Check for
updates

AttentionPoolMobileNeXt: An automated construction damage detection model based on a new convolutional neural network and deep feature engineering models

Mehmet Aydin[1] · Prabal Datta Barua[2] · Sreenivasulu Chadalavada[3] ·
Sengul Dogan[1] ⦿ · Turker Tuncer[1] · Subrata Chakraborty[4,5] · Rajendra U. Acharya[6]

Abstract

In 2023, Turkiye faced a series of devastating earthquakes and these earthquakes affected millions of people due to damaged constructions. These earthquakes demonstrated the urgent need for advanced automated damage detection models to help people. This study introduces a novel solution to address this challenge through the AttentionPoolMobile-NeXt model, derived from a modified MobileNetV2 architecture. To rigorously evaluate the effectiveness of the model, we meticulously curated a dataset comprising instances of construction damage classified into *five* distinct classes. Upon applying this dataset to the AttentionPoolMobileNeXt model, we obtained an accuracy of 97%. In this work, we have created a dataset consisting of five distinct damage classes, and achieved 97% test accuracy using our proposed AttentionPoolMobileNeXt model. Additionally, the study extends its impact by introducing the AttentionPoolMobileNeXt-based Deep Feature Engineering (DFE) model, further enhancing the classification performance and interpretability of the system. The presented DFE significantly increased the test classification accuracy from 90.17% to 97%, yielding improvement over the baseline model. AttentionPoolMobileN-eXt and its DFE counterpart collectively contribute to advancing the state-of-the-art in automated damage detection, offering valuable insights for disaster response and recovery efforts.

Keywords AttentionPoolMobileNeXt · Construction damage classification · Deep feature engineering · Image classification

1 Introduction

Earthquakes pose perilous threats to human life and property, ranking among the most devastating natural disasters humanity faces [1, 2]. In densely populated urban regions, the risk of building damage or collapse during seismic events reaches an exceedingly grave magnitude [3]. Following an earthquake, the rapid and meticulous assessment of structural harm becomes paramount for effective emergency response, rescue operations, and

---

Extended author information available on the last page of the article

Table 1 Literature review based on damage detection

Study	Methods	Data	Results (%)
Marín-García et al. [12]	CNN	765 building facade images	Avg Pre: 89.40
Hacıefendioglu et al. [13]	CNN	1068 wood images	Acc: 92.16
Hake et al. [14]	VGG19	99.764 harbor structures images	Acc: 98.80 Rec: 95.00
Chen et al. [15]	CNN	600 sample images	Acc: 92.30
Teng et al. [16]	1D-CNN	100 numerical models	Acc: 97.30
Zhang et al. [17]	CNN	3549 images	Acc: 89.60
Hake et al. [18]	ResNet-50	1300 images	Acc: 98.00
Chennareddy et al. [19]	CNN	4380 images	Acc: 95.00
Mostofi et al. [20]	Ensemble Learning	759 images	Acc: 63.00 Pre: 71.00 F1: 63.00
Kamoona et al. [21]	CNN	74 wood images	Sen: 87.00 Pre: 77.00
Kumar et al. [22]	YOLO-v3	800 images	Acc: 94.24
Pan and Yang [23]	YOLOv2	686 images	Acc: 98.20
Zhang et al. [24]	YOLO-ResNet50	5810 images	F1: 91.97 Rec: 94.53 Pre: 91.00
Teng et al. [25]	CNN	48.400 images	Acc: 99.00

**CNN: Convolutional neural network; Avg Pre: Average precision; Acc: accuracy; STFT: short-time Fourier transform; Rec: recall; Pre: precision; F1: F1-score; Sen: Sensitivity

subsequent reconstruction efforts [4]. In recent times, remote sensing technologies have emerged as invaluable instruments for discerning, detecting, and evaluating natural calamities, harnessing diverse data modalities including aerial or satellite images, Lidar, and SAR [5–7]. Thus, the precise classification of distinct forms of building impairment through the scrutiny of remote sensing imagery has become an exigent concern [8].

Various models for construction damage detection have been proposed in the literature. Some of these models are discussed below. Roy and Bhaduri [9] utilized DenseNet and Swin-Transformer for damage detection. The proposed DenseSPH-YOLOv5 model aimed to improve the accuracy and efficiency of damage detection in engineering informatics, using a large-scale road damage dataset (9053 road damage images) and achieving a precision of 89.51%. Seemab et al. [10] presented a method for detecting propagating cracks in reinforced concrete beams using digital image correlation measurements, providing effective crack detection. Zhu and Tang [11] proposed an automated hydraulic structures damage detection approach using drones, offering profound insights into maintenance and repairs by leveraging drone imagery and cutting-edge artificial intelligence techniques. A literature review regarding damage detection is given in Table 1.

It may be noted from the above table that, the research gaps are as follows:

– Explainable models are not much explored.
– The CNNs employed for such works are either customized versions of VGG or 1D-CNN. Additionally, while YOLO is a prevalent tool used for damage detection, there is a shortage of original CNN models with attention mechanisms.
– The two class datasets: (i) damaged and (ii) non-damaged have been used.

1.1 Motivation and our model

The year 2023 witnessed a series of catastrophic earthquakes in Turkiye, resulting in the deaths of over 50,000 people and rendering millions homeless. To alleviate the suffering of those affected by this tragedy, the foremost priority is to detect the extent of damage caused. However, this task can be a tedious and time-consuming process. To address this issue, we propose an automated damage detection model and have curated a novel dataset comprising five classes, namely (i) Debris, (ii) Damaged Buildings, (iii) Non-Damaged Buildings, (iv) Damaged Highways, and (v) Non-Damaged Highways.

To achieve high classification performance while retaining a lightweight deep learning model, we have modified the popular deep learning model, MobileNetV2 [26]. Since the introduction of vision transformers (ViT) [27], we have observed that attention blocks are efficient in achieving high classification performance. Therefore, we have added pooling-based attention blocks to MobileNetV2 to enhance its classification performance. Moreover, we have used the ConvNeXt strategy to modify MobileNetV2 blocks and we have obtained a more lightweight model. This modification has given rise to AttentionPoolMobileNeXt.

We have proposed a pyramidal deep feature engineering (DFE) model by utilizing the presented AttentionPoolMobileNeXt CNN. Firstly, we have trained the training dataset utilized by the proposed CNN. After that, we used the dropout layer of this pretrained AttentionPoolMobileNeXt as a deep feature extractor and this layer generated 256 features. We have utilized four leveled multilevel discrete wavelet transform (MDWT) [28] approximation to generate wavelet filters and we have extracted features from the raw image, low-low pass filter band (LL band), and low-high pass filter band (LH band). Subsequently, we have employed iterative neighborhood component analysis (INCA) [29] as a feature selector to identify the most relevant features, which have been classified using a support vector machine (SVM) [30, 31] classifier. Our proposed AttentionPoolMobileNeXt has achieved a testing accuracy of 90.17%, demonstrating the effectiveness of the cooperation between deep learning and feature engineering. Furthermore, our presented MDWT and AttentionPoolMobileNeXt deep learning models have attained a classification accuracy of 97%.

We have tried to address the literature gaps by introducing the following:

– Introduced a novel deep CNN model.
– Attained interpretable results from the proposed attention model to concentrate on regions of interest.
– Compiled a new image dataset comprising five distinct classes.

1.2 Innovations and contributions

The innovations and contributions of our work are given below:

Novelties:

– Collected a novel image dataset dedicated to construction damage detection, addressing the need for specialized data in this domain (automated damage detection).
– Introduced an attention-based CNN architecture named AttentionPoolMobileNeXt, emphasizing its unique design for improved performance.

– Presented a comprehensive deep feature engineering model by synergizing Attention-PoolMobileNeXt, the MDWT-based approximation inspired by watermarking methods, and machine learning algorithms such as INCA and SVM.

Contributions:

– Proposed a novel automated damage detection model for accurate classification of five classes.
– Developed a lightweight deep learning model designed for rapid and accurate classification of damages, aiming to address the need for efficiency in real-world scenarios.
– Achieved an exceptional test classification accuracy of over 90% on the curated dataset, demonstrating the efficacy and reliability of the proposed approaches.

Our work contributed to the field of construction damage detection and classification. By introducing innovative combinations of attention-based CNN architecture, MDWT-based feature extraction, and machine learning algorithms, our work not only addresses the challenges associated with existing methods but also provides comprehensive technical details, emphasizing both methodological and empirical advancements.

2 Dataset

To evaluate our model's performance, we collected a novel image dataset consisting of *five* distinct classes [32–36]. The dataset's classes are as follows: (1) Debris, (2) Damaged building, (3) Damaged highway, (4) Non-damaged building, and (5) Non-damaged highway. Furthermore, we partitioned the dataset into training and testing subsets. The images were stored in either JPEG or JPG format and had varying dimensions. The dataset's attributes are presented in Table 2.

As can be observed from Table 2, the construction damage image dataset is inherently imbalanced.

The collected dataset is publicly available on Kaggle, and researchers can download it using the following URL: https://www.kaggle.com/datasets/turkertuncer/damaged-constructions-image-dataset.

Table 2 Characteristics of the collected image dataset for construction damage classification

No	Class name	Train	Test	Total
1	Debris	413	100	513
2	Damaged building	448	100	548
3	Damaged highway	719	100	819
4	Non-damaged building	1659	200	1859
5	Non-damaged highway	530	100	630
Total		3769	600	4369

3 The proposed AttentionPoolMobileNeXt

Our essential objective is to present a novel lightweight deep learning model, termed AttentionPoolMobileNeXt. To achieve this, we have expanded upon the MobileNetV2 architecture by incorporating two attention blocks, drawing inspiration from Pool-Former [37]. Additionally, we have made enhancements to the MobileNetV2 blocks akin to ConvNeXt, utilizing combinations of convolution + normalization and convolution + activation. To provide a comprehensive understanding of our proposed model, we initially provide a succinct overview of the MobileNetV2 architecture. We have used the attention layer to focus region of interest and we have used a ConvNeXt-based strategy to extract a meaningful feature map with less number of learnable parameters than MobileNetV2.

MobileNetV2 leverages convolutions, linear bottlenecks, and inverted residuals [26], employing 1×1 and 3×3 convolutions for extracting image features. Feature transformation involves bottleneck inputs, 1×1 convolution, and depth-wise convolution with 3×3 filter sizes. While the architecture integrates addition-based shortcuts reminiscent of Residual Networks to address the vanishing gradient problem, it lacks attention blocks. To address this limitation and enhance classification performance, we have introduced two attention blocks to the MobileNetV2 architecture. Our selection of the MobileNetV2 architecture stems from its efficiency, with ongoing research efforts focusing on the development of MobileNetV3 [38].

In this study, we extended the MobileNetV2 architecture by incorporating attention blocks, resulting in the development of AttentionPoolMobileNeXt. This lightweight deep-learning model is designed to enhance classification performance. Our proposed architecture encompasses convolution, residual mobile ConvNeXt blocks (ResMoB), mobile ConvNeXt blocks (MoB), pooling-based attention blocks, and output blocks. To facilitate a clearer understanding of AttentionPoolMobileNeXt, we have provided a schematic overview of the model.

Figure 1 illustrates our utilization of both average pooling and maximum pooling to construct an attention mechanism resembling that of PoolFormer. This strategic implementation has allowed us to introduce a more lightweight model compared to Mobile-NetV2. Furthermore, the transition details of this AttentionPoolMobileNeXt are presented in Table 3.

Table 3 outlines the architecture and operations performed at various stages of the proposed AttentionPoolMobileNeXt.

4 The proposed deep feature engineering model

This research introduces a novel DFE model based on the presented AttentionPoolMobileNeXt, aimed at enhancing test classification. This pyramidal DFE model comprises three pivotal phases: (i) deep feature extraction utilizing MDWT and the pretrained AttentionPoolMobileNeXt, (ii) feature selection based on INCA, and (iii) classification using Support Vector Machine (SVM). To assess the effectiveness of our model, we apply it to test images and present the results. Figure 2 illustrates a block diagram of our developed deep feature engineering model based on AttentionPoolMobileNeXt, offering a clearer insight into its architectural design.
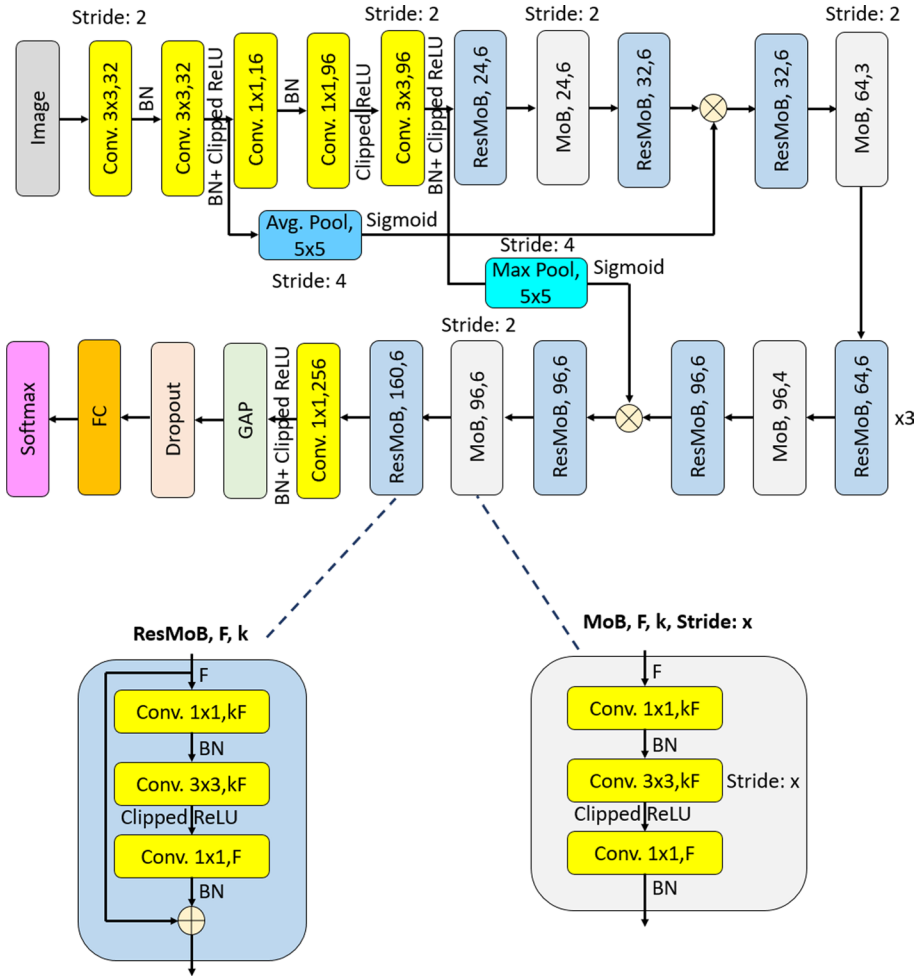
Fig. 1 Schematic representation of the presented AttentionPoolMobileNeXt. Herein, Conv.: convolution, BN: batch normalization, Avg. Pool: average pooling, Max Pool: maximum pooling, GAP: global average pooling

The steps of the developed deep feature engineering model are presented as follows.

Step 1: Read each image from the collected test image dataset.

Step 2: Apply MDWT with four levels to each image. In this step, we have used low-pass filter bands (LL and LH) to extract features. Moreover, the 'haar' mother wavelet mother function has been used to get images.

$$\left[LL_1, LH_1, HL_1, HH_1 \right] = \delta(Im) \tag{1}$$

$$\left[LL_k, LH_k, HL_k, HH_k \right] = \delta\left(LL_{k-1} \right), k \in \{2, 3, 4\} \tag{2}$$

Table 3 Transition list of the presented AttentionPoolMobileNeXt

Stage	Input	Operation	Output
Stem 1	224×224×3	3×3, 32, BN, stride: 2, 3×3, 32, BN, Clipped ReLU,	112×112×32
Stem 2	112×112×32	Average pooling, 5×5, 32, stride: 4, sigmoid	28×28×32
Stem 3	112×112×32	1×1, 32, BN, 1×1, 96, Clipped ReLU, 3×3, 96, BN, Clipped ReLU, stride: 2	56×56×96
Stem 4	56×56×96	Maximum pooling, 5×5, 32, stride: 4, sigmoid	14×14×96
ResMoB 1	56×56×96	$\begin{bmatrix} 1\times1,144 \\ 3\times3,144 \\ 1\times1,24 \end{bmatrix}$	56×56×24
MoB 1	56×56×24	$\begin{bmatrix} 1\times1,144 \\ 3\times3,144, stride:2 \\ 1\times1,24 \end{bmatrix}$	28×28×24
ResMoB 2	28×28×24	$\begin{bmatrix} 1\times1,192 \\ 3\times3,192 \\ 1\times1,32 \end{bmatrix}$	28×28×32
Pool attention 1		$(ResMoB2) \otimes (Stem2)$	56×56×32
ResMoB 3	28×28×32	$\begin{bmatrix} 1\times1,192 \\ 3\times3,192 \\ 1\times1,32 \end{bmatrix}$	28×28×32
MoB 2	28×28×32	$\begin{bmatrix} 1\times1,192 \\ 3\times3,192, stride:2 \\ 1\times1,64 \end{bmatrix}$	14×14×64
ResMoB 4–7	14×14×64	$\begin{bmatrix} 1\times1,384 \\ 3\times3,384 \\ 1\times1,64 \end{bmatrix} \times 3$	14×14×64
MoB 3	14×14×64	$\begin{bmatrix} 1\times1,384 \\ 3\times3,384 \\ 1\times1,96 \end{bmatrix}$	14×14×96

Table 3 (continued)

| Stage	Input	Operation	Output
ResMoB 8	14×14×96	$\begin{bmatrix} 1 \times 1,576 \\ 3 \times 3,576 \\ 1 \times 1,96 \end{bmatrix}$	14×14×96
Pool attention 2	14×14×96	$(ResMoB8) \otimes (Stem4)$	14×14×96
ResMoB 9	14×14×96	$\begin{bmatrix} 1 \times 1,576 \\ 3 \times 3,576 \\ 1 \times 1,96 \end{bmatrix}$	14×14×96
MoB 4	14×14×96	$\begin{bmatrix} 1 \times 1,576 \\ 3 \times 3,576, stride : 2 \\ 1 \times 1,96 \end{bmatrix}$	7×7×96
ResMoB 10	7×7×96	$\begin{bmatrix} 1 \times 1,960 \\ 3 \times 3,960 \\ 1 \times 1,160 \end{bmatrix}$	7×7×160
Output	7×7×160	1×1, 256, BN, Clipped ReLU, GAP, Dropout with a probability: 0.5, fully connected layer, softmax, classification	Number of classes
Total learnable parameters			About 1 Million
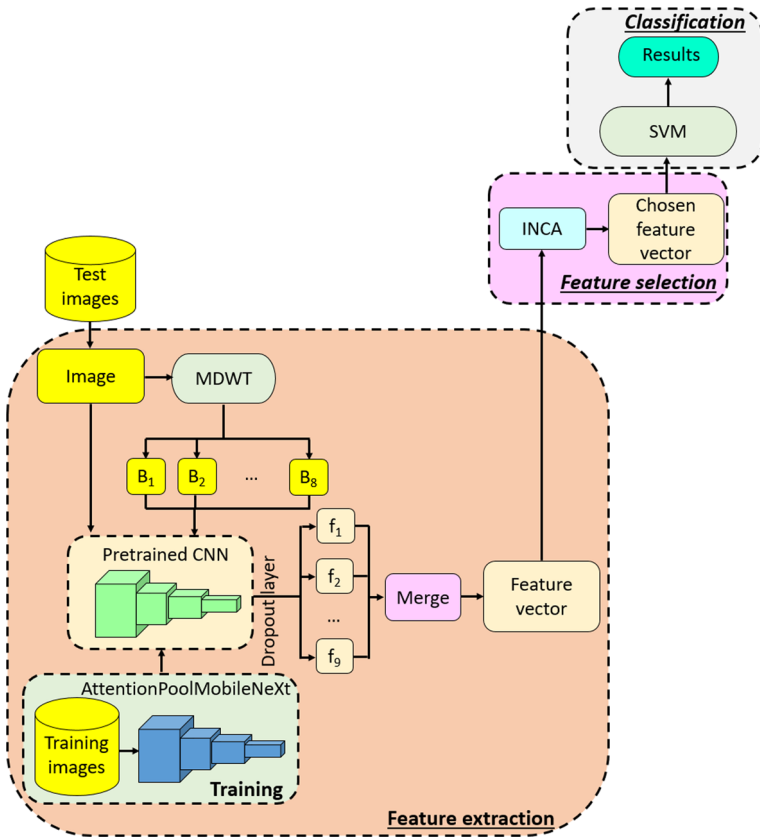
Fig. 2 Block diagram of the developed deep feature engineering model based on the recommended AttentionPoolMobileNeXt. Herein, MDWT: multilevel discrete wavelet transform, B: wavelet band, f: individual features, INCA: iterative neigbourhood component analysis

where LL, LH, HL and HH are the low-low, low–high, high-low and high-high filter bands, $\delta()$ defines the DWT function and Im is the raw image. We have used four leveled MDWT to get LL and LH bands. MDWT generates floating point bands. Therefore, we have used normalization to get images.

Step 3: Create features from the raw image and low-pass filter bands. We have used the pretrained AttentionPoolMobileNeXt CNN. We have used the dropout layer of the presented AttentionPoolMobileNeXt. We have generated 256 features from the generated images.

$$f(j) = \alpha(Im), j \in \{1, 2, \dots, 256\} \tag{3}$$

$$f(j + h \times 256) = \alpha\left(LL_t\right), t \in \{1, 2, 3, 4\}, h \in \{2, 4, 6, 8\} \tag{4}$$

$$f(j + 256 \times (2h - 1)) = \alpha\left(LH_t\right), \tag{5}$$

Herein, f defines the feature vector, $\alpha(.)$ is the pretrained AttentionPoolMobileNeXt. In this phase, we have generated 9 feature vectors and we merged these feature vectors. In this step, 2304 ($= 256 \times 9$) features have been extracted.

Step 4: Apply the iterative feature selector to the generated features with a length of 2304. INCA was proposed by Tuncer et al. [29] in 2020. In this work, an improved Neighborhood Component Analysis (NCA) [39] has been utilized to select the most relevant feature vector from the created set of features. Our approach, the iterative neighborhood component analysis (INCA), automatically determines the optimal number of features to be selected. Specifically, the qualified feature indexes are initially generated using the NCA feature selector. We then create a loop to iteratively select the relevant features and use a classifier to calculate the selected feature vectors and loss value array. The feature vector with the minimum misclassification value is chosen as the final feature vector. The mathematical formulation of our proposed feature selector is provided below for further clarity.

$$id1 = NCA(f, y) \tag{6}$$

$$s^{r-sv+1}(dim, i) = f(dim, id1(i)), i \in \{1, 2, \dots, r\}, \\ dim \in \{1, 2, \dots, NI\}, r \in \{sv, sv+1, \dots, fv\} \tag{7}$$

$$loss(r) = C(s^r, y) \tag{8}$$

Herein, $id1$ represents the qualified indexes that are generated by NCA ($NCA(.)$) feature selector, y is the actual/real output, s implies the selected feature vector, sv defines start value of the loop and fv is the final value of the loop. NI is the number of images. We have calculated loss values ($loss$) of the selected feature vector by deploying a classifier ($C(.)$). By using the loss value array calculated, the most suitable feature vector has been selected below.

$$id2 = min(loss) \tag{9}$$

$$selfeat = s^{id2} \tag{10}$$

where $id2$ is the index of the loss value with minimum loss value and *selfeat* is the final selected feature.

In this work, we have calculated the loss values employing SVM classifier. In this work, the most relevant 116 features have been selected.

Step 5: Fed to choose 116 features for the SVM classifier with 10-fold cross-validation. SVMs have long been recognized as one of the preeminent shallow classifiers. To this end, we sought to combine the power of SVMs with that of INCA in our research. We have tuned the parameters of the used SVM by deploying the Bayesian optimization [40]. The hyperparameters utilized for the SVM are outlined below: Kernel: Polynomial, Kernel Scale: 1, Standardize: True, Polynomial Order: 2, Box Constraint: 983.4589707080628, Coding: One-vs-All, Validation: 10-Fold Cross-Validation

We have dubbed this particular instantiation of the SVM the Quadratic SVM.

$$pr = SVM(selfeat, y) \tag{11}$$

Herein, *pr* defines the predicted vector by applying the SVM classifier.

5 Experimental results

Our work introduces two novel contributions to the field: The proposed CNN, and a deep feature engineering model. We partitioned the dataset into separate training and testing sets, and present the results of our experiments in this section.

5.1 Setup

To implement our proposed AttentionPoolMobileNeXt model, we utilized a personal computer (PC) equipped with an NVidia Ge-force 2070 graphical processing unit (GPU), 64 GB of memory, a 3.6 GHz processor, and the Windows 11 operating system. We employed the MATLAB programming environment, leveraging both the deep network designer and classification learner toolboxes to create our proposed models.

In our approach, we first trained the AttentionPoolMobileNeXt model on the training dataset and then computed a pretrained AttentionPoolMobileNeXt. The training options for the AttentionPoolMobileNeXt model were an initial learning rate of 0.005, a maximum of 20 epochs, and a mini-batch size of 32. We split the data into training and validation sets at an 80:20 ratio.

5.2 Results

In this section presents the classification results obtained using our proposed models. Our first step was to train the AttentionPoolMobileNeXt model, and provide the classification curve for this model during the training phase in Fig. 3.
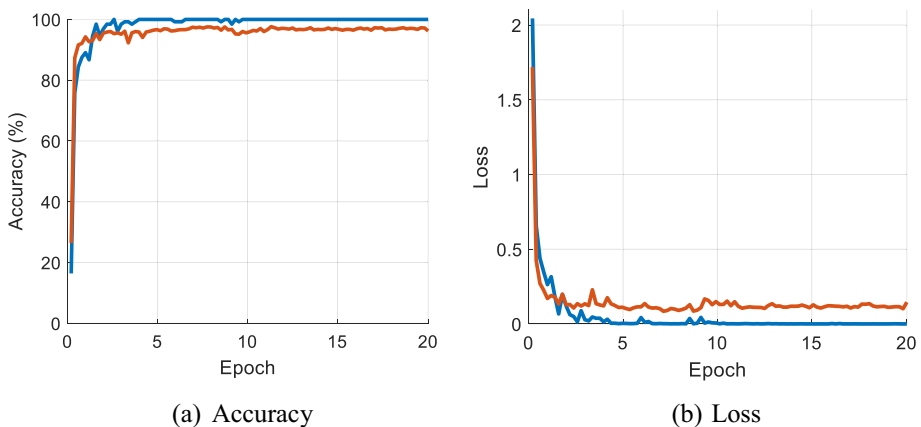

(a) Accuracy   (b) Loss

Fig. 3 Training and validation curves obtained for the proposed AttentionPoolMobileNeXt with damaged construction dataset

Fig. 4 Confusion matrix obtained for the proposed method. * 1: Debris, 2: Damaged building, 3: Damaged highway, 4: Non-damaged building, 5: Non-damaged highway

Figure 3 provides evidence of AttentionPoolMobileNeXt's remarkable training accuracy, which reached 100%. In addition, the final validation accuracy was 97.35%. To evaluate the model's performance further, we utilized the test dataset and generated a confusion matrix, as depicted in Fig. 4.

As depicted in Fig. 4, the AttentionPoolMobileNeXt model exhibited a test classification accuracy of 90.17%. To enhance this performance, we introduced a DFE model that improved the classification capabilities of AttentionPoolMobileNeXt. In this DFE model, we extracted features from various sources, including the dropout layer of the pretrained AttentionPoolMobileNeXt, raw image data, and the LL and LH wavelet bands of the images.

Our deep feature extractor generated 256 features from each input, and with nine inputs, a total of 2304 features were extracted from each image. Employing INCA as a feature selector, we identified the most valuable 116 features out of the initial 2304. In the final phase, a SVM was employed for classification, resulting in our developed DFE model achieving an impressive 97% classification accuracy.

Fig. 5 Confusion matrix obtained for the proposed DFE model based on AttentionPool-MobileNeXt

The confusion matrix for our AttentionPoolMobileNeXt-based model is presented in Fig. 5.

As shown in Fig. 5, our deep feature engineering model achieved a classification accuracy of 97%.

To comprehensively evaluate the classification performance of our model, we employed commonly used metrics such as accuracy, recall, precision, and F1-score. The details of these metrics are given below [41]:

Classification accuracy: It is the ratio of correctly predicted instances to the total instances in the dataset. A high accuracy indicates the overall correctness of the model's predictions. However, it may not be suitable for imbalanced datasets. Therefore, we need to use other metrics.

Recall: It measures the ability of a model to capture all the relevant instances and is termed class-wise classification accuracy. High recall implies fewer instances of the positive class being overlooked, which is crucial when false negatives are costly.

Precision: It assesses the accuracy of positive predictions made by the model. High precision indicates that a positive prediction by the model is likely to be accurate, minimizing false positives.

F1-score: It is the harmonic mean of precision and recall, providing a balanced metric. F1-score considers both false positives and false negatives, making it a suitable metric when there is an imbalance between classes.

Table 4 Performance measures (%) obtained for the presented models

Metric	Class	AttentionPoolMobile-NeXt	AttentionPoolMobileNeXt-based DFE
Accuracy	Overall	90.17	97
Recall	Debris	56	94
	Damaged building	100	99
	Damaged highway	97	95
	Non-damaged building	100	99.50
	Non-damaged highway	88	95
	Overall	88.20	96.50
Precision	Debris	100	96.91
	Damaged building	95.24	97.06
	Damaged highway	68.79	94.06
	Non-damaged building	98.04	98.51
	Non-damaged highway	93.62	96.94
	Overall	91.14	96.70
F1	Debris	71.79	95.43
	Damaged building	97.56	98.02
	Damaged highway	80.50	94.53
	Non-damaged building	99.01	99
	Non-damaged highway	90.72	95.96
	Overall	87.92	96.59

The accuracy provides an overall view, recall, precision, and F1-score offer insights into specific aspects of a model's performance, particularly when dealing with imbalanced datasets or scenarios where certain types of errors are more critical than others.

The calculated test results are summarized in Table 4.

Table 4 reveals that the proposed model has achieved a 90.17% and 97% test classification accuracies. Notably, the non-damaged buildings class emerged as the best-performing class for the DFE model, exhibiting a remarkable 99.50% recall rate. In addition, AttentionPoolMobileNeXt demonstrated exceptional recall performance of 100% for the damaged building and non-damaged buildings class. However, both models exhibited poor performance in the debris class, which emerged as the worst-performing class

Precision metrics reveal that the AttentionPoolMobileNeXt-based DFE model excels in precision across various categories, showcasing notable improvements in Damaged highway, Non-damaged building, and Non-damaged highway classifications. These enhancements emphasize the model's increased accuracy in correctly predicting positive instances, minimizing false positives.

Analyzing the F1-Score, a metric that balances precision and recall, the AttentionPoolMobileNeXt-based DFE model consistently exhibits improvements across different classes and overall performance. Particularly noteworthy are the substantial improvements in Debris, Damaged building, and Non-damaged building classifications, emphasizing the model's improved balance between precision and recall.

The class-specific performance analysis further underscores the AttentionPoolMobileNeXt-based DFE model's proficiency in classifying Debris, Damaged buildings, and Non-damaged buildings, achieving high recall, precision, and F1-Score. Notable improvements are also observed in the Damaged highway classification, showcasing the DFE model's effectiveness across diverse construction damage categories.

The overall performance metrics, including accuracy, recall, precision, and F1-Score, collectively demonstrate the superiority of the AttentionPoolMobileNeXt-based DFE model over the baseline AttentionPoolMobileNeXt model. These findings underscore the efficacy and reliability of the deep feature engineering approach in advancing construction damage classification.

5.3 Explainable results

In our study, the application of AttentionPoolMobileNeXt, coupled with the Gradient-weighted Class Activation Mapping (Grad-CAM) method [42, 43], has provided valuable insights in the domain of construction damage detection. Figure 6 visually represents our model's capability in accurately identifying damaged areas, emphasizing the role of attention blocks within AttentionPoolMobileNeXt.

It may be noted from Fig. 6 that, the embedded attention mechanisms effectively focus on key features which are indicative of construction damage. Grad-CAM highlights specific regions in the images where the AttentionPoolMobileNeXt's attention is concentrated, offering visual interpretation of the decision-making process. This transparency contributes to AttentionPoolMobileNeXt's interpretability, a crucial aspect to instill robustness in our findings.

The attention blocks play a crucial role in enabling the model to discern intricate patterns and subtleties associated with different forms of construction damages. This meticulous attention to relevant features enhances the overall classification accuracy of our model in identifying damaged areas.
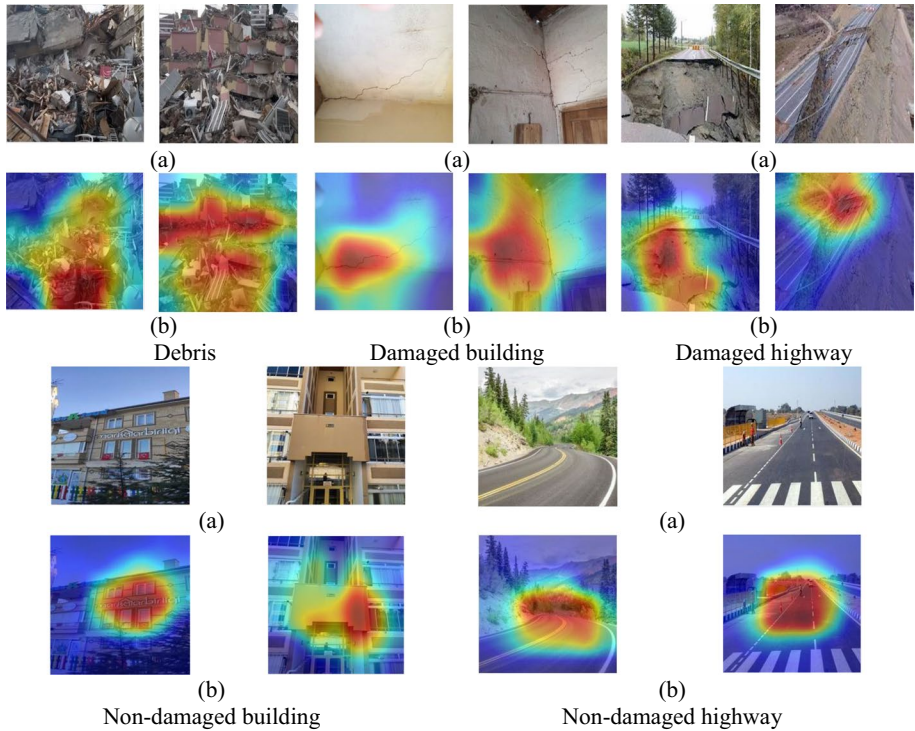
Fig. 6 Heat map images obtained using the Grad-CAM technique for different images

Additionally, the Grad-CAM visualization acts as an interpretability tool and window to the internal workings of AttentionPoolMobileNeXt during classification. This helps to grasp the rationale behind specific decisions, providing valuable insights for further refinement of the model and practical application in real-world scenarios.

The combined use of AttentionPoolMobileNeXt and Grad-CAM has resulted in obtaining higher classification accuracies, and interpretability. The visual representations in Fig. 6 justifies the effectiveness of attention mechanisms in enhancing model performance and offer a transparent view into the decision-making processes, promoting trust and understanding of the model's predictions.

6 Discussions

We have introduced a novel attention-based CNN by adapting MobileNetV2, termed AttentionPoolMobileNeXt. This model represents a cutting-edge approach to deep feature engineering for image classification. The primary objective of AttentionPoolMobileNeXt is to explore the classification outcomes achieved through attention mechanisms in conjunction with a lightweight network. We deploy this network with a humanitarian focus, particularly in response to the profound impact of the 2023 seismic events in Turkiye. These earthquakes underscore the critical need for swift damage detection to assist disaster-stricken communities, considering the time-intensive nature of manual assessment.

Fig. 7 Number of features selected in various wavelet bands
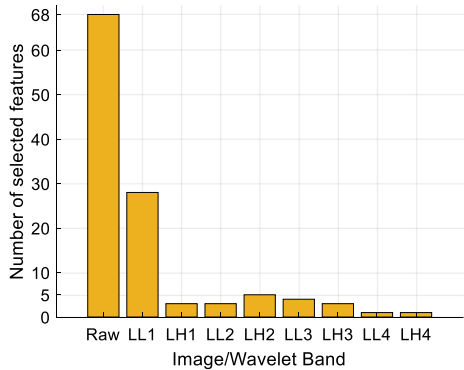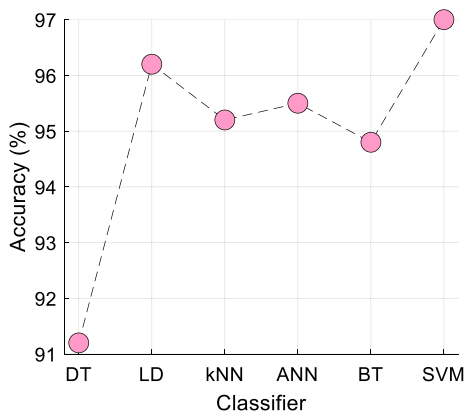

Fig. 8 Classification accuracy obtained for various classifiers

Therefore, an automated damage detection model is imperative, and leveraging the capabilities of deep learning stands out as one of the most effective approaches to address this urgent requirement.

We formulate the damage detection problem as a computer vision problem and gather an image dataset from open-source image datasets. Our presented AttentionPoolMobileNeXt and AttentionPoolMobileNeXt-based deep feature engineering models achieve classification accuracies of 90.17% and 97%, respectively. In the presented DFE model, we generate deep features from raw image low-pass filter images. INCA, an iterative feature selector, is then used to select the most relevant features, and SVM is employed to classify the selected features.

In the following (Fig. 7), we provide a detailed analysis of these features.

Figure 7 reveals that out of the selected 116 features, 68 were generated from raw images, while the other 48 were generated using wavelet bands. Notably, LL (36 features were generated from LL bands) bands are more beneficial than LH (12 out of the selected 112 features were generated from LH bands) bands. Furthermore, Fig. 8 demonstrated that all inputs contributed to obtaining an accuracy of 97%.

We utilized SVM for both selecting the most relevant feature vector in INCA and obtaining classification results. In selecting the appropriate SVM classifier, we conducted tests using decision tree (DT), linear discriminant (LD), k-nearest neighbors

Fig. 9 The comparative results.
*Mob: MobileNetV2, Effb0:
EfficientNetb0, IncV3: Inception-
tionV3, IncResNetV2: Inception-
ResNetV2 and AttPoolMob:
AttentionPoolMobileNeXt
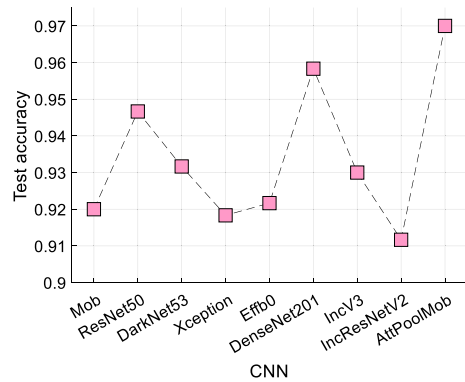
(k-NN), artificial neural network (ANN), bagged tree (BT), and SVM classifiers. The results of these tests are illustrated in Fig. 9.

It can be noted from Fig. 8 that, the best classifier used is SVM, which attained a 97% classification accuracy. Additionally, LD attained a 96.17% classification accuracy for our selected features. In contrast, the worst-performing classifier is DT, which achieved an accuracy of 91.33%.

The comparative results are presented in Table 5.

The information from Table 5 highlights that Liu et al. [47] achieved the closest result to our method, attaining a 98% accuracy. It is crucial to note, however, that Liu et al. [47] utilized a two-class dataset, while our study employed a more diverse five-class construction dataset. This distinction underscores the complexity of our dataset, and despite this increased challenge, we achieved a commendable 97% classification accuracy.

Furthermore, we employed our dataset to showcase the high classification perfor-
mance of our model, employing well-established CNNs: MobileNetV2, ResNet50, DarkNet53, Xception, EfficientNetb0, DenseNet201, InceptionV3, and Inception-
ResNetV2. The test classification accuracies of these CNNs were compared with our proposed AttentionPoolMobileNeXt CNN, and the outcomes are presented in Fig. 9. To obtain accurate test classification accuracies and facilitate a reliable comparison, we uti-
lized the test classification results of these CNNs by applying DFE approaches, similar to our model. This comprehensive evaluation provides a clear perspective on the perfor-
mance of our model in comparison to widely recognized CNN architectures.

Figure 9 indicates that, our model achieved the highest test classification accuracy of 97% using our curated dataset. In comparison, MobileNetV2, our inspired CNN, attained a test classification accuracy of 92%. DenseNet201 performed the best, achiev-
ing a test classification accuracy of 95.83%. Our proposed AttentionPoolMobileNeXt is the lightest among them, with only approximately 1 million learnable parameters.

The findings, advantages, and limitations of our proposed method are given below.

Findings:

– Presented AttentionPoolMobileNeXt demonstrates a proficient ability to accurately identify areas affected by construction damage (Fig. 6).

Table 5 Comparison of our work with other state-of-the-art techniques

Study	Aim	Method	Dataset	Number of classes	Split ratio	Accuracy (%)
Naito et al. [44]	Building-damage detection	Scale-invariant feature transform, CNN	2016 Kumamoto earthquake	4	10-fold CV	88.40
Stramondo et al. [45]	Earthquake damage detection	Pixel-based and object-oriented procedures	1. Izmit 2. Bam earthquake	1. 3 2. 2	50:50	1. 76.00 for Bam 2. 89.00 for Izmit
Abdi et al. [46]	Earthquake building damage detection	CNN	2010 Haiti earthquake	2	60:40	91.00
Liu et al. [47]	Building damage detection	YOLOv5, Bi-directional feature pyramid network, depthwise separable convolution	2008 Wenchuan earthquake	4	70:30	98.00
Seydi et al. [48]	Building damage detection	CNN	2010 Haiti earthquake	2	54:10:36	88.00
Lin et al. [49]	Seismic building damage	CNN	Yushu dataset	4	60:20:20	90.14
Zhan et al. [50]	Damaged building extraction	Mask R-CNN	2016 Kumamoto Earthquake	4	88:12	95.10
Our method	Construction damage detection	AttentionPoolMobileNeXt model	2023 Kahramanmaras earthquake	5	10-fold CV	97.00

– The attention blocks within AttentionPoolMobileNeXt focuses on salient features of construction damages and helps in recognizing intricate patterns associated with diverse forms of damage.
– Grad-CAM provides transparency in the decision-making process by highlighting specific regions where the model's attention is concentrated. This visual interpretation enhances the model's elucidation, fostering confidence in the predictions.
– Developed DFE model has increased the test accuracy from 90.17% to 97%.

Merits:

– Collected diverse image dataset involving five classes for automatic construction damage detection.
– Proposed the AttentionPoolMobileNeXt model by incorporating two pooling functions and two attention blocks into the MobileNetV2, to obtain high classification performance.
– Developed AttentionPoolMobileNeXt model is a lighter model than MobileNetV2 as it used 1 million learnable parameters.
– Presented AttentionPoolMobileNeXt reached higher classification performances than other commonly known CNNs (see Fig. 9).
– Developed both CNN and DFE-based models have demonstrated superior classification performances.
– Generated models outperformed the existing models, highlighting their potential usefulness in practical applications.

Limitations:

– Although we tested our model on a significant dataset, additional evaluation on other datasets could further validate its performance. In this work, we focused on the serial earthquakes occurred in Turkiye. The model needs to be validated using the dataset obtained from other earthquake sites.

7 Conclusions

The impact of natural disasters on human lives is significant, and detecting the damage caused by these disasters is crucial. However, this process can be time-consuming, particularly for large-scale disasters. To address this issue, we propose a novel damage detection model to assist civil and construction engineers in identifying areas of damage more efficiently.

Our proposed model is based on attention-based CNN called AttentionPoolMobileNeXt. To evaluate the effectiveness of our approach, we acquired a new image dataset consisting of *five* classes. Our AttentionPoolMobileNeXt model achieved a 90.17% accuracy, while the AttentionPoolMobileNeXt-based DFE model reached an even higher 97% accuracy. These results demonstrate the effectiveness of our developed models for construction damage detection.

Our future work will focus on collecting more diverse and comprehensive construction damage datasets. We also aim to develop a more efficient attention CNN to achieve higher classification performance with fewer parameters than current lightweight CNNs.

Data availability The authors are committed to making the data available if requested by the journal.

Declarations

Ethical approval Ethical approval was not required for this research.

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

References

1. Chukwuka OJ, Ren J, Wang J, Paraskevadakis D (2023) A comprehensive research on analyzing risk factors in emergency supply chains. J Humanit Logist Supply Chain Manag 13(3):249–292. https://doi.org/10.1108/jhlscm-10-2022-0108
2. Kitch SL (2023) Reproductive rights and ecofeminism. Humanities 12(2):34
3. Kuo SS, Marshall JT, Rowberry R (2022) The Cambridge handbook of disaster law and policy: Risk, recovery, and redevelopment. Cambridge University Press
4. McEntire D, Souza J, Collins ML, Peters EJ, Sadiq A-A (2012) An introspective glance into damage assessment: challenges and lessons learned from the Paso Robles (San Simeon) earthquake. Nat Hazards 61:1389–1409
5. Council NR (2014) Opportunities to use remote sensing in understanding permafrost and related ecological characteristics: report of a workshop. The National Academies Press. https://dels.nationalacademies.org/Report/Opportunities-Remote-Sensing/18711
6. Ang L-M, Seng KP, Zungeru AM, Ijemaru GK (2017) Big sensor data systems for smart cities. IEEE Internet Things J 4(5):1259–1271
7. Reding DF, Eaton J (2020) Science and technology trends 2020–2040: exploring the S and T edge. NATO S and T Organization, pp 94–103
8. Dong L, Shan J (2013) A comprehensive review of earthquake-induced building damage detection with remote sensing techniques. ISPRS J Photogramm Remote Sens 84:85–99
9. Roy AM, Bhaduri J (2023) DenseSPH-YOLOv5: An automated damage detection model based on DenseNet and Swin-Transformer prediction head-enabled YOLOv5 with attention mechanism. Adv Eng Inform 56:102007
10. Seemab F, Schmidt M, Baktheer A, Classen M, Chudoba R (2023) Automated detection of propagating cracks in RC beams without shear reinforcement based on DIC-controlled modeling of damage localization. Eng Struct 286:116118
11. Zhu Y, Tang H (2023) Automatic damage detection and diagnosis for hydraulic structures using drones and artificial intelligence techniques. Remote Sens 15(3):615
12. Marín-García D, Bienvenido-Huertas D, Carretero-Ayuso MJ, Della Torre S (2023) Deep learning model for automated detection of efflorescence and its possible treatment in images of brick facades. Autom Constr 145:104658

13. Hacıefendioğlu K, Ayas S, Başağa HB, Toğan V, Mostofi F, Can A (2022) Wood construction damage detection and localization using deep convolutional neural network with transfer learning. Eur J Wood Wood Prod 80(4):791–804

14. Hake F, Göttert L, Neumann I, Alkhatib H (2022) Using machine-learning for the damage detection of harbour structures. Remote Sens 14(11):2518

15. Chen L, Chen W, Wang L, Zhai C, Hu X, Sun L, Tian Y, Huang X, Jiang L (2023) Convolutional neural networks (CNNs)-based multi-category damage detection and recognition of high-speed rail (HSR) reinforced concrete (RC) bridges using test images. Eng Struct 276:115306

16. Teng S, Chen X, Chen G, Cheng L (2023) Structural damage detection based on transfer learning strategy using digital twins of bridges. Mech Syst Signal Process 191:110160

17. Zhang X, Wogen BE, Liu X, Iturburu L, Salmeron M, Dyke SJ, Poston R, Ramirez JA (2023) Machine-aided bridge deck crack condition state assessment using artificial intelligence. Sensors 23(9):4192

18. Hake F, Scherff M, Neumann I, Alkhatib H (2023) Using semantic segmentation for the damage detection of port and marine infrastructures. Berlin, Offenbach: Wichmann, pp 93–106

19. Chennareddy LN, Gandabathula SV, Jasthi VV, Shaik F (2023) Railway Bridge Inspection using CNN. In: 2023 7th International Conference on Computing Methodologies and Communication (ICCMC). IEEE, pp 416–421

20. Mostofi F, Toğan V, Ayözen YE, BehzatTokdemir O (2022) Predicting the impact of construction rework cost using an ensemble classifier. Sustainability 14(22):14800

21. Kamoona AM, Gostar AK, Wang X, Easton M, Bab-Hadiashar A, Hoseinnezhad R (2024) Anomaly detection of defect using energy of point pattern features within random finite set framework. Eng Appl Artif Intell 130:107706

22. Kumar P, Batchu S, Kota SR (2021) Real-time concrete damage detection using deep learning for high rise structures. IEEE Access 9:112312–112331

23. Pan X, Yang T (2020) Postdisaster image-based damage detection and repair cost estimation of reinforced concrete buildings using dual convolutional neural networks. Comput Aided Civ Infrastruct Eng 35(5):495–510

24. Zhang J, Yang X, Li W, Zhang S, Jia Y (2020) Automatic detection of moisture damages in asphalt pavements from GPR data with deep CNN and IRS method. Autom Constr 113:103119

25. Teng S, Chen G, Gong P, Liu G, Cui F (2020) Structural damage detection using convolutional neural networks combining strain energy and dynamic response. Meccanica 55:945–959

26. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L-C (2018) Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4510–4520

27. Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S (2020) An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:201011929

28. Fang S-H, Chang W-H, Tsao Y, Shih H-C, Wang C (2016) Channel state reconstruction using multilevel discrete wavelet transform for improved fingerprinting-based indoor localization. IEEE Sens J 16(21):7784–7791

29. Tuncer T, Dogan S, Özyurt F, Belhaouari SB, Bensmail H (2020) Novel multi center and threshold ternary pattern based method for disease detection method using voice. IEEE Access 8:84532–84540

30. Vapnik V (1998) The support vector method of function estimation. In: Nonlinear Modeling. Springer, pp 55–85

31. Vapnik V (2013) The nature of statistical learning theory. Springer Science & Business Media, Berlin, Germany

32. Karim F, Rajbangshi K (2023) Clean/littered road classification. https://www.kaggle.com/datasets/faizalkarim/cleandirty-road-classification. Accessed 05.10.2023

33. Viren (2023) Pothole and plain road images. https://www.kaggle.com/datasets/virenbr11/pothole-and-plain-rode-images. Accessed 12.09.2023

34. Haberler.com (2023) Frightening images in Hatay: the 3-kilometer road first split and then collapsed. https://www.dailymotion.com/video/x8ifici. Accessed 12.10.2023

35. Cuneyt O (2023) Here is the last state of Hatay. The last scene before the destruction. https://www.youtube.com/watch?v=TnlvaWRMpLs. Accessed 05.10.2023

36. Roy J (2006) E-government in Canada: transformation for the digital age. Ottowa, ON, Canada: University of Ottawa Press

37. Yu W, Luo M, Zhou P, Si C, Zhou Y, Wang X, Feng J, Yan S (2022) Metaformer is actually what you need for vision. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 10819–10829

38. Koonce B, Koonce B (2021) MobileNetV3. Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization, pp 125–144
39. Goldberger J, Hinton GE, Roweis S, Salakhutdinov RR (2004) Neighbourhood components analysis. Adv Neural Inf Process Syst 17:513–520
40. Frazier PI (2018) A tutorial on Bayesian optimization. arXiv preprint arXiv:180702811
41. Warrens MJ (2008) On the equivalence of Cohen's kappa and the Hubert-Arabie adjusted Rand index. J Classif 25(2):177–183
42. Loh HW, Ooi CP, Seoni S, Barua PD, Molinari F, Acharya UR (2022) Application of explainable artificial intelligence for healthcare: a systematic review of the last decade (2011–2022). Comput Methods Prog Biomed 226:107161
43. Jahmunah V, Ng EYK, Tan R-S, Oh SL, Acharya UR (2022) Explainable detection of myocardial infarction using deep learning models with Grad-CAM technique on ECG signals. Comput Biol Med 146:105550
44. Naito S, Tomozawa H, Mori Y, Nagata T, Monma N, Nakamura H, Fujiwara H, Shoji G (2020) Building-damage detection method based on machine learning utilizing aerial photographs of the Kumamoto earthquake. Earthq Spectra 36(3):1166–1187
45. Stramondo S, Bignami C, Chini M, Pierdicca N, Tertulliani A (2006) Satellite radar and optical remote sensing for earthquake damage detection: results from different case studies. Int J Remote Sens 27(20):4433–4447
46. Abdi G, Esfandiari M, Jabari S (2021) Building damage detection in post-event high-resolution imagery using deep transfer learning. In: 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS. IEEE, pp 531–534
47. Liu C, Sui H, Wang J, Ni Z, Ge L (2022) Real-time ground-level building damage detection based on lightweight and accurate YOLOv5 using terrestrial images. Remote Sens 14(12):2763
48. Seydi ST, Rastiveis H, Kalantar B, Halin AA, Ueda N (2022) BDD-Net: An end-to-end multiscale residual CNN for earthquake-induced building damage detection. Remote Sens 14(9):2214
49. Lin Q, Ci T, Wang L, Mondal SK, Yin H, Wang Y (2022) Transfer learning for improving seismic building damage assessment. Remote Sens 14(1):201
50. Zhan Y, Liu W, Maruyama Y (2022) Damaged building extraction using modified Mask R-CNN model using post-event aerial images of the 2016 Kumamoto earthquake. Remote Sens 14(4):1002

Authors and Affiliations

**Mehmet Aydin[1] · Prabal Datta Barua[2] · Sreenivasulu Chadalavada[3] ·
Sengul Dogan[1] ⓘ · Turker Tuncer[1] · Subrata Chakraborty[4,5] · Rajendra U. Acharya[6]**

✉ Sengul Dogan
sdogan@firat.edu.tr

Mehmet Aydin
222144104@firat.edu.tr

Prabal Datta Barua
Prabal.Barua@usq.edu.au

Sreenivasulu Chadalavada
Sreeni.Chadalavada@usq.edu.au

Turker Tuncer
turkertuncer@firat.edu.tr

Subrata Chakraborty
Subrata.Chakraborty@uts.edu.au

Rajendra U. Acharya
Rajendra.Acharya@usq.edu.au

[1] Department of Digital Forensics Engineering, Technology Faculty, Firat University, Elazig, Türkiye

[2] School of Business (Information System), University of Southern Queensland, Toowoomba, Australia

[3] School of Engineering, University of Southern Queensland, Springfield Lakes, QLD 4300, Australia

[4] School of Science and Technology, Faculty of Science, Agriculture, Business and Law, University of New England, Armidale, NSW 2351, Australia

[5] Center for Advanced Modelling and Geospatial Information Systems, Faculty of Engineering and IT, University of Technology Sydney, Sydney, NSW 2007, Australia

[6] School of Mathematics, Physics and Computing, University of Southern Queensland, Toowoomba, Australia