Case Studies in Construction Materials 19 (2023) e02530

Contents lists available at ScienceDirect

Case Studies in Construction Materials

journal homepage: www.elsevier.com/locate/cscm

Classification of defects in wooden structures using pre-trained models of convolutional neural network

Rana Ehtisham [a], Waqas Qayyum [a], Charles V. Camp [b], Vagelis Plevris [c,*], Junaid Mir [d], Qaiser-uz Zaman Khan [a], Afaq Ahmad [a,b]

[a] *Department of Civil Engineering, University of Engineering & Technology Taxila, Pakistan*
[b] *Civil Engineering Department, The University of Memphis, TN, USA*
[c] *Department of Civil and Environmental Engineering, Qatar University, Doha, Qatar*
[d] *Department of Electrical Engineering, University of Engineering & Technology Taxila, Pakistan*

ARTICLE INFO

Keywords:
Wooden defects
Defects Classification
Pre-trained Models
CNN

ABSTRACT

Wooden structures, over time, are challenged by different types of defects. Due to mechanical and weathering effects, these defects can occur in the form of cracks, live and dead knots, dampness, and others. Because of the risk of damage or complete failure, treatment of these defects is necessary, but doing so necessitates their proper identification and classification (categorization). Crack identification and categorization must be part of the inspection procedure for engineering structures in the built environment. Convolutional neural networks (CNNs), a sub-type of Deep Learning (DL), can automatically classify the images of wooden structures to identify such defects. In this study, ten pre-trained models of CNN, namely ResNet18, ResNet50, ResNet101, ShuffleNet, GoogLeNet, Inception-V3, MobileNet-V2, Xception, Inception-ResNet-V2, and NASNet-Mobile are evaluated for the tasks of classification and prediction of defects in wooden structures. Each pre-trained CNN model is additionally trained and validated on an image dataset of 9000 images, equally divided into three classes: cracks, knots, and intact (undamaged). A smaller dataset of 300 images is separately used for testing purposes. Statistical parameters such as accuracy, precision, recall, and F1-score are computed for each CNN model. The Inception-V3 model proved to be the best CNN model for classifying defects in wooden structures based on the model's accuracy, processing time and overall performance.

1. Introduction

The development of advanced engineered wood products has made wood a competitive material for large construction projects. Consequently, there has been an increase in the use of wooden structures in new buildings. As timber structures, particularly in public buildings, have become more abundant [1], the need for evaluating and monitoring these structures has become crucial. Wooden defects encompass imperfections or irregularities in wood, including knots, cracks, decay, insect damage, and other structural issues. These defects have a significant impact on important qualities and characteristics of wood, such as its strength and durability. Knots, for example, can affect the quality of surface finishes and diminish the aesthetic appeal of the final product. Additionally, knots affect mechanical properties such as elasticity, stiffness, and rigidity [2]. Cracks are also well-known defects in wooden structures, as

* Corresponding author.
E-mail address: vplevris@qu.edu.qa (V. Plevris).

https://doi.org/10.1016/j.cscm.2023.e02530
Received 4 April 2023; Received in revised form 22 September 2023; Accepted 2 October 2023
Available online 4 October 2023

depicted in Fig. 1. They can arise from various factors such as drying, growing stresses, or environmental conditions [3].

Artificial intelligence (AI) and machine learning (ML) techniques have been widely applied in the field of civil and structural engineering [4,5]. These applications include various areas such as structural health monitoring [6], identification of structural damage [7,8], structural modeling [9–12], dynamic analysis of structures [13], planning and management of construction projects [14], risk assessment [15], prediction of strength and other structural attributes [16,17], enhancement of energy efficiency [18], among others. Recently, AI methods have also been utilized for classifying wooden defects, which has significantly reduced the time and cost associated with assessing the condition of wooden structures [19,20]. By automatically and rapidly identifying, evaluating, and addressing structural defects, the risk of damage to the structure and its occupants can be greatly minimized. Therefore, using modern ML-based monitoring techniques, wooden structural defects can be quickly identified and rectified more efficiently, providing an early indication of any potential issues.

Deep learning (DL) is an effective approach in machine learning for feature extraction, transformation, and pattern analysis in both supervised and unsupervised conditions. It involves multiple layers of nonlinear information processing [21]. In particular, Convolutional Neural Networks (CNNs), a type of DL model designed for visual data analysis, have been proven to be highly successful in tasks such as image recognition and classification. Therefore, they are well-suited for capturing and learning complex features from images, identifying patterns and anomalies, and detecting and analyzing various types of defects, including wood defects [22,23], and tire defects [24–27], among others. To give an example, consider an image which is composed of a matrix of pixel values. The initial layer of representation in the learning process often detects the presence of edges at specific angles and locations within the image. Even with minor variations in the positioning of these edges, the subsequent layer is capable of recognizing patterns by identifying specific combinations of edges. Similarly, subsequent layers continue this process by identifying features as combinations of other elements. For example, the third layer may combine patterns to form larger combinations that correspond to recognized parts of well-known features. This particular subset of deep learning places emphasis on discernible visual features in order to differentiate between different images.

Initially, CNNs were predominantly employed for solving relatively straightforward tasks such as recognizing hand-written digits or character recognition [28]. However, in modern times, approaches based on CNNs have gained widespread recognition and acceptance as the preferred method for tasks such as image classification, object identification, and image segmentation. The typical architecture of a CNN is illustrated in Fig. 2.

This study utilizes ten pre-trained CNN models to classify defects in wooden structures. The models used are ResNet18, ResNet50, ResNet101, ShuffleNet, GoogLeNet, Inception-V3, MobileNet-V2, Xception, Inception-ResNet-V2, and NASNet-Mobile. Each model undergoes training, validation, and finally testing on the same datasets to assess their performance. The evaluation is based on processing time and accuracy in predicting true defect classifications. A comparative study is conducted by calculating the confusion matrix of each model. Additionally, each model independently tests three images to validate its performance. This work stands out due to the utilization of pre-trained CNN models, which are subsequently trained on a unique dataset comprising custom-made images of wood defects. This training is conducted specifically to accurately classify these structures and their potential defects. It is worth noting that new pre-trained CNN models continue to emerge in current literature, demonstrating impressive performance in various computer vision tasks. Examples of such models include GhostNetV2 [29] and FasterNet [30].

The first section of this work provided an introduction to wooden defects and the use of CNN in defect classification. Section 2 provides a summary of previous research studies on wooden defect classification and the application of CNN in this field. Section 3 outlines the proposed methodology for this study. The experimental study is presented and summarized in Section 4. The results and discussions are presented in Section 5, and the conclusions are summarized in Section 6.

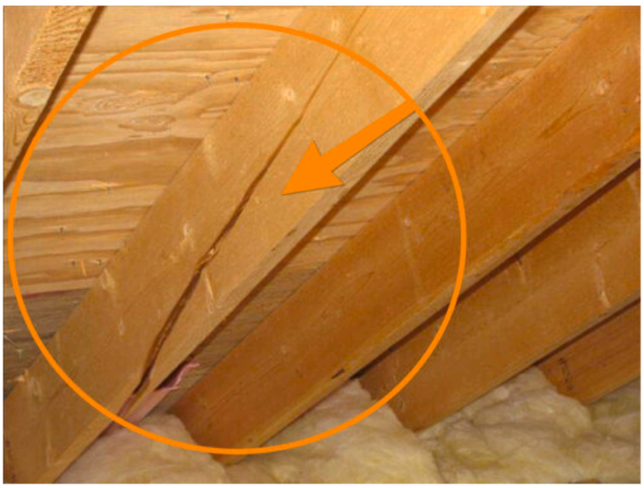


Fig. 1. Cracks in a wooden structure.

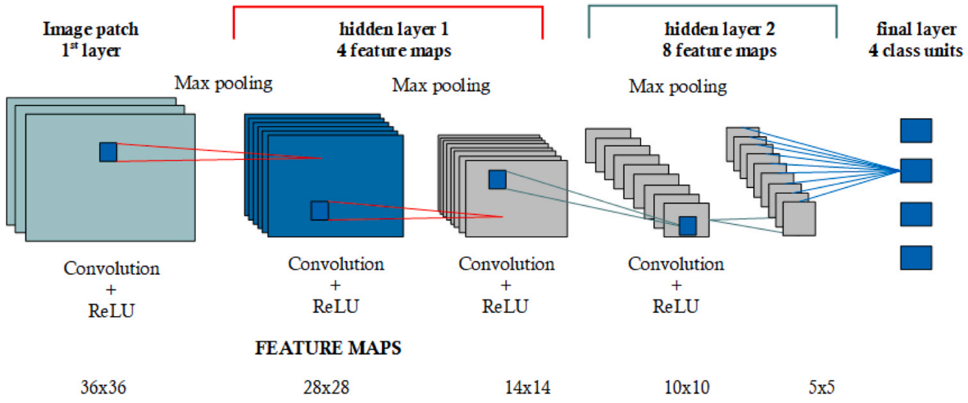


Fig. 2. Typical CNN architecture.

2. Background study

Numerous researchers have utilized pre-trained CNNs for the purpose of object and image recognition. The CNN architecture consists of convolutional, pooling, fully connected, and non-linear layers. The commonly used non-linear operations include sigmoid, tanh, and ReLU. To classify a dataset of 1.2 million images into 1000 classes, a deep CNN was trained using the ImageNet and Alexnet pre-trained models [31]. He et al. [32] developed a residual learning framework called ResNet, which was trained on the ImageNet dataset and emerged as the winner in the 2015 ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) classification competition. Howard et al. [33] created a mobile and embedded vision application model known as MobileNet, which achieved exceptional performance in various tasks such as face recognition, fine-grained categorization, item identification, and extensive geolocalization, particularly in ImageNet classification. GoogLeNet, proposed in ILSVRC 2014, is another pre-trained CNN model whose architecture was assessed for image detection and classification [34]. Additionally, there are several other pre-trained CNN models available for image classification, such as InceptionV3 [34], VGG-16,19 [35], DenseNet [36], ResNet [37], Inception-ResNet [38], DarkNet-19 [39], Xception [40], EfficientNet [41], ShuffleNet [42], and SqueezeNet [43].

The inspection and analysis of wooden structures are crucial due to their widespread use as the primary material in many large public buildings. Computer vision-based inspection systems have gained significant attention for quality control purposes. The failure of timber structures is often attributed to the propagation of cracks along the direction of the grain, accounting for 75% of failures [44]. Magnière et al. [1] carried out an investigation to gather information about the most frequent characteristics of cracked timber structures. The research presented an analysis of the key features of timber elements and crack distributions that are commonly observed. These features were utilized to establish a numerical model that aimed to understand the effects of cracks on the stiffness and load-bearing capacity of timber beams [1].

He et al. [45] used deep CNNs for feature extraction and detection of wood defects. The CNN used in the study consisted of an input layer, a softmax layer, 4 convolutional layers, 4 max-pooling layers, 3 fully connected layers, and an output layer. It was trained using DCNN TensorFlow. A dataset of 600 slices of red and camphor pine wood was created for wood defect analysis, which was divided into training, validation, and testing groups. The training dataset included 42,750 knots, 40,050 cracks, and 41,200 mildew stains, while the achieved overall accuracy reached 99.13% [45]. Kamal et al. [2] attempted the classification of wood defects using laws texture energy measures and supervised learning approach. Texture feature extraction was performed using the gray level co-occurrence matrix and laws texture energy measures, while a feed-forward back-propagation neural network was used as classifier. The Mean Square Error (MSE) for training data was found to be 0.0718 when laws texture energy measures based features were used (with 90.5% overall average classification accuracy), as compared to gray level co-occurrence matrix based input features where MSE for training data was found to be 0.10728 (with 84.3% accuracy) [2].

In their study, Ma et al. [46] created the first collection of image data capturing cracks in historic timber constructions. They employed the YOLO model to successfully recognize and identify cracks in ancient timber architecture. This breakthrough enabled the development of a new theory for the effective operation and maintenance of historic timber buildings [46]. Similarly, Cabaleiro et al. [47] introduced a crack detection system specifically designed for timber beams. They utilized LiDAR data to sample the beams and demonstrated that their method could automate the detection and assessment of cracks wider than 3 mm. The point clouds used in their analysis had an average resolution of less than 1 mm.

Munawar et al. [48] conducted a comprehensive review of studies on crack detection using machine learning and image processing techniques. They reviewed 30 research articles published in reputable journals and conferences within the last decade. Maniat et al. [49] conducted a detailed examination and comparison of various methodologies for automated crack detection. They investigated the potential of these methods to identify cracks accurately. In their study on pavement quality evaluation, image patches acquired from Google Street View (GSV) and categorized by CNN were compared to those from a commercial visual inspection company. The comparison showed that utilizing GSV images for assessing pavement condition can be practical and efficient. Furthermore, the tested network was also applied to new data, demonstrating how CNN can assist in categorizing pavement photos into multiple crack categories.

The study of Sony et al. [50] focuses on addressing the challenges associated with post-disaster inspections, which are often difficult to complete promptly due to their labor-intensive, risky, and time-consuming nature. The researchers have conducted a systematic review of contemporary structural health monitoring (SHM) research that utilizes CNNs to overcome these challenges. The use of modern CNN-based designs and advanced SHM technologies, such as cameras, drones, and robots, enables infrastructure owners to accurately and autonomously detect various types of damage in different structures [50]. In the study by Cha and Choi [51], CNN was trained on a dataset of 40,000 images with a resolution of 256 $\times$ 256 pixels to detect concrete cracks. The results were promising with accuracy levels reaching 98%.

In their study, Zhang et al. [52] utilized sliding windows in conjunction with a trained CNN to analyze images with resolutions exceeding 256 $\times$ 256 pixels. The flexibility and robustness of this methodology were assessed by testing it on a set of 55 photos, each with a resolution of 5888 $\times$ 3584 pixels. It is important to note that these photos were obtained from a distinct structure that was not utilized during the training or validation stages. Different types of pavement distress, including linear or longitudinal cracks, network cracks, fatigue cracks or potholes, patches, and pavement markings, were taken into consideration. The proposed method demonstrated an accuracy and detection rate of 83.8% when classifying the images in the test set. This performance is particularly encouraging when compared to existing research in this field [52].

Chaiyasarn et al. [53] developed a crack detection technique that utilizes deep CNNs and Support Vector Machines (SVMs). In order to enhance the classification performance, CNN is used to extract features from digital images, while SVM acts as a substitute classifier for a softmax layer. The researchers collected images of masonry fissures from historical locations using digital photography and an unmanned aerial vehicle. The proposed system was trained and validated using these digital photos. The results demonstrate that the model combining CNN and SVM outperforms the one using only CNN, with a detection accuracy of approximately 86% in the validation images [53]. Ehtisham et al. [54] adopted four pre-trained CNN models to classify cracks and their orientations. They obtained a dataset of 32,000 images from available web resources, with 75% of the photographs displaying cracks and 25% not showing cracks. Among the models used, the ResNet50 model exhibited the best average training time of 8208 s and achieved a classification accuracy of 86.2%, surpassing the performance of other models.

In their study, Ahmed et al. [55] utilized a dataset consisting of 48,000 images and employed Resnet50 for the purpose of crack detection. Resnet50 was trained on this dataset and achieved an impressive accuracy rate of 99.8%. The dataset was divided equally into two classes: crack and normal (non-crack), with 24,000 images assigned to the crack group and another 24,000 to the normal group. In contrast, Mohamed et al. [56] proposed a feedforward backpropagation NN scheme. Their research demonstrated that the NN yielded an average error rate of 18.81%. This is a significant improvement of 10% compared to the prior learning technique (updated 3D Make toolkit) used for depth recovery.

Özgenel and Sorguç [57] conducted a comprehensive analysis of the performance of popular pre-trained networks, examining factors such as the size of the training dataset, network depth, number of training epochs, and adaptability to different construction materials. The objective of their study was to provide valuable insights for new researchers and highlight important considerations when using CNNs for crack detection tasks. Guzmán-Torres et al. [58] investigated the impact of different DL topologies, network depths, regularization approaches, and transfer learning methods on the performance of learning strategies. They proposed a novel transfer learning technique to enhance the accuracy of a micro- and macro-crack DL classifier in their research. In a study conducted by Qayyum et al. [59], the performance of three models, namely GoogLeNet, MobileNet-V2, and Inception-V3, was evaluated. These models were trained using 48,000 images to classify them as cracked (C) or uncracked (UC), and 24,000 images to classify them as diagonal crack (DC), horizontal crack (HC), or vertical crack (VC). The accuracy rates for GoogLeNet, MobileNet-V2, and Inception-V3 in categorizing C and UC were found to be 95.7%, 96.3%, and 97.2%, respectively. On the other hand, when using DC as the positive case, the accuracy rates for GoogLeNet, MobileNet-V2, and Inception-V3 in categorizing DC, HC, and VC were determined to be 84%, 91%, and 92% respectively.

In their study, Mishra et al. [60] utilized sensor data and applied their methodology to various tasks including predicting the early-stage compressive strength of concrete, determining when to remove formwork, monitoring vibration and curing quality, identifying cracks in buildings and potholes on roads, assessing the quality of construction, diagnosing corrosion, detecting different types of damage, and evaluating seismic risk. Munawar et al. [61] conducted experiments using the deep hierarchical CNN architecture, comparing the GF technique, Baseline method, deep crack BN, deep crack GF, and SegNet. The results showed that the GF technique outperformed the other methods, with a global accuracy of 99%, class average accuracy of 93.9%, mean intersection of union overall classes (IoU) of 87.9%, precision of 83.8%, recall of 87.9%, and F-score of 85.8%.

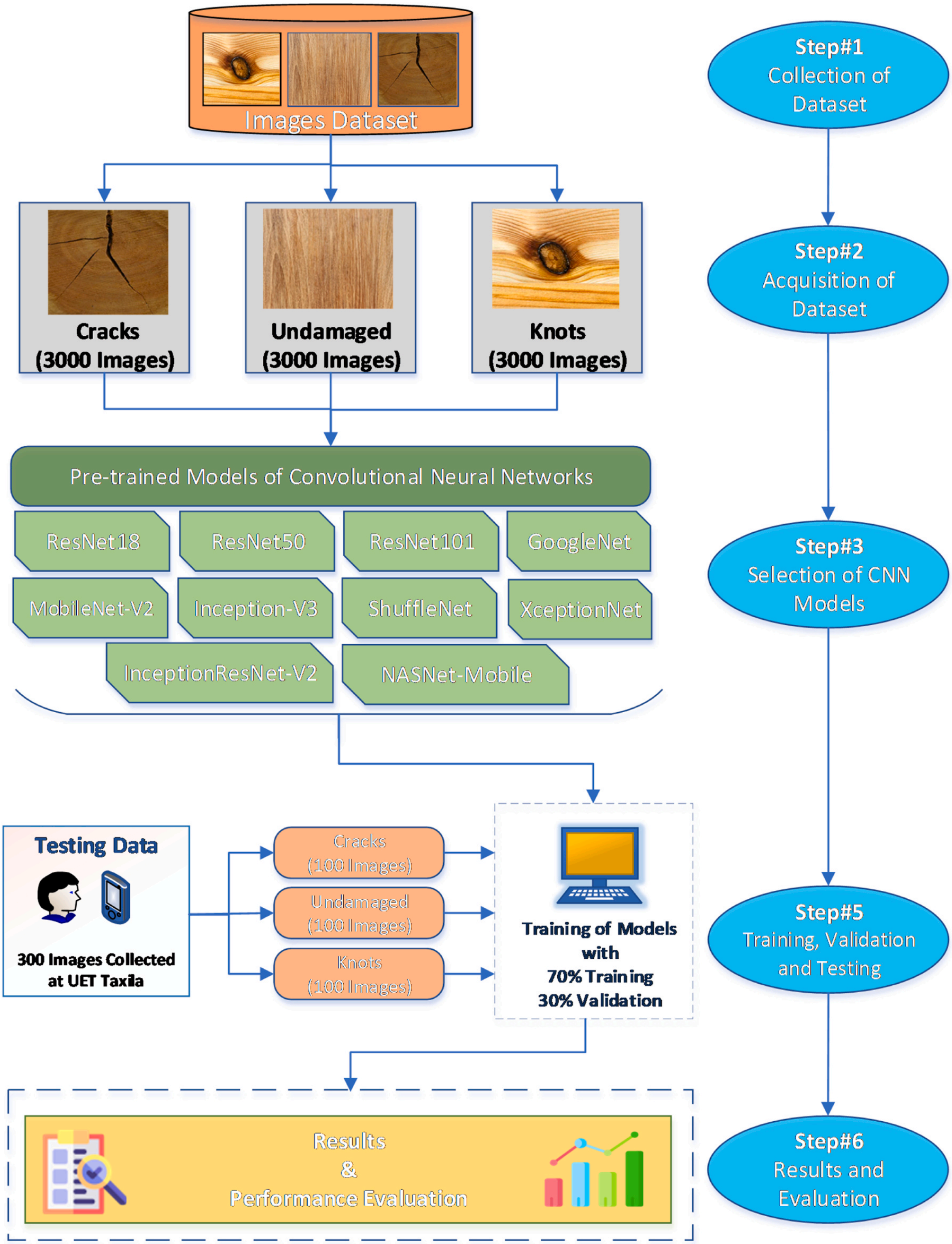


Fig. 3. Flowchart of the methodology.

Gopalakrishnan et al. [62] utilized a deep CNN (DCNN) that was trained on the ImageNet database, which contains a vast amount of images. The purpose of their research was to automatically identify cracks in images of hot-mix asphalt and Portland cement concrete surfaced pavement. These images also contained other irregularities and defects aside from cracks. The researchers achieved the best results by using a single-layer neural network classification algorithm, with the “Adam” optimizer, that was trained on pre-trained ImageNet VGG-16 DCNN parameters. They also fine-tuned and tested AlexNet, comparing it to a current network based on the suggested architecture and sampling and training technique proposed by Wang et al. [63]. The suggested framework demonstrated exceptional results in terms of detection accuracy, precision, and the F1-score measure due to its nonlinear learning capability, training dataset integrity, and active learning method.

3. Methodology

This study utilizes a six-step procedure for classifying and predicting wooden defects, as demonstrated in Fig. 3. The initial step involves collecting the dataset, which is then sorted into the appropriate categories. Subsequently, one of the ten pre-trained CNN models is chosen. This selected model is trained and validated using the same input datasets and subsequently tested using images obtained from a wooden structure in Taxila, Pakistan. Finally, a confusion matrix is generated for these pre-trained models to compare their performance in predicting defects in wooden structures. Additionally, the processing time for each model is recorded to facilitate a comparative analysis.

3.1. Pre-trained CNN models

CNN models are constructed using multiple deep layers and interconnected convolutional blocks in a specific order, as shown in Fig. 4. Currently, there exists a wide variety of pre-trained CNN models available in the literature. The primary advantage of utilizing pre-trained models is their capacity to learn comprehensive and meaningful visual feature representations through extensive training with large datasets. These pre-trained models have the ability to capture complex patterns and features in images. By utilizing pre-trained models, researchers and practitioners can leverage the knowledge and feature extraction capabilities gained during training, even when working with limited task-specific training data. This approach is particularly valuable in situations where it is not feasible or practical to obtain a large labeled dataset for training from scratch.

When utilizing a pre-trained CNN model, the typical procedure involves the following steps: (i) Model Initialization, (ii) Feature Extraction, (iii) Fine-tuning or Transfer Learning, and (iv) Training and Evaluation. The Model Initialization comprises the importation of the model, including its layers, configurations, and the weights acquired during the pre-training phase. The feature extraction phase entails removing the final fully connected layers, thereby considering the output of the preceding layers as a high-level representation of the input images’ features. During the training and evaluation step, the adapted or fine-tuned model is trained using a new dataset specific to the task at hand. The training process involves inputting the dataset into the model, calculating the loss, and updating the model’s weights through techniques like backpropagation. Subsequently, the model is evaluated on a separate validation or test dataset to assess its performance. This is done using various evaluation metrics such as accuracy, precision, recall, or F1 score.

In the present study, some of the most widely used pre-trained models are employed, namely (i) ResNet18, (ii) ResNet50, (iii) ResNet101, (iv) ShuffleNet, (v) GoogLeNet, (vi) Inception-V3, (vii) MobileNet-V2, (viii) Xception, (ix) Inception-ResNet-V2, and (x) NASNet-Mobile. These pre-trained models are acquired from MATLAB’s Deep Network Designer App (version 2020a) and have been further trained on the wooden defects’ dataset.

The concept of residual learning is developed by the residual neural network (**ResNet**) models. These models are trained to learn residual functionality by analyzing the inputs towards each layer of the network. Instead of assuming that each layer perfectly matches the fundamental mapping as planned, residual nets allow multiple layers to match a residual mapping. These residual blocks are layered together to construct networks, such as ResNet-50 which consists of 50 deep layers.

ShuffleNet is specifically developed for usage on portable computers with limited computing power. The design combines channel shuffling and parameter group convolution to increase efficiency while maintaining accuracy. The Inception design serves as the foundation of GoogLeNet. It utilizes Inception modules that enable the network to choose between different convolutional filter sizes in each block. These modules are stacked on top of each other within the Inception model’s network. Periodically, the model reduces the

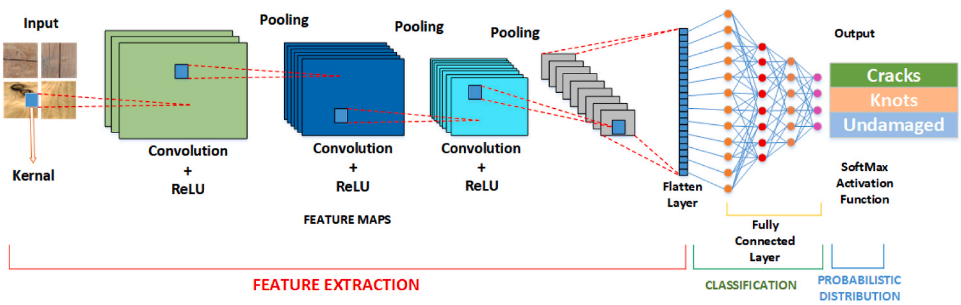


Fig. 4. CNN’s deep layers.

Table 1
Characteristics of the ten pre-trained CNN models.

Pre-trained Model of CNN	No. of Deep Layers (DL)	Syntax (net =)	Size (Mb)	Parameters (Millions)	Input Image resolution
ResNet18	18	resnet18	44	11.7	224 × 224 × 3
ResNet50	50	resnet50	96	25.6	224 × 224 × 3
ResNet101	101	resnet101	167	44.6	224 × 224 × 3
ShuffleNet	50	shufflenet	5.4	1.4	224 × 224 × 3
GoogLeNet	22	googlenet	27	7.0	224 × 224 × 3
Inception-V3	48	inceptionv3	89	23.9	299 × 299 × 3
MobileNet-V2	53	mobilenetv2	13	3.5	224 × 224 × 3
Xception	71	xception	88	22.9	299 × 299 × 3
Inception-ResNet-V2	164	inceptionresnetv2	215	55.9	299 × 299 × 3
NASNet-Mobile	58	nasnetmobile	23	5.3	224 × 224 × 3

grid's resolution through max-pooling layers with a stride of two. **Inception-V3**, a member of the Inception family, uses label smoothing, factorized 7 × 7 convolutions, and an auxiliary classifier to transport label data to deeper levels of the network.

MobileNetV2 is a CNN architecture that aims to improve performance on mobile devices. The design of this architecture focuses on the connections between the residual layers of a backward residual architecture. To introduce non-linearity, lightweight depth-wise convolutions are applied to the features in the intermediate expansion layer. MobileNetV2 starts with a 32-filter fully convolutional layer, followed by 19 layers of residual constraints [64].

The specifications of the CNN models used in the present study are presented in Table 1. Each model is capable of classifying images into 1000 categories. The resolution of the input image is determined by the requirements of the pre-trained models used in the study. Each pre-trained model has its own default input size, which is typically determined by its architecture and training process. In our study, seven of the models (ResNet18, ResNet50, ResNet101, ShuffleNet, GoogLeNet, MobileNet-V2, NASNet-Mobile) use a resolution of 224 × 224 × 3, while three of them (Inception-V3, Xception, Inception-ResNet-V2) use a resolution of 299 × 299 × 3.

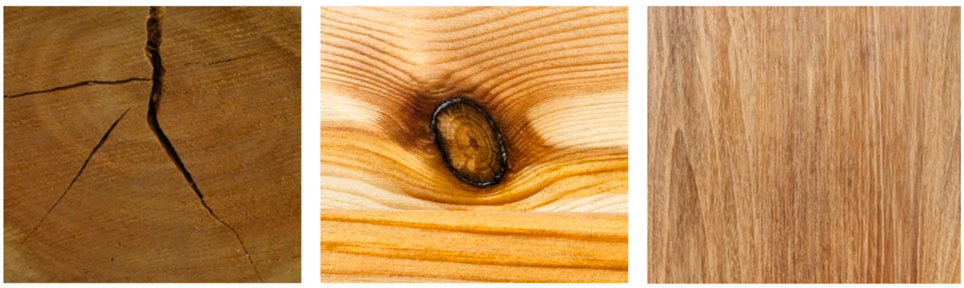
4. Experimental study

4.1. Acquisition of the dataset

The dataset used for training and validation in this study consists of images that were collected from various online sources [65]. The smaller testing dataset, on the other hand, was obtained by the authors from a real-world wooden structure. The dataset consists of a total of 9000 RGB images, which have been categorized into three groups based on the type of defects present. These defects are classified into three categories: (i) cracks, (ii) knots, and (iii) undamaged, as shown in Fig. 5. Each category has a dataset of 3000 images, with each image having a resolution of 3472 × 4624 pixels. Additionally, there is a separate dataset for each category consisting of 100 images, which is used for testing purposes.

4.2. Training, validation and testing

In order to train and validate the CNN pre-trained models, a total of 9000 RGB images are utilized. These images are divided into three equal categories, with a training and validation ratio of 70% and 30% respectively. For testing purposes, an additional set of 300 images is employed, with 100 images per class. It is important to note that the same training, validation, and testing datasets are used for all models. The CNN models utilized undergo training and testing using consistent parameters. These parameters include an epoch value of 3, a minimum batch size of 10, and an initial learning rate of 0.003.



(a) (b) (c)

Fig. 5. Example wood images and their classifications: (a) Cracks, (b) Knots, (c) Undamaged.

5. Results and discussion

The current study assesses ten CNN models used for predicting defects in wooden structures. These models, namely ResNet18, ResNet50, ResNet101, ShuffleNet, GoogLeNet, Inception-V3, MobileNet-V2, Xception, Inception-ResNet-V2, and NASNet-Mobile, are based on their implementation in MATLAB [66]. Each CNN model is trained, validated, and tested using a large dataset. The study compares the accuracy, precision, recall, F1-score, and confusion matrix of the models, while also recording the processing time required for training. The training involves three different classifications: cracks, knots, and undamaged. All CNN models are trained on a desktop workstation equipped with a 10th generation Core i7 CPU, 32 GB DDR4 RAM and a 12 GB 3060 graphics card.

In the context of machine learning classifications, the accuracy performance of a model can be quantified by the ratio of correctly predicted values to the total number of predictions made. This can be further explained as the ratio of true positives and true negatives to the sum of all positive and negative predictions generated by the model, as stated in Eq. (1).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

where *TP* is a true positive prediction, *TN* is a true negative prediction, *FP* is a false positive prediction, and *FN* is a false negative prediction value [58].

Table 2 presents the accuracies achieved by the trained networks in each of the three categories. The Inception-V3 model exhibited the highest accuracies, with 99% and 98% for crack and undamaged image predictions, respectively. The Inception-ResNet-V2 model achieved a maximum accuracy of 99% in knot prediction. On the other hand, the MobileNet-V2 model demonstrated the lowest accuracies, with a crack accuracy and undamaged image prediction accuracy of 83%. The ShuffleNet model had the lowest accuracy in classifying knot images, with a rate of 93%.

The precision of the model's performance is defined as the accuracy of all positive prediction values. In other words, the precision metric quantifies the percentage of accurately predicted positive labels as follows [58]:

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

The recall parameter of the model's performance is defined as the true positive prediction rate. In other words, the model uses the recall score to evaluate its accuracy in correctly predicting positive values compared to the actual positive values. The calculation for the recall score is provided as [58]:

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

Table 3 displays the precision values for the trained networks across the three categories. The ResNet18, ResNet101, and Inception-V3 models exhibit the highest precision of 99% in predicting cracks. Similarly, the Inception-ResNet-V2 model shows a precision of 99% in predicting knots and undamaged images. On the other hand, the MobileNet-V2 model demonstrates the lowest precision of 67% in predicting crack images. For the prediction of knots and undamaged images, the ShuffleNet model exhibits the lowest precision values of 89% and 79%, respectively.

Table 4 shows all the recall values for the trained networks for each of the three classification categories. For the prediction of knots, the maximum recall parameter is obtained as 98%, achieved by the Inception-ResNet-V2 model. The maximum (best) recall value for the prediction of undamaged images is 97%, achieved by the ResNet101, ResNet18, and Inception-V3 models. On the other hand, the maximum recall value for the prediction of crack images is 99%, achieved by Inception-V3 and Inception-ResNet-V2. The minimum recall value for the prediction of undamaged images is 50% (by MobileNet-V2 model); for crack images, it is equal to 81% (by NASNet-Mobile model), and for knots images, it is equal to 91% (by Xception and NASNet-Mobile models).

The F1-score of the model is defined as the harmonic mean of the recall and precision values for the predictions. The F1-score is a metric used to evaluate the accuracy of a model, considering both precision and recall equally. The calculation for the F1-score is provided as follows [45]:

Table 2
Accuracies of predictions related to (i) cracks, (ii) knots, and (iii) undamaged state.

ID	CNN's Models	Accuracy (%)		
		Cracks	Knots	Undamaged
1	ResNet18	96	97	95
2	ResNet50	96	97	95
3	ResNet101	98	97	97
4	ShuffleNet	93	93	86
5	GoogLeNet	95	97	94
6	Inception-V3	**99**	97	**98**
7	MobileNet-V2	83	97	83
8	Xception	91	94	91
9	Inception-ResNet-V2	97	**99**	97
10	NASNet-Mobile	88	94	89

Table 3
Precision in predicting cracks, knots, and undamaged state.

ID	CNN Model	Precision (%)		
		Cracks	Knots	Undamaged
1	ResNet18	**99**	96	89
2	ResNet50	98	96	95
3	ResNet101	**99**	96	93
4	ShuffleNet	86	89	79
5	GoogLeNet	93	95	92
6	Inception-V3	**99**	96	97
7	MobileNet-V2	67	96	96
8	Xception	88	91	85
9	Inception-ResNet-V2	93	**99**	**99**
10	NASNet-Mobile	84	91	83

Table 4
Recalls on predicting cracks, knots, and undamaged state.

ID	CNN Model	Recall (%)		
		Cracks	Knots	Undamaged
1	ResNet18	90	96	**97**
2	ResNet50	91	96	90
3	ResNet101	95	96	**97**
4	ShuffleNet	83	89	81
5	GoogLeNet	93	96	91
6	Inception-V3	**99**	96	**97**
7	MobileNet-V2	90	96	50
8	Xception	85	91	88
9	Inception-ResNet-V2	**99**	**98**	93
10	NASNet-Mobile	81	91	85

Table 5
F1-Score of cracks, knots, and undamaged predictions.

ID	CNN's Models	F1-score (%)		
		Cracks	Knots	Undamaged
1	ResNet18	94	96	93
2	ResNet50	94	96	96
3	ResNet101	97	96	95
4	ShuffleNet	84	89	80
5	GoogLeNet	93	96	91
6	Inception-V3	**99**	96	**97**
7	MobileNet-V2	80	96	66
8	Xception	86	91	87
9	Inception-ResNet-V2	96	**98**	96
10	NASNet-Mobile	82	91	84

The model's F1-score is specified as the harmonic mean of the recall and precision of the prediction values. The F1-score is a model evaluation measure that assesses a model's accuracy by giving Precision and Recall equal weight and is calculated as follows [58]:

$$\mathrm{F1-score} = \frac{2\left(\frac{TP}{TP+FP}\right) \times \left(\frac{TP}{TP+FN}\right)}{\left(\frac{TP}{TP+FP}\right) + \left(\frac{TP}{TP+FN}\right)} \tag{4}$$

Table 5 shows all the F1-score values for the trained networks for each of the three categories. The maximum value of the F1-score for the prediction of cracks is 99%, exhibited by the Inception-V3 model. The maximum F1-score for knots prediction is 98%, achieved by the Inception-ResNet-V2 model, and for the prediction of undamaged images, the corresponding maximum F1-score is 97%, by the Inception-V3 model. The minimum overall F1-score is 66% by the MobileNet-V2 model for the prediction of undamaged images. For the cases of cracks and knots, the corresponding minimum F1-score values are equal to 80% and 89%, by the MobileNet-V2 and ShuffleNet models, respectively.

Fig. 6 illustrates the performance of pre-trained models in predicting 100 test images in each wooden structure category. These images were not used in any training process and are therefore new to the models. The Inception-V3 model achieved the highest overall accuracy of 97.3% when predicting cracks, knots, and undamaged images. Conversely, the MobileNet-V2 model had the lowest overall accuracy of 79.7%. In terms of predicting knots, the Inception-ResNet-V2 model demonstrated the best overall performance, while the

Fig. 6. Confusion matrix, for each model.

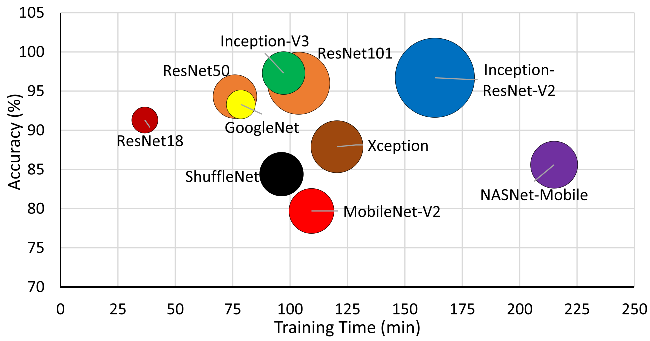


Fig. 7. Accuracy, training time, and number of deep layers for the CNN models.

Inception-V3 model produced the most accurate predictions for cracks.

Fig. 7 presents a graphical representation of the correlation between the accuracy, training time, and deep layers of ten pre-trained CNN models. In the graph, the training time is in the horizontal axis, the accuracy on the vertical axis and the size of the circle is related to the number of deep layers for the CNN models. Among these models, the Inception-ResNet-V2 stands out with a maximum of 164 deep layers, a training time of 163 min, and an overall accuracy of 96.7%. On the other end of the spectrum, the NASNet-Mobile model required the longest training time, consuming 215 min. It had 58 deep layers and achieved an overall accuracy of 85.6%. The Inception-V3 model, with only 48 deep layers, demonstrated the highest overall accuracy of 97.3%, and it had a training time of 97 min. In contrast, the MobileNet-V2 model had the lowest accuracy of 79.7%. This model consisted of 53 deep layers and required 109 min for training. ResNet18 with 18 deep layers had the lowest training time of 37 min with an overall accuracy of 91.3%.

One randomly selected image from each class is chosen to demonstrate the performance of each trained CNN model on a practical example. The results indicate that the trained models of ResNet50, ResNet101, Inception-V3, GoogLeNet, Xception, and Inception-ResNet-V2 accurately predicted the images according to their respective classes. Fig. 8A and B depict that the trained models of ResNet18, MobileNet-V2, and NASNet-Mobile exhibit confusion in predicting the undamaged image, while ShuffleNet and MobileNet-V2 display confusion in predicting the knot image. However, all models correctly predicted the crack image.

6. Conclusions

The purpose of this research is to conduct a comparative study on the classification and prediction of defects in wooden structures. To achieve this, ten pre-trained CNN models, including ResNet18, ResNet50, ResNet101, Inception-V3, GoogLeNet, ShuffleNet, InceptionResNet-V2, MobileNet-V2, Xception, and NASNet-Mobile, were further trained, validated, and tested on images with cracks, knots, and undamaged areas. A dataset containing 9000 images was used for training and validation purposes. The models were then independently tested on 300 images (100 images per class), and their performance was evaluated using the confusion matrix, accuracy, precision, recall, F1-score, overall performance, and training time. The InceptionNet-V3 model, with 48 deep layers, outperformed the other models with an overall accuracy rate of 97.3% and a training time of 97 min. The ShuffleNet model, with a small size of 5.4 MB and 50 deep layers, achieved a marginal overall accuracy rate of 84.4%, making it suitable for low-configured machines. The ResNet50 model had the highest accuracy of 94.3% and the shortest training time of 76 min. To demonstrate the results, one image from each category was randomly selected for testing the models. The outcomes showed that the models ResNet50, ResNet101, Inception-V3, GoogLeNet, Xception, and Inception-ResNet-V2 correctly classified images according to their respective classes.

Accurately identifying and classifying defects enables manufacturers to eliminate or address faulty wood prior to its use in construction or manufacturing processes. This endeavor is vital for upholding the overall quality and reliability of wood products, thereby reducing the risk of structural failures or product malfunctions. Additionally, early identification and classification of wood defects during production contribute to cost reduction efforts. By removing defective wood or applying appropriate treatments, manufacturers can avoid the utilization of subpar or compromised materials, thereby minimizing waste and sidestepping expensive rework or replacements. Manual inspection of wood defects can be a time-consuming and labor-intensive task. However, by leveraging a CNN model, the classification of defects can be automated, resulting in a significant reduction in the time and effort required for inspection. This work enables faster and more efficient evaluation of wood products, facilitating increased production throughput and quicker response times. After the classification of defects, specific characteristics such as crack width, orientation, knot area, etc., can be computed, which could be explored in future research.

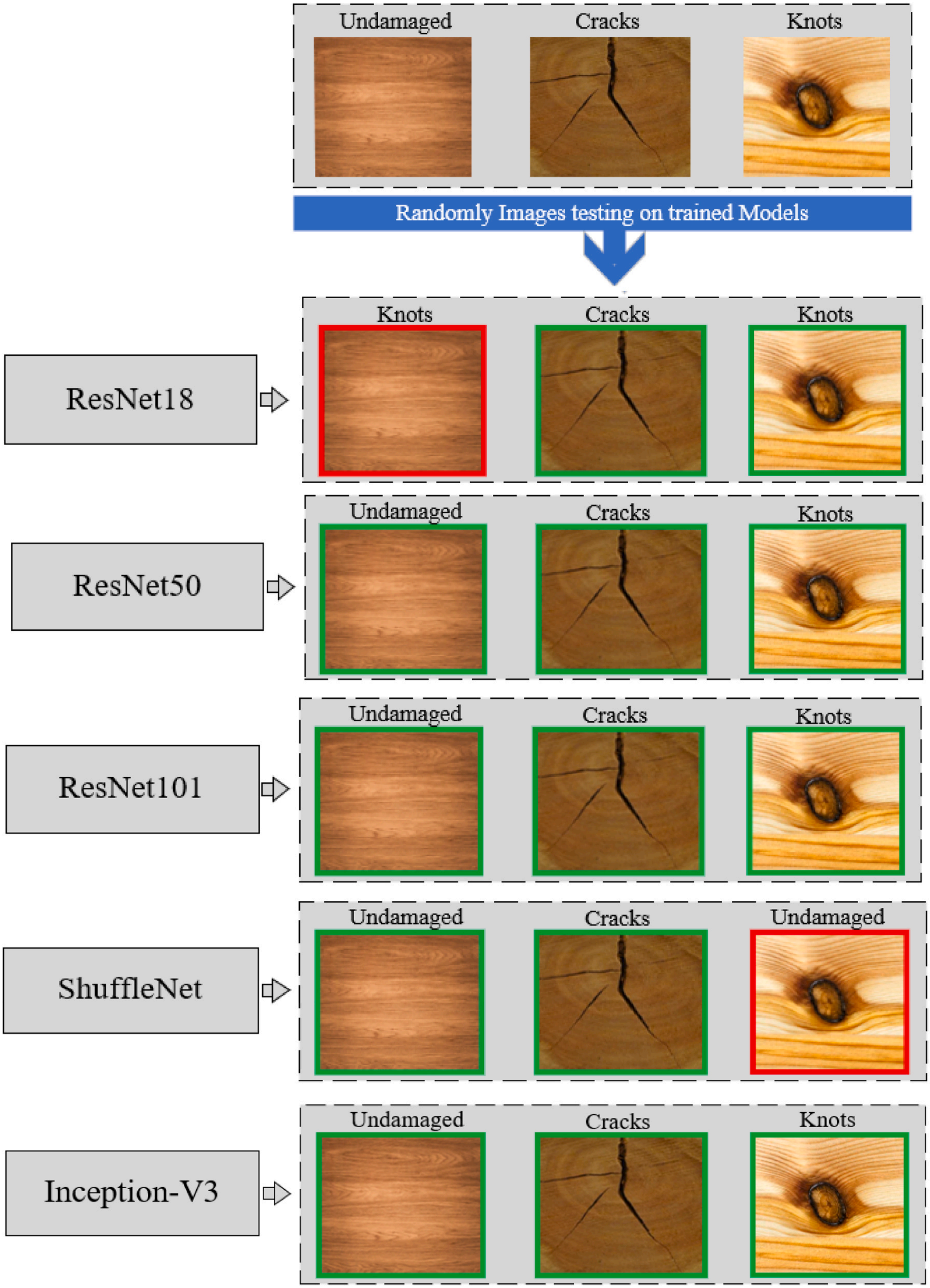


Fig. 8. A. CNN model results of randomly selected testing images (ResNet18, ResNet50, ResNet101, ShuffleNet and Inception-V3). B. CNN model results of randomly selected testing images (GoogleNet, MobileNet-V2, Xception, Inception-ResNet-V2 and NASNet-Mobile).

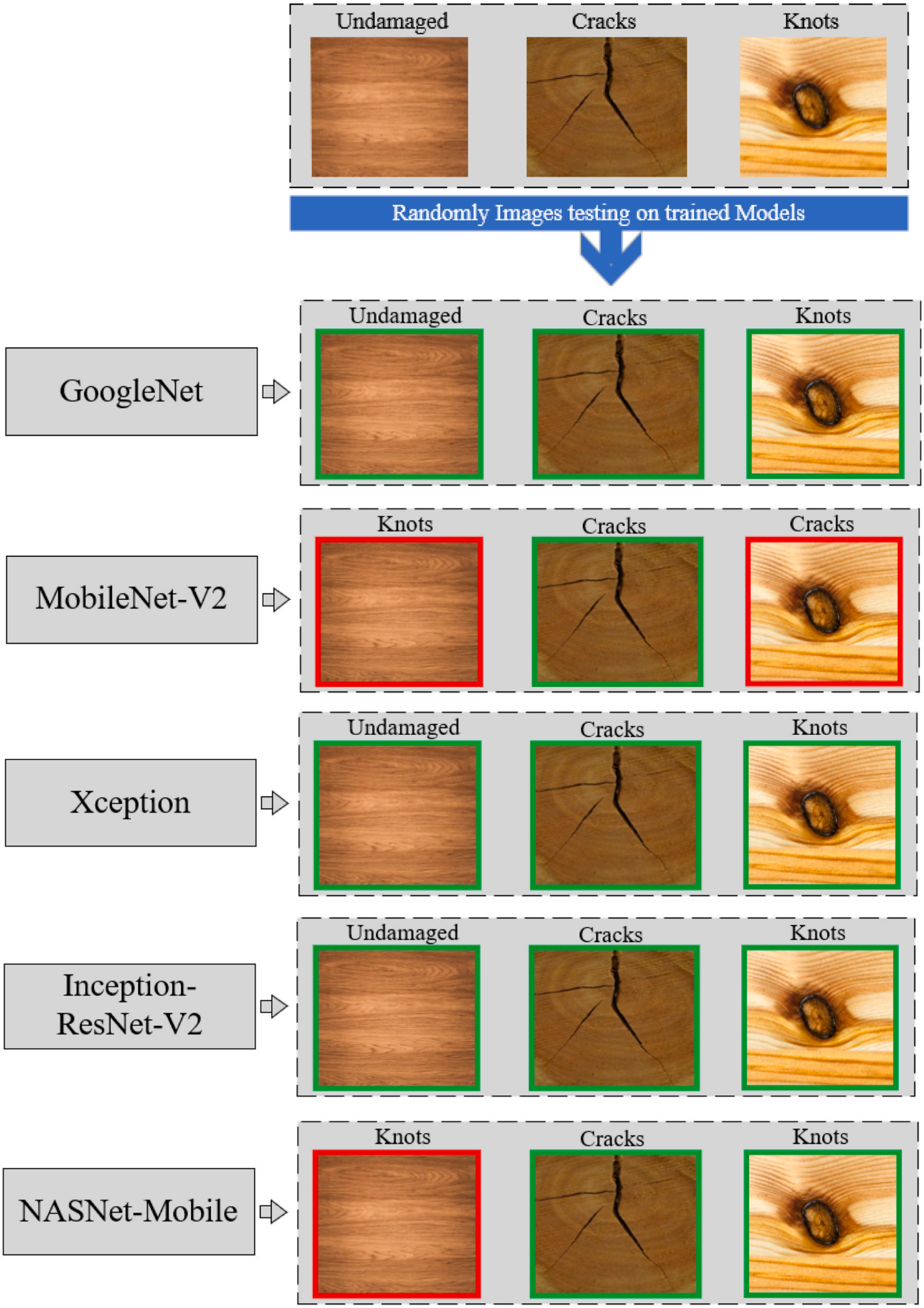


Fig. 8. (*continued*).

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests. Afaq Ahmad reports that financial support was provided by Punjab Higher Education Commission.

Data Availability

Data will be made available on request.

Acknowledgment

The funding for this study, with grant number PHEC/ARA/PIRCA/20529/22, is sponsored by the Punjab Higher Education Commission (PHEC) in Pakistan. The authors express their profound gratitude to the commission for its support. Open Access funding has been provided by the Qatar National Library.

References

[1] Magnière, N., S. Franke, B. Franke, Investigation on elements presenting cracks in timber structures, in *World Conference on Timber Engineering (WCTE 2014)*. 2014: Quebec, Canada.
[2] K. Kamal, R. Qayyum, S. Mathavan, T. Zafar, Wood defects classification using laws texture energy measures and supervised learning approach, Adv. Eng. Inform. 34 (2017) 125–135, https://doi.org/10.1016/j.aei.2017.09.007.
[3] Mergny, E., R. Mateo, M. Esteban, T. Descamps, and P. Latteur, *Influence of cracks on the stiffness of timber structural elements*, in World Conference on Timber Engineering (WCTE *2016)*. 2016: Vienna, Austria.
[4] N.D. Lagaros, V. Plevris, Artificial Intelligence (AI) Applied in Civil Engineering, Appl. Sci. 12 (15) (2022), https://doi.org/10.3390/app12157595.
[5] N.D. Lagaros, V. Plevris (eds.), Artificial Intelligence (AI) Applied in Civil Engineering, 698, MDPI, 2022, https://doi.org/10.3390/books978-3-0365-5084-8.
[6] Zafar, A., J. Mir, V. Plevris, and A. Ahmad, *Machine Vision based Crack Detection for Structural Health Monitoring using Haralick Features*, in 2nd Conference on Sustainability in Civil Engineering (CSCE'20). 2020: Capital University of Science & Technology, Islamabad, Pakistan.
[7] O. Avci, O. Abdeljaber, S. Kiranyaz, Structural Damage Detection in Civil Engineering with Machine Learning: Current State of the Art, Springer International Publishing, Cham, 2022, https://doi.org/10.1007/978-3-030-75988-9_17.
[8] M. Georgioudakis, V. Plevris, A combined modal correlation criterion for structural damage identification with noisy modal data, Adv. Civ. Eng. 2018 (3183067) (2018) 20, https://doi.org/10.1155/2018/3183067.
[9] G. Solorzano, V. Plevris, Computational intelligence methods in simulation and modeling of structures: a state-of-the-art review using bibliometric maps, Front. Built Environ. 8 (2022), https://doi.org/10.3389/fbuil.2022.1049616.
[10] Solorzano, G. and V. Plevris, *ANN-based surrogate model for predicting the lateral load capacity of RC shear walls*, in *8th European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS 2022)*. 2022: Oslo, Norway. DOI: ⟨https://doi.org/10.23967/eccomas.2022.050⟩.
[11] G. Solorzano, V. Plevris, DNN-MLVEM: a data-driven macromodel for rc shear walls based on deep neural networks, Mathematics 11 (10) (2023) 2347, https://doi.org/10.3390/math11102347.
[12] G. Solorzano, V. Plevris, An open-source framework for modeling RC shear walls using deep neural networks, Adv. Civ. Eng. 2023 (2023) 7953869, https://doi.org/10.1155/2023/7953869.
[13] M. Georgioudakis, V. Plevris, Response spectrum analysis of multi-story shear buildings using machine learning techniques, Computation 11 (7) (2023) 126, https://doi.org/10.3390/computation11070126.
[14] S.O. Abioye, et al., Artificial intelligence in the construction industry: a review of present status, opportunities and future challenges, J. Build. Eng. 44 (2021), 103299, https://doi.org/10.1016/j.jobe.2021.103299.
[15] X. Wang, R.K. Mazumder, B. Salarieh, A.M. Salman, A. Shafieezadeh, Y. Li, Machine learning for risk and resilience assessment in structural engineering: progress and future trends, J. Struct. Eng. 148 (8) (2022) 03122003, https://doi.org/10.1061/(ASCE)ST.1943-541X.0003392.
[16] M. Imran Waris, V. Plevris, J. Mir, N. Chairman, A. Ahmad, An alternative approach for measuring the mechanical properties of hybrid concrete through image processing and machine learning, Constr. Build. Mater. 328 (126899) (2022), https://doi.org/10.1016/j.conbuildmat.2022.126899.
[17] M. Nikoo, G. Hafeez, G. Doudak, V. Plevris, in: V. Plevris, A. Ahmad, N.D. Lagaros (Eds.), Predicting the Fundamental Period of Light-Frame Wooden Buildings by Employing Bat Algorithm-Based Artificial Neural Network, in Artificial Intelligence and Machine Learning Techniques for Civil Engineering, IGI Global, Hershey, PA, USA, 2023, pp. 139–162, https://doi.org/10.4018/978-1-6684-5643-9.ch006.
[18] S. Seyedzadeh, F.P. Rahimian, I. Glesk, M. Roper, Machine learning for estimation of building energy consumption and performance: a review, Vis. Eng. 6 (1) (2018) 5, https://doi.org/10.1186/s40327-018-0064-7.
[19] Y. Yang, X. Zhou, Y. Liu, Z. Hu, F. Ding, Wood defect detection based on depth extreme learning machine, Appl. Sci. 10 (21) (2020) 7488, https://doi.org/10.3390/app10217488.
[20] Pa Sun, Wood quality defect detection based on deep learning and multicriteria framework, Math. Probl. Eng. 2022 (2022) 4878090, https://doi.org/10.1155/2022/4878090.
[21] Y. LeCun, Y. Bengio, G. Hinton, Deep Learn. 521 (7553) (2015) 436–444, https://doi.org/10.1038/nature14539.
[22] A. Urbonas, V. Raudonis, R. Maskeliūnas, R. Damaševičius, Automated identification of wood veneer surface defects using faster region-based convolutional neural network with data augmentation and transfer learning, Appl. Sci. 9 (22) (2019) 4898, https://doi.org/10.3390/app9224898.
[23] J.-K. Park, B.-K. Kwon, J.-H. Park, D.-J. Kang, Machine learning-based imaging system for surface defect inspection, Int. J. Precis. Eng. Manuf. -Green. Technol. 3 (3) (2016) 303–310, https://doi.org/10.1007/s40684-016-0039-x.
[24] Z. Zheng, S. Zhang, B. Yu, Q. Li, Y. Zhang, Defect inspection in tire radiographic image using concise semantic segmentation, IEEE Access 8 (2020) 112674–112687, https://doi.org/10.1109/ACCESS.2020.3003089.
[25] Z. Zheng, H. Yang, L. Zhou, B. Yu, Y. Zhang, HLU2-net: a residual u-structure embedded U-net with hybrid loss for tire defect inspection, IEEE Trans. Instrum. Meas. 70 (2021) 1–11, https://doi.org/10.1109/TIM.2021.3126847.
[26] Z. Zheng, S. Zhang, J. Shen, Y. Shao, Y. Zhang, A two-stage CNN for automated tire defect inspection in radiographic image, Meas. Sci. Technol. 32 (11) (2021), 115403, https://doi.org/10.1088/1361-6501/ac13f8.
[27] M. Zhao, Z. Zheng, Y. Sun, Y. Chang, C. Tian, Y. Zhang, MSANet: efficient detection of tire defects in radiographic images, Meas. Sci. Technol. 33 (12) (2022), 125401, https://doi.org/10.1088/1361-6501/ac85d1.
[28] J. Teuwen, N. Moriakov, Chapter 20 - Convolutional neural networks, in: S.K. Zhou, D. Rueckert, G. Fichtinger (Eds.), Handbook of Medical Image Computing and Computer Assisted Intervention, Academic Press, 2020, pp. 481–501, https://doi.org/10.1016/B978-0-12-816176-0.00025-9.
[29] Y. Tang, K. Han, J. Guo, C. Xu, C. Xu, Y. Wang, GhostNetV2: enhance cheap operation with long-range attention (arXiv:2211.12905), ArXiv e-prints (2022), https://doi.org/10.48550/arXiv.2211.12905.

[30] J. Chen, et al., Run, don't walk: chasing higher FLOPS for faster neural networks (arXiv:2303.03667), ArXiv e-prints (2023), https://doi.org/10.48550/arXiv.2303.03667.

[31] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, Commun. ACM 60 (6) (2017) 84–90, https://doi.org/10.1145/3065386.

[32] He, K., X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016. DOI:⟨https://doi.org/10.1109/CVPR.2016.90⟩.

[33] A.G. Howard, et al., MobileNets: efficient convolutional neural networks for mobile vision applications (arXiv:1704.04861), ArXiv e-prints (2017), https://doi.org/10.48550/arXiv.1704.04861.

[34] Szegedy, C., V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna. Rethinking the Inception Architecture for Computer Vision. in *2016* IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016. DOI:⟨https://doi.org/10.1109/CVPR.2016.308⟩.

[35] O. Russakovsky, et al., Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (3) (2015) 211–252, https://doi.org/10.1007/s11263-015-0816-y.

[36] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition (arXiv:1409.1556), ArXiv e-prints (2015) 14, https://doi.org/10.48550/arXiv.1409.1556.

[37] Zoph, B., V. Vasudevan, J. Shlens, and Q.V. Le, Learning Transferable Architectures for Scalable Image Recognition, in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2018, IEEE Computer Society. p. 8697–8710. DOI:⟨https://doi.org/10.1109/CVPR.2018.00907⟩.

[38] Szegedy, C., S. Ioffe, V. Vanhoucke, and A. Alemi, Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. Proceedings of the AAAI Conference on Artificial Intelligence, 2017. **31**(1) DOI:⟨https://doi.org/10.1609/aaai.v31i1.11231⟩.

[39] J. Redmon, A. Farhadi, YOLO9000: better, faster, stronger (arXiv:1612.08242), ArXiv e-prints (2016), https://doi.org/10.48550/arXiv.1612.08242.

[40] Chollet, F. *Xception: Deep Learning with Depthwise Separable Convolutions*. in *2017* IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017. DOI:⟨https://doi.org/10.1109/CVPR.2017.195⟩.

[41] M. Tan, Q.V. Le, EfficientNet: rethinking model scaling for convolutional neural networks (arXiv:1905.11946), ArXiv e-prints (2020), https://doi.org/10.48550/arXiv.1905.11946.

[42] Zhang, X., X. Zhou, M. Lin, and J. Sun. ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. in *2018* IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018. DOI:⟨https://doi.org/10.1109/CVPR.2018.00716⟩.

[43] F.N. Iandola, S. Han, M.W. Moskewicz, K. Ashraf, W.J. Dally, K. Keutzer, SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size (arXiv:1602.07360), ArXiv e-prints (2016), https://doi.org/10.48550/arXiv.1602.07360.

[44] Zhang, Y.X., Y.Q. Zhao, Y. Liu, L.Q. Jiang, and Z.W. Chen. Identification of wood defects based on LBP features. in 2016 35th Chinese Control Conference (CCC). 2016. DOI:⟨https://doi.org/10.1109/ChiCC.2016.7554010⟩.

[45] T. He, Y. Liu, Y. Yu, Q. Zhao, Z. Hu, Application of deep convolutional neural network on feature extraction and detection of wood defects, Measurement 152 (2020), 107357, https://doi.org/10.1016/j.measurement.2019.107357.

[46] J. Ma, W. Yan, G. Liu, S. Xing, S. Niu, T. Wei, Complex texture contour feature extraction of cracks in timber structures of ancient architecture based on YOLO algorithm, Adv. Civ. Eng. 2022 (2022) 7879302, https://doi.org/10.1155/2022/7879302.

[47] M. Cabaleiro, R. Lindenbergh, W.F. Gard, P. Arias, J.W.G. van de Kuilen, Algorithm for automatic detection and analysis of cracks in timber beams from LiDAR data, Constr. Build. Mater. 130 (2017) 41–53, https://doi.org/10.1016/j.conbuildmat.2016.11.032.

[48] H.S. Munawar, A.W.A. Hammad, A. Haddad, C.A.P. Soares, S.T. Waller, Image-based crack detection methods: a review, Infrastructures 6 (8) (2021) 115 (DOI).

[49] M. Maniat, C.V. Camp, A.R. Kashani, Deep learning-based visual crack detection using google street view images, Neural Comput. Appl. 33 (21) (2021) 14565–14582, https://doi.org/10.1007/s00521-021-06098-0.

[50] S. Sony, K. Dunphy, A. Sadhu, M. Capretz, A systematic review of convolutional neural network-based structural condition assessment techniques, Eng. Struct. 226 (2021), 111347, https://doi.org/10.1016/j.engstruct.2020.111347.

[51] Y.-J. Cha, W. Choi, Vision-Based Concrete Crack Detection Using a Convolutional Neural Network, Springer International Publishing,, Cham, 2017.

[52] C. Zhang, E. Nateghinia, L.F. Miranda-Moreno, L. Sun, Pavement distress detection using convolutional neural network (CNN): A case study in Montreal, Canada, Int. J. Transp. Sci. Technol. 11 (2) (2022) 298–309, https://doi.org/10.1016/j.ijtst.2021.04.008.

[53] Chaiyasarn, K., W. Khan, L. Ali, M. Sharma, D. Brackenbury, M. Dejong, Crack Detection in Masonry Structures using Convolutional Neural Networks and Support Vector Machines, in Proceedings of the 35th International Symposium on Automation and Robotics in Construction (ISARC). 2018, International Association for Automation and Robotics in Construction (IAARC). p. 118–125. DOI:⟨https://doi.org/10.22260/ISARC2018/0016⟩.

[54] Ehtisham, R., C.V. Camp, J. Mir, N. Chairman, A. Ahmad, Evaluation of Pre-trained ResNet and MobileNetV2 CNN models for the Concrete Crack Detection and Crack Orientation Classification, in 1st International Conference on Advances in Civil & Environmental Engineering (1st ICACEE-2022). 2022: University of Engineering & Technology Taxila, Pakistan.

[55] Ahmed, C.F., A. Cheema, W. Qayyum, E. Rana, A. Ahmad, Detection of Pavement cracks of UET Taxila using pre-trained model Resnet50 of CNN, in 1st International Conference on Advances in Civil and Environmental Engineering. 2022: University of Engineering & Technology Taxila, Taxila, Pakistan.

[56] Y.S. Mohamed, H.M. Shehata, M. Abdellatif, T.H. Awad, Steel crack depth estimation based on 2D images using artificial neural networks, Alex. Eng. J. 58 (4) (2019) 1167–1174, https://doi.org/10.1016/j.aej.2019.10.001.

[57] Sorguç, A.G., Performance Comparison of Pretrained Convolutional Neural Networks on Crack Detection in Buildings, in Proceedings of the 35th International Symposium on Automation and Robotics in Construction (ISARC). 2018, International Association for Automation and Robotics in Construction (IAARC). p. 693–700. DOI:⟨https://doi.org/10.22260/ISARC2018/0094⟩.

[58] J.A. Guzmán-Torres, M.Z. Naser, F.J. Domínguez-Mota, Effective medium crack classification on laboratory concrete specimens via competitive machine learning, Structures 37 (2022) 858–870, https://doi.org/10.1016/j.istruc.2022.01.061.

[59] Qayyum, W., A. Ahmad, N. Chairman, and A. Aljuhni, *Evaluation of GoogLenet, Mobilenetv2, and Inceptionv3, pre-trained convolutional neural networks for detection and classification of concrete crack images*, in 1st International Conference on Advances in Civil and Environmental Engineering. 2022: University of Engineering & Technology Taxila, Taxila, Pakistan.

[60] M. Mishra, P.B. Lourenço, G.V. Ramana, Structural health monitoring of civil engineering structures by using the internet of things: a review, J. Build. Eng. 48 (2022), 103954, https://doi.org/10.1016/j.jobe.2021.103954.

[61] H.S. Munawar, F. Ullah, A. Heravi, M.J. Thaheem, A. Maqsoom, Inspecting buildings using drones and computer vision: a machine learning approach to detect cracks and damages, Drones 6 (1) (2022) 5, https://doi.org/10.3390/drones6010005.

[62] K. Gopalakrishnan, S.K. Khaitan, A. Choudhary, A. Agrawal, Deep Convolutional Neural Networks with transfer learning for computer vision-based data-driven pavement distress detection, Constr. Build. Mater. 157 (2017) 322–330, https://doi.org/10.1016/j.conbuildmat.2017.09.110.

[63] Z. Wang, G. Xu, Y. Ding, B. Wu, G. Lu, A vision-based active learning convolutional neural network model for concrete surface crack detection, Adv. Struct. Eng. 23 (13) (2020) 2952–2964, https://doi.org/10.1177/1369433220924792.

[64] C. Feng, M.-Y. Liu, C.-C. Kao, T.-Y. Lee, Deep active learning for civil infrastructure defect detection and classification, Comput. Civ. Eng. (2017) 298–306, https://doi.org/10.1061/9780784480823.036.

[65] P. Kodytek, A. Bodzas, P. Bilik, A large-scale image dataset of wood surface defects for automated vision-based quality control processes, F1000Research 10 (581) (2022), https://doi.org/10.12688/f1000research.52903.2.

[66] MathWorks. Pretrained Deep Neural Networks. 2023; Available from: ⟨https://www.mathworks.com/help/deeplearning/ug/pretrained-convolutional-neural-networks.html⟩ (Accessed June 1, 2023).