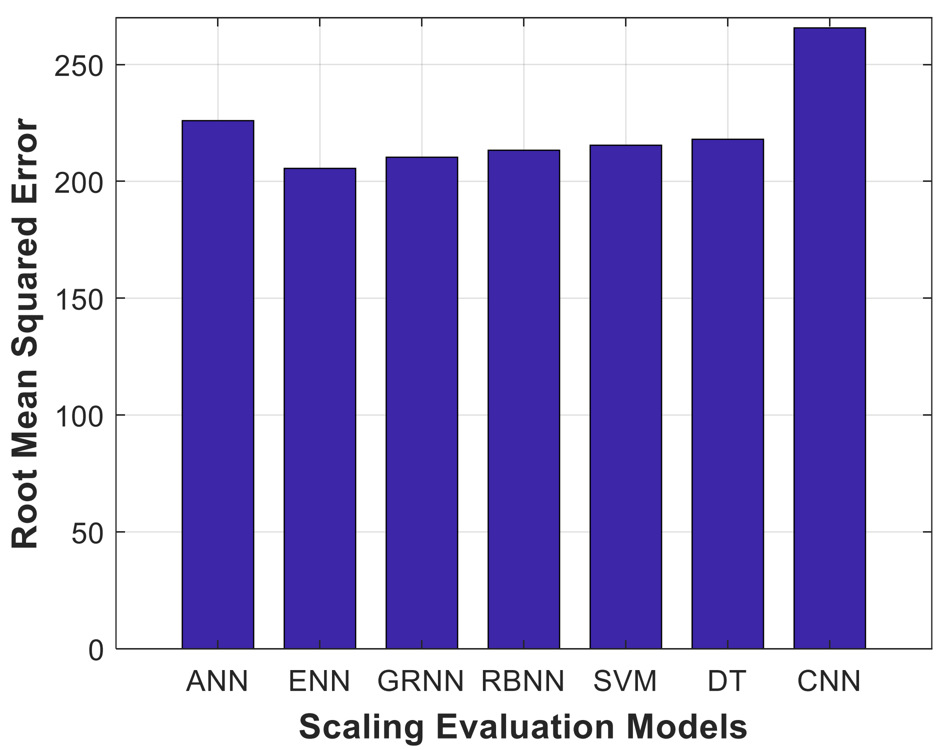
(a) Performances of the meta-heuristic-based Elman neural network models



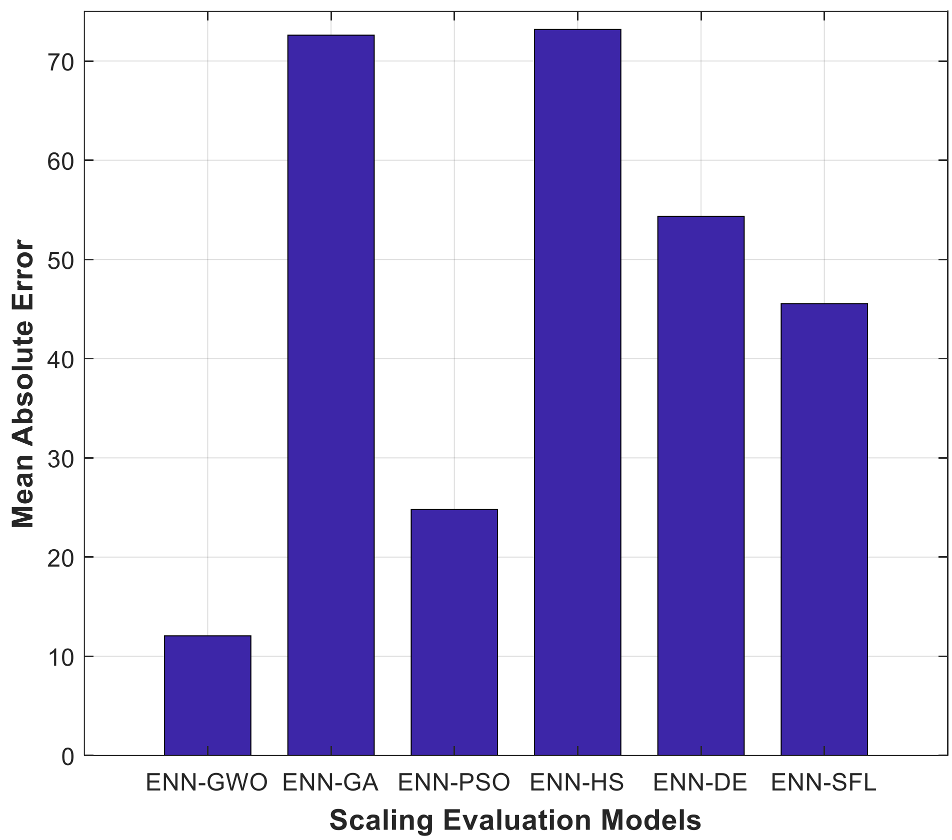
(b) Performances of the conventional prediction models



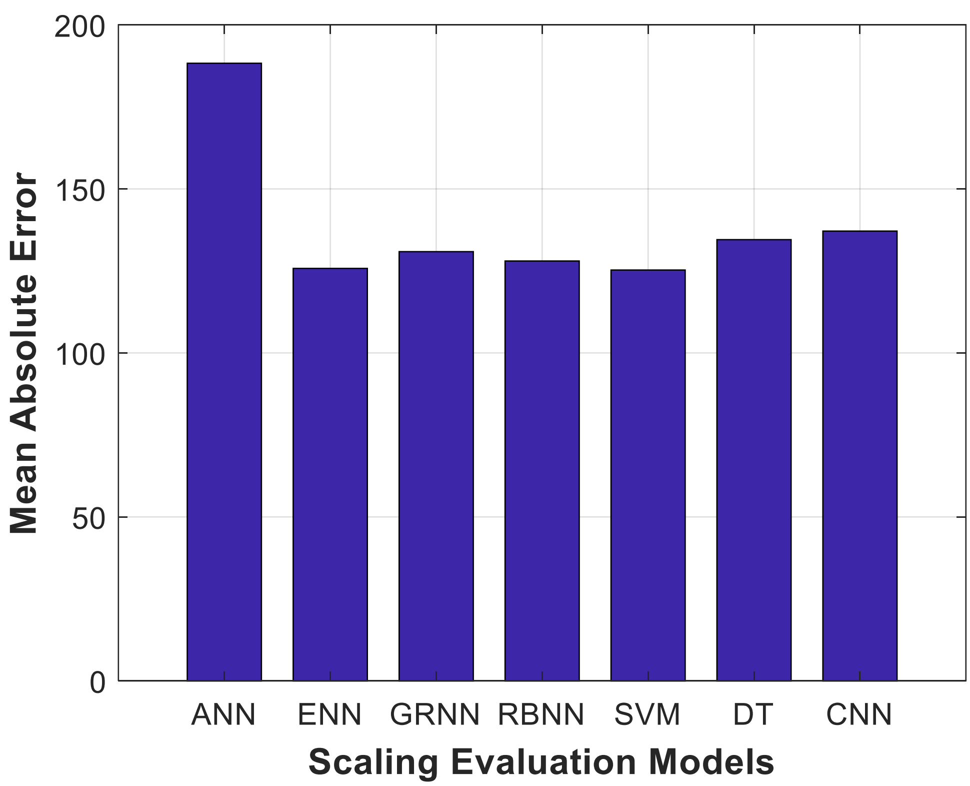
(a) Performances of the meta-heuristic-based Elman neural network models



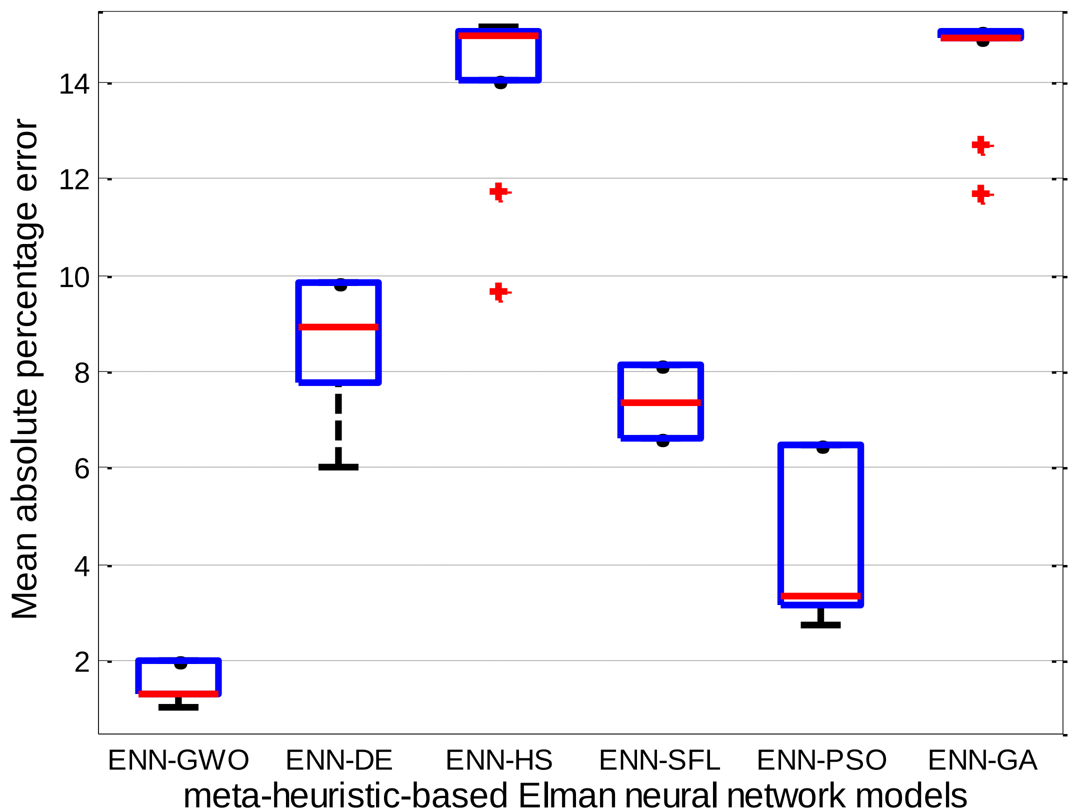
(b) Performances of the conventional prediction models



(a) Performances of the meta-heuristic-based Elman neural network models



(b) Performances of the conventional prediction models


Mean absolute percentage error
14
12
10
8
6
4
2
ENN-GWO
ENN-DE
ENN-HS
ENN-SFL
ENN-PSO
ENN-GA
meta-heuristic-based Elman neural network models

List of Tables

Table 1: Performance comparison between the meta-heuristic-based models for scaling detection

Segmentation model	Average mean-squared error	Average mean absolute error	Average peak signal to noise ratio	Average cross entropy
MCE − GA	0.191	0.43	55.364	27309.737
MCE − PSO	0.187	0.426	55.426	26120.03
MCE − HS	0.181	0.42	55.555	30353.82
MCE − DE	0.187	0.426	55.428	26741.602
MCE − SFL	0.186	0.425	55.445	26233.768
MCE − GWO	0.175	0.407	55.754	26011.019

Table 2: Overall comparative analysis between the different scaling detection models

Segmentation model	Average mean-squared error	Average mean absolute error	Average peak signal to noise ratio	Average cross entropy
$\mathrm{MCE} - \mathrm{GWO}$	0.175	0.407	55.754	26011.019
Otsu	0.273	0.512	53.813	40621.781
K-means clustering	0.232	0.471	54.495	35876.343
Fuzzy C-means clustering	0.266	0.505	53.898	40626.631
Expectation maximization	0.249	0.488	54.174	37078.934

Table 3: $P-$values of the different performance indicators of the segmentation models using Shapiro-Wilk test for normality

Model	Description	$P-$value
Otsu	Mean absolute error	7.82×10^{-1} (H_0)
K-means clustering	Mean absolute error	5.19×10^{-2} (H_0)
Fuzzy C-means clustering	Mean absolute error	7.6×10^{-1} (H_0)
Expectation maximization	Mean absolute error	5.23×10^{-2} (H_0)
$MCE-GA$	Mean absolute error	5.3×10^{-2} (H_0)
$MCE-PSO$	Mean absolute error	6×10^{-2} (H_0)
$MCE-HS$	Mean absolute error	8.11×10^{-2} (H_0)
$MCE-DE$	Mean absolute error	3.05×10^{-1} (H_0)
$MCE-SFL$	Mean absolute error	6.53×10^{-2} (H_0)
$MCE-GWO$	Mean absolute error	7.87×10^{-2} (H_0)
Otsu	Peak signal to noise ratio	6.77×10^{-1} (H_0)
K-means clustering	Peak signal to noise ratio	2.84×10^{-1} (H_0)
Fuzzy C-means clustering	Peak signal to noise ratio	6.54×10^{-1} (H_0)
Expectation maximization	Peak signal to noise ratio	3.86×10^{-1} (H_0)
$MCE-GA$	Peak signal to noise ratio	6.33×10^{-2} (H_0)
$MCE-PSO$	Peak signal to noise ratio	4.77×10^{-2} (H_0)
$MCE-HS$	Peak signal to noise ratio	6.1×10^{-2} (H_0)
$MCE-DE$	Peak signal to noise ratio	2.82×10^{-1} (H_0)
$MCE-SFL$	Peak signal to noise ratio	2.24×10^{-1} (H_0)
$MCE-GWO$	Peak signal to noise ratio	6.04×10^{-2} (H_0)

Table 4: Statistical comparison of the developed scaling segmentation model against other models for MAE based on student's t-test

Pair of segmentation models	**P − value**
(MCE − GWO, Otsu)	H_1 (P − value =0)
(MCE − GWO, K-means clustering)	H_1 (P − value =0)
(MCE − GWO, Fuzzy C-means clustering)	H_1 (P − value =0)
(MCE − GWO, Expectation maximization)	H_1 (P − value =0)
(MCE − GWO, MCE − GA)	H_1 (P − value $=1.2\times10^{-2}$)
(MCE − GWO, MCE − PSO)	H_1 (P − value =0)
(MCE − GWO, MCE − HS)	H_1 (P − value $=2.6\times10^{-3}$)
(MCE − GWO, MCE − DE)	H_1 (P − value =0)
(MCE − GWO, MCE − SFL)	H_1 (P − value =0)

Table 5: Statistical comparison of the developed scaling segmentation model against other models for PSNR based on student's t-test

Pair of segmentation models	$\mathbf{P-value}$
$(\mathrm{MCE-GWO}$, Otsu)	$\mathrm{H_1}$ ($\mathrm{P-value}$ =0)
$(\mathrm{MCE-GWO}$, K-means clustering)	$\mathrm{H_1}$ ($\mathrm{P-value}$ =1×10^{-5})
$(\mathrm{MCE-GWO}$, Fuzzy C-means clustering)	$\mathrm{H_1}$ ($\mathrm{P-value}$ =0)
$(\mathrm{MCE-GWO}$, Expectation maximization)	$\mathrm{H_1}$ ($\mathrm{P-value}$ =0)
$(\mathrm{MCE-GWO, MCE-GA})$	$\mathrm{H_1}$ ($\mathrm{P-value}$ =6.79×10^{-3})
$(\mathrm{MCE-GWO, MCE-PSO})$	$\mathrm{H_1}$ ($\mathrm{P-value}$ =2×10^{-5})
$(\mathrm{MCE-GWO, MCE-HS})$	$\mathrm{H_1}$ ($\mathrm{P-value}$ =4.6×10^{-4})
$(\mathrm{MCE-GWO, MCE-DE})$	$\mathrm{H_1}$ ($\mathrm{P-value}$ =3×10^{-5})
$(\mathrm{MCE-GWO, MCE-SFL})$	$\mathrm{H_1}$ ($\mathrm{P-value}$ =4×10^{-5})

Table 6: Comparative analysis of the performance metrics of the prediction models based on split validation

Prediction model	Mean absolute percentage error	Root-mean squared error	Mean absolute error
ENN − GWO	1.513%	29.836	12.066
ENN − GA	10.145%	122.627	72.602
ENN − PSO	2.907%	47.875	24.794
ENN − HS	9.854%	110.087	73.187
ENN − DE	5.987%	81.255	54.348
ENN − SFL	5.183%	61.23	45.529
ANN	22.627%	225.963	188.306
ENN	17.026%	205.539	125.760
GRNN	18.276%	210.309	130.873
RBNN	17.789%	213.304	128.027
SVM	17.807%	215.447	125.272
DT	18.237%	217.996	134.530
CNN	17.199%	265.646	137.141

Table 7: Comparative analysis of the performance metrics of the Prediction models based on 10-fold cross validation

Prediction model	Mean absolute percentage error	Root-mean squared error	Mean absolute error
ENN − GWO	1.521%	29.992	12.137
ENN − GA	10.241%	123.815	73.326
ENN − PSO	2.936%	48.361	25.047
ENN − HS	9.943%	111.164	73.878
ENN − DE	6.035%	81.958	54.779
ENN − SFL	5.22%	61.696	45.881
ANN	23.306%	232.823	194.135
ENN	17.196%	207.731	127.081
GRNN	18.551%	213.517	133.007
RBNN	18.145%	217.737	130.642
SVM	18.377%	222.356	129.371
DT	18.839%	225.317	139.033
CNN	17.594%	271.828	140.401

Table 8: $P-$values of the mean absolute percentage error of the prediction models using Shapiro-Wilk test for normality

Prediction model	$P-$value	Prediction model	$P-$value
$ENN-GWO$	1.09×10^{-3} (H_1)	ANN	4.12×10^{-2} (H_1)
$ENN-GA$	8.5×10^{-4} (H_1)	ENN	3×10^{-5} (H_1)
$ENN-PSO$	1.04×10^{-2} (H_1)	$GRNN$	1.19×10^{-2} (H_1)
$ENN-HS$	4.9×10^{-4} (H_1)	$RBNN$	4.57×10^{-2} (H_1)
$ENN-DE$	8.9×10^{-4} (H_1)	SVM	4.29×10^{-2} (H_1)
$ENN-SFL$	2.8×10^{-3} (H_1)	DT	3.62×10^{-2} (H_1)
		CNN	4.79×10^{-2} (H_1)

Table 9: Statistical comparison of the developed $\mathrm{ENN-GWO}$ model against other prediction learning models based on non-parametric tests

Pair of machine learning models	Wilcoxn	Mann-Whitney-U	Kruskal–Wallis	Binomial sign	Mood’s median
$(\mathrm{ENN-GWO}$ $,\mathrm{ENN-GA})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=4.95\times10^{-3})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=1.42\times10^{-4})$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$
$(\mathrm{ENN-GWO},$ $\mathrm{ENN-PSO})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=4.95\times10^{-3})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=1.37\times10^{-4})$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$
$\mathrm{ENN-GWO},$ $\mathrm{ENN-HS})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=5.06\times10^{-3})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=1.46\times10^{-4})$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$
$(\mathrm{ENN-GWO},$ $\mathrm{ENN-DE})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=4.78\times10^{-3})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=1.28\times10^{-4})$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$
$(\mathrm{ENN-GWO},$ $\mathrm{ENN-SFL})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=4.98\times10^{-3})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=1.37\times10^{-4})$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$
$(\mathrm{ENN-GWO},$ $\mathrm{ANN})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=5.06\times10^{-3})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=1.46\times10^{-4})$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$
$(\mathrm{ENN-GWO},$ $\mathrm{ENN})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=5.06\times10^{-3})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=1.46\times10^{-4})$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$
$(\mathrm{ENN-GWO},$ $\mathrm{GRNN})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=5.06\times10^{-3})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=1.42\times10^{-4})$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$
$(\mathrm{ENN-GWO},$ $\mathrm{RBNN})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=5.03\times10^{-3})$	$\mathrm{H_1}$ $(\mathrm{P-value}$ $=1.37\times10^{-4})$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$	$\mathrm{H_1}$ $(\mathrm{P-value}=0)$

(ENN − GWO, SVM)	H_1 (P − value $=5.06\times10^{-3}$)	H_1 (P − value $=1.46\times10^{-4}$)	H_1 (P − value =0)	H_1 (P − value =0)	H_1 (P − value =0)
(ENN − GWO, DT)	H_1 (P − value $=5.06\times10^{-3}$)	H_1 (P − value $=1.45\times10^{-4}$)	H_1 (P − value =0)	H_1 (P − value =0)	H_1 (P − value =0)
(ENN − GWO, CNN)	H_1 (P − value $=5.06\times10^{-3}$)	H_1 (P − value $=1.46\times10^{-4}$)	H_1 (P − value =0)	H_1 (P − value =0)	H_1 (P − value =0)

Table 10: Sample of the cluster memberships of scaling area obtained from the FCM algorithm

Data point	Degree of membership				Assigned Cluster
	Cluster 1	Cluster 2	Cluster 3	Cluster 4	
710.013	0.557	0.343	0.072	0.027	Cluster 1
1217.916	0.002	0.005	0.015	0.979	Cluster 4
775.289	0.131	0.775	0.072	0.022	Cluster 2
827.366	0.007	0.980	0.011	0.003	Cluster 2
1059.227	0.011	0.050	0.810	0.129	Cluster 3
930.423	0.028	0.414	0.515	0.043	Cluster 3

Table 11: Severity levels of scaling area and depth

Condition category	Scaling area	Scaling depth
Good	Less than 30%	Less than 1.7 mm
Medium	Between 30% and 40%	Between 1.7 and 3 mms
Poor	Between 40% and 50%	Between 3 and 4 mms
Very Poor	More than 50%	More than 4 mm