Konya Mühendislik Bilimleri Dergisi., c. 13, s. 4, 1158-1174, 2025
Konya Journal of Engineering Sciences v. 13, n. 4, 1158-1174, 2025
ISSN: 2667-8055 (Electronic)
DOI: 10.36306/konjes.1628600

DETECTION AND PREDICTION OF CONCRETE CRACKS USING DEEP LEARNING-BASED IMAGE PROCESSING METHODS FOR QUALITY CONTROL

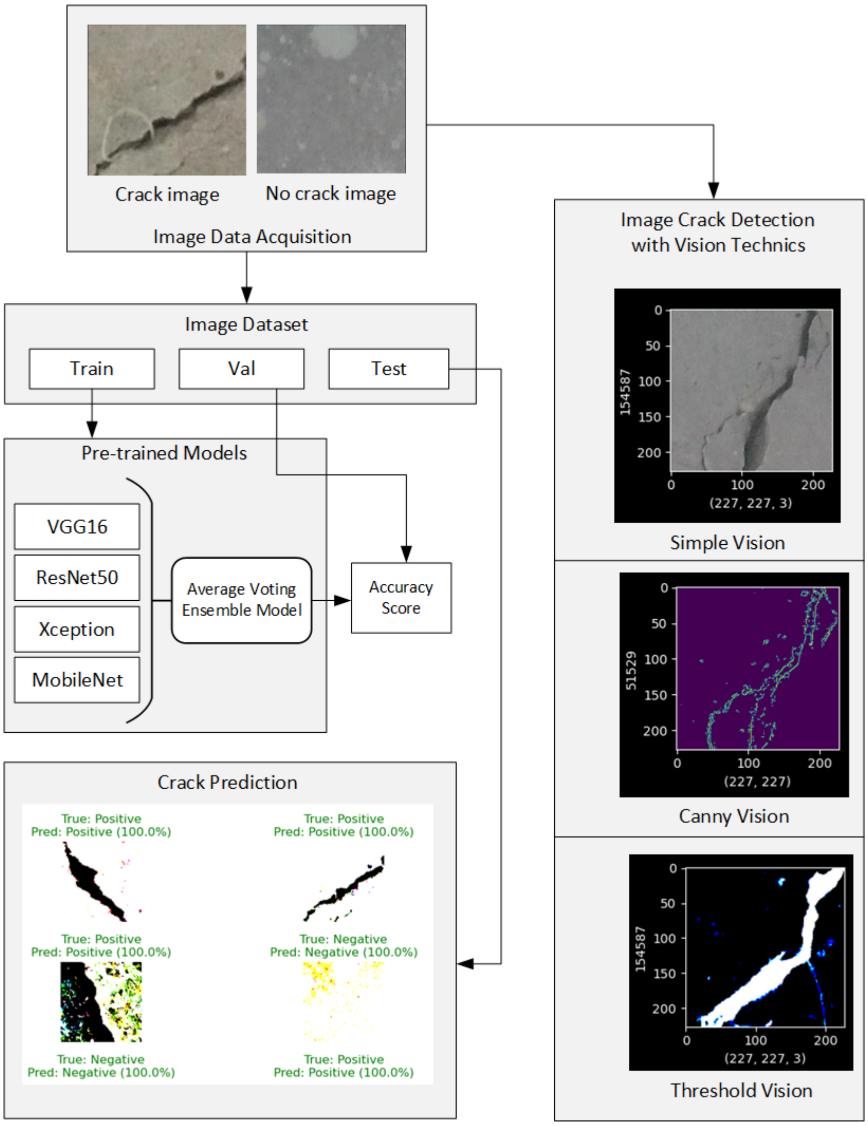
[1,*] İbrahim KARATAŞ

[1] *Osmaniye Korkut Ata University, Civil Engineering Department, Osmaniye, TÜRKİYE*
[1] ibrahimkaratas@osmaniye.edu.tr

Highlights

- Detection of Concrete Cracks using Vision Technics
- Prediction of Concrete Cracks using Voting Ensemble Model
- Cracks occurring on concrete surfaces can be predicted with minimal error.

Graphical Abstract



Flowchart of the method proposed in this research

***Corresponding Author:** İbrahim KARATAŞ, ibrahimkaratas@osmaniye.edu.tr

Konya Mühendislik Bilimleri Dergisi., c. 13, s. 4, 1158-1174, 2025
Konya Journal of Engineering Sciences v. 13, n. 4, 1158-1174, 2025
ISSN: 2667-8055 (Electronic)
DOI: 10.36306/konjes.1628600

DETECTION AND PREDICTION OF CONCRETE CRACKS USING DEEP LEARNING-BASED IMAGE PROCESSING METHODS FOR QUALITY CONTROL

[1,*] **İbrahim KARATAŞ**

[1] *Osmaniye Korkut Ata University, Civil Engineering Department, Osmaniye, TÜRKİYE*
[1] ibrahimkaratas@osmaniye.edu.tr

(Received: 28.01.2025; Accepted in Revised Form: 01.08.2025)

ABSTRACT: One of the most critical defects in the quality control process of concrete elements is the detection of cracks. Furthermore, cracks are among the most significant indicators affecting concrete strength. Manual crack detection presents numerous disadvantages in terms of time, labor, cost, high error probability, and practical implementation challenges. Therefore, this study aims to detect cracks on concrete surfaces using vision techniques and automatically predict them using deep learning methods. Images classified as crack and non-crack, selected from a dataset obtained from the literature, were initially analyzed using Canny and Threshold methods. Subsequently, analyses were conducted using a novel voting ensemble model that combines deep learning models such as VGG16, ResNet50, Xception, and MobileNet. According to the results, cracks were successfully detected using vision techniques, and the proposed voting ensemble model achieved an accuracy value of 99.75% with a loss value of 0.00618. The findings demonstrate that automated quality control of concrete surfaces specifically for cracks can be performed with high accuracy.

Keywords: *Concrete Cracks, Quality Control, Vision Technics, Pre-train Models, Voting Ensemble Model*

1. INTRODUCTION

The quality of concrete, one of the most commonly used construction materials in the building industry, is crucial for the durability and quality of constructed structures. Throughout a structure's lifecycle, concrete materials can be damaged under mechanical, physical, and chemical effects due to constantly changing conditions. One of the most significant defects and problems that occur in concrete under these effects is cracking [1].

Cracks can develop rapidly due to material aging and exposure to intense loads. They frequently form and propagate both on the surface and within the structure, ultimately leading to structural damage and failure. As cracks develop, they tend to reduce the effective loading area, which leads to increased stress and consequently diminishes the structure's load-bearing capacity [2]. The formation and density of cracks are also considered primary indicators of concrete maintenance requirements [1]. Early detection of cracks is vital for concrete maintenance needs and structural durability and safety, particularly in regions with high seismic risk such as Turkey [3]. Concrete cracks can occur due to various factors such as temperature, pressure, or changes in concrete quality on concrete surfaces. Detecting these cracks is crucial, as unidentified and unrepaired cracks can compromise the safety of buildings or roads [4]. Manual crack detection is typically disadvantageous in terms of time, labor, cost, high error probability, and implementation challenges. Literature indicates that in recent years, research has increasingly focused on using image processing techniques, machine learning, and deep learning-based algorithms as alternatives to manual detection.

Kim and Cho (2018) demonstrated the effectiveness of deep learning for automatic crack detection, highlighting its potential for enhanced accuracy and efficiency in real-world applications. They successfully developed an automatic crack detection method using CNN, one of the deep learning methods [5]. Rajadurai and Kang (2021) focused on image-based crack detection algorithms based on deep convolutional neural networks that detect and classify cracks with higher classification rates using transfer learning. An image dataset consisting of two consecutive image classes (non-cracked and

***Corresponding Author:** İbrahim KARATAŞ, ibrahimkaratas@osmaniye.edu.tr

cracked) was trained using the AlexNet model. The results yielded high prediction accuracy for both image classes [6]. Zhou et al. (2022) proposed a structural surface crack detection and measurement method based on images captured by Unmanned Aerial Vehicles (UAV) to perform remote visual inspection of cracks in inaccessible parts of large crane structures. A Faster Region-Based Convolutional Neural Network (R-CNN) algorithm was used to classify and detect cracks. The results demonstrated that the proposed method could meet the requirements for automatic detection and measurement of crane surface cracks under complex backgrounds. The prediction accuracy was determined to be 95.4% [7]. Feng et al. (2020) attempted to predict cracks on dam surfaces using a CNN deep learning model. During the data collection phase, images on the dam surface were collected using a predetermined trajectory and subsequently prepared for the model. Crack regions were manually labeled on cropped images to create the crack dataset, and the CNN network architecture was designed. According to the results obtained for crack detection, recall, sensitivity, F-score, and IoU values were determined as 80.45%, 80.31%, 79.16%, and 66.76%, respectively [8]. Bandi et al. (2024) conducted crack prediction research using ResNet-50, which is part of the deep CNN network containing 50 layers, utilizing approximately 56,000 annotated images consisting of both cracked and uncracked bridge deck images, wall images, and pavement images. According to the results obtained, they predicted cracks with approximately 80% accuracy [9]. Abualigah et al. (2024) emphasized the challenges of finding cracks in concrete. Particularly, the way light reflects on concrete and how smooth or rough the surface is can make it difficult to detect cracks [4]. To assist in finding cracks, researchers can easily identify important details or features from concrete images through models that have already learned to recognize patterns using pre-trained models. This enables detailed visualization of even small cracks that are difficult to detect with the naked eye.

While numerous studies have focused on automated crack detection using various image processing and machine learning techniques, this research makes several unique contributions to the field:

1. **Integration of classical and modern approaches:** Unlike many studies that focus exclusively on either traditional image processing or deep learning methods, this work combines Canny and Threshold techniques with advanced deep learning models to leverage the strengths of both approaches.
2. **Novel voting ensemble architecture:** This study introduces a robust voting ensemble model that integrates multiple pre-trained deep learning architectures (VGG16, ResNet50, Xception, and MobileNet), addressing the limitations of single-model approaches and significantly improving prediction accuracy to 99.75%.
3. **Practical quality control framework:** The methodology developed in this research is specifically designed for integration into concrete quality control processes, providing a realistic pathway for practical implementation in construction environments.
4. **Comparative analysis:** This study provides comprehensive comparative insights between traditional image processing methods and advanced deep learning techniques for crack detection, offering valuable guidance for practitioners on method selection based on specific requirements.

This research aims to develop a highly accurate, automated system for detecting and predicting cracks on concrete surfaces using a combination of vision techniques and deep learning methods, ultimately enhancing quality control processes in concrete construction and inspection.

2. MATERIAL AND METHODS

The methodology used in this study to determine the presence of cracks on concrete surfaces during quality control processes is illustrated in Figure 1. According to Figure 1, initially, images of cracked and non-cracked concrete were obtained from the literature. These acquired images were utilized in two phases. In the first phase, image processing techniques were employed for the visual detection of these cracks. In the second phase, deep learning methods were investigated to predict the presence of cracks on the concrete surface in the images. All analyses conducted within the scope of this study are explained under the following headings.

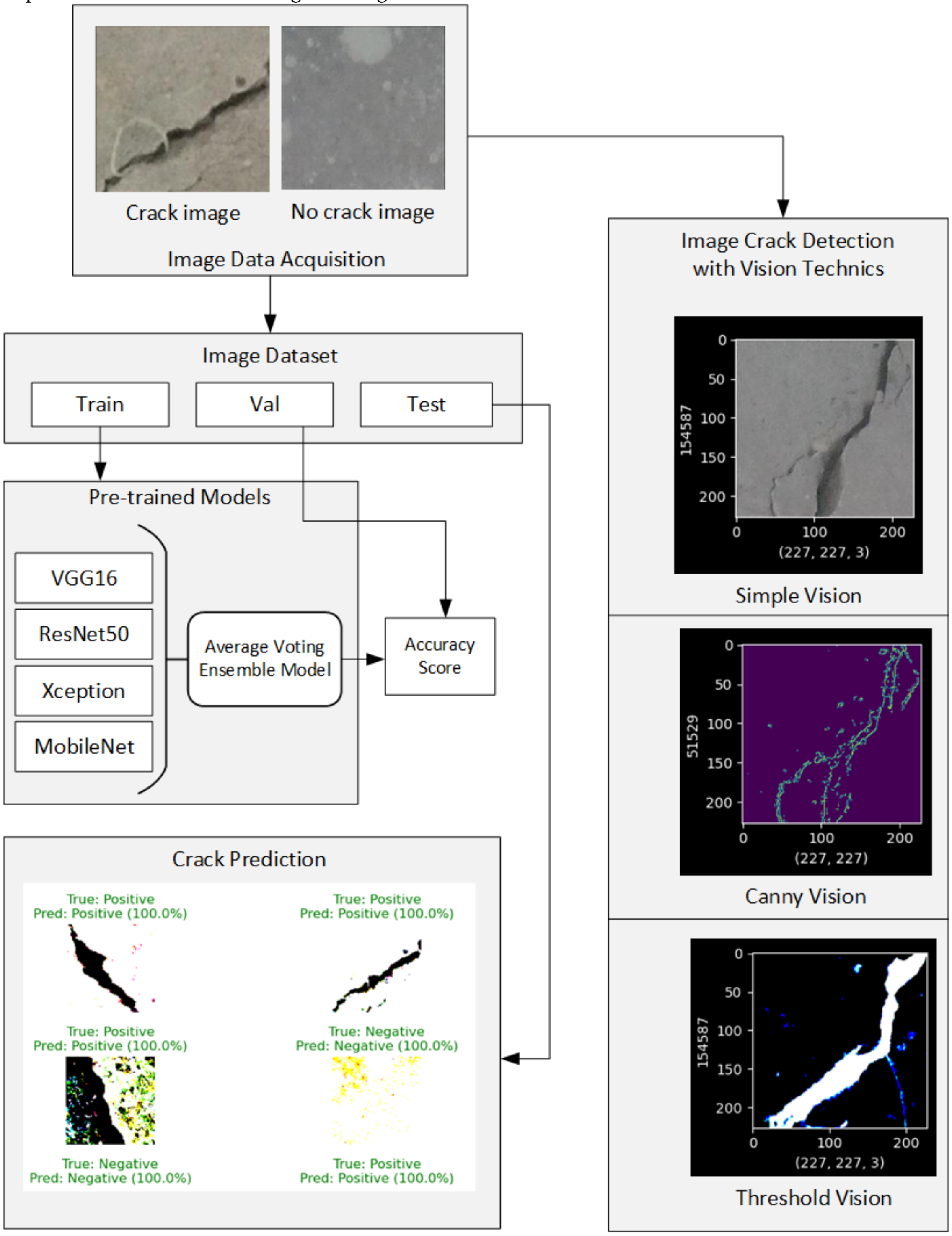


Figure 1. Flowchart of the method proposed in this research.

2.1. Image Dataset

For crack detection and prediction in this study, the METU dataset collected at the Middle East Technical University campus was selected [10]. The image dataset consists of 20,000 images equally divided between cracked and non-cracked classes, with an input image size of 227 × 227 × 3 pixels. However, for this study, a subset of 2,000 images was selected, comprising 1,000 images each of cracked and non-cracked specimens. This reduction in dataset size was necessary due to the computational demands and time constraints associated with analyzing the full 20,000 images. Specifically, 2,000 images were selected from the METU dataset to optimize computational efficiency. This sample size was determined to be optimal for achieving prediction accuracy while mitigating concerns about overfitting. To address potential overfitting issues with the limited dataset, pre-trained models and an ensemble voting approach were employed, with validation performed using an independent test set comprising 20% of the images completely isolated from the training process. Additionally, our transfer learning approach utilizing pre-trained models compensates for the limitations of a smaller dataset by leveraging robust feature representations learned from substantially larger image collections. These images, labeled as Positive and Negative, were initially divided into training and testing sets using an 80-20 split for analysis. Subsequently, the training set was further subdivided into training and validation sets using another 80-20 split, resulting in 1,280, 320, and 400 images for training, validation, and testing sets, respectively, prepared for model implementation.

2.2. Image crack detection with Vision Technics

Simple vision, Canny vision, and threshold vision techniques are frequently employed in image processing projects [1,7,11-14]. Simple vision techniques incorporate fundamental image processing steps such as filtering and contrast enhancement to improve the visibility of cracks against the concrete background. This technique utilizes preprocessing stages to reduce noise and enhance image quality [13].

The Canny vision technique is a widely adopted method for edge detection in images, particularly effective in identifying crack boundaries. This technique incorporates various steps including noise reduction, gradient calculation, non-maximum suppression, and hysteresis with edge tracking, which collectively facilitate accurate identification of crack edges [12,13]. The Canny technique employs a multi-stage algorithm for edge detection to identify a broad spectrum of edges in images. However, it exhibits certain limitations as it struggles to detect weak edges and distinguish subtle changes in grayscale, resulting in discontinuous detected edges [1]. The steps implemented for the Canny edge detection algorithm are briefly as follows:

- Noise reduction in the image is typically accomplished using a Gaussian filter. The Gaussian filter reduces high-frequency noise by blurring the image, resulting in a smoother image.
- Subsequently, the gradient is calculated to determine intensity variations in the image. The gradient represents the rates of change in horizontal and vertical directions for each pixel in the image. This calculation is performed using Sobel operators.
- After gradient calculation, the gradient magnitude and direction are determined for each pixel. The gradient magnitude indicates the edge strength, while the gradient direction indicates the edge orientation.
- Non-Maximum Suppression (NMS) is employed to ensure edges are thinner and more distinct, checking whether each pixel is a local maximum in its gradient direction and zeroing non-local maximum pixels.
- Finally, edges are classified as strong or weak using two different threshold values. Strong edges are pixels that exceed the upper threshold and are considered definitive edges, while weak edges are pixels that exceed the lower threshold but not the upper threshold.

The threshold vision technique is utilized to segment images into cracked and non-cracked regions by converting grayscale images into binary images based on a predetermined threshold value. Adaptive thresholding can be particularly effective under varying lighting conditions, providing more precise crack detection by dynamically adjusting the threshold according to local image characteristics [12]. Threshold is a classical image segmentation method that enables the segmentation of crack regions from the background. The threshold method implementation follows these concise steps [11].

- Three-channel RGB images are initially converted to single-channel grayscale images, where each pixel possesses a value between 0 and 255.
- Subsequently, pixels are classified as either target or background based on a predefined threshold.

- Binarization is a method that divides pixels into two categories in threshold segmentation, while multi-threshold segmentation divides pixels into multiple categories. This process separates the image into two levels (black and white). This method aids in image simplification for applications such as object detection and segmentation.

In this study, crack detection was performed visually using these techniques. The most suitable values for crack detection were determined by experimenting with different upper and lower threshold values for the Canny method and various threshold values for the threshold method.

2.3. Image Crack Prediction

2.3.1 Pre-Trained Models

In this study, pre-trained models frequently cited in literature and utilized across various applications were employed for crack predictions. These models are particularly significant as they retain features previously learned from comprehensive datasets. The study utilized pre-trained deep learning models including VGG16, ResNet50, Xception, and MobileNet. The selection of these models is crucial as each brings distinct capabilities in feature extraction. For instance, Xception, an advanced version of the Inception architecture, excels in identifying complex patterns with its 98 MB parameters. ResNet50, with its deeper architecture comprising 81 layers, offers enhanced performance due to its size and number of parameters [4]. In this study, these pre-trained models were combined using a voting ensemble model to achieve high prediction accuracy.

VGG16 is a convolutional neural network (CNN) model with 16 layers deep, consisting of 138,357,544 parameters. It features a simple and uniform architecture utilizing small receptive fields (3x3 convolutional layers) throughout the network. In terms of crack detection, it is used to extract features from images processed by other components of the detection system. It is also employed in various automated systems for detecting cracks in concrete surfaces. Its convolutional operations facilitate accurate crack detection and classification by helping create feature maps crucial for identifying regions with high target probability [4,6].

ResNet50 is a 50-layer deep convolutional neural network designed to address the gradient problem. It incorporates skip connections that allow gradients to flow more effectively through the network during training. Due to its robust feature extraction capabilities and architecture's ability to capture complex patterns and details in images, it is frequently used for various applications including image classification and segmentation. The model is typically pre-trained on large datasets, which enhances its ability to generalize to new tasks. This enables the model to be optimized for specific applications such as concrete crack prediction [4,15].

Xception is a convolutional neural network model built upon the Inception architecture by replacing standard convolutions with depthwise separable convolutions. This design choice enhances the model's efficiency and performance in feature extraction tasks. Due to its feature extraction capabilities, it has become a popular model for tasks such as segmentation and image classification [16,17].

MobileNet is a lightweight convolutional neural network designed for mobile and embedded vision applications. It utilizes depthwise separable convolutions to reduce the number of parameters and computational cost. Despite its lightweight architecture, MobileNet can extract robust features, making it suitable for tasks such as image classification and segmentation. Its architecture allows for a balance between accuracy and efficiency, and has been effectively utilized in semantic segmentation tasks, providing clear and accurate results [18].

In this study, these four models were combined using a Voting ensemble model to achieve enhanced prediction performance. The Voting ensemble method is based on a voting scheme that combines the pre-trained models described above to achieve better performance. It combines predictions by assuming the model with the most votes is the winner [19]. This method prevents various errors from these previously trained models in concrete crack prediction, prevents overfitting, and increases prediction accuracy [20]. The Average voting ensemble model is a common approach where predictions from multiple models are combined to make a final decision, typically leading to better accuracy and robustness. This method is based on an average voting strategy where each model in the ensemble contributes to its prediction, ultimately resulting in a final prediction determined by the collective majority average vote among these models [21]. Although weighted voting and meta-learning approaches were evaluated in the background of the study, simple average voting was determined to provide superior performance with significantly less complexity. Since computational efficiency was prioritized for practical applications in construction environments, the average voting method provides the most efficient approach with minimal processing overhead, making it more suitable for potential real-time applications in the field. Additionally, the average voting method facilitates the integration of additional models or updating of existing ones without the complex retraining of meta-learners when analyses need to be repeated with new data. In summary, the voting ensemble technique was chosen to address the limitations of individual models by combining their strengths, reducing their weaknesses, and providing a more robust and accurate solution for concrete crack prediction. The ensemble approach leverages the collective intelligence of multiple models to enhance diagnostic reliability and performance.

2.3.2 Model Evaluation Metrics

After preparing the dataset and training the combined pre-trained models with the voting ensemble model, the results were compared with test data, and the model's loss and accuracy values were used to evaluate its performance. Additionally, a confusion matrix was constructed to determine these values. This matrix provides information about which classes were correctly predicted and which were incorrectly predicted. According to the example error matrix shown in the figure, rows represent actual classes while columns represent predicted classes. Here, cells within the diagonal indicate correctly classified examples, while off-diagonal cells represent misclassified examples. Through these analyses, it becomes apparent which class examples were confused with or incorrectly predicted as other classes. Furthermore, common terms used to calculate other evaluation metrics are derived from this matrix. The terms shown here are calculated across four categories: true positive (TP), false positive (FP), true negative (TN), and false negative (FN), providing important insights into model performance. Accuracy values represent the percentage of correctly predicted values from the analysis, calculated according to the equation.

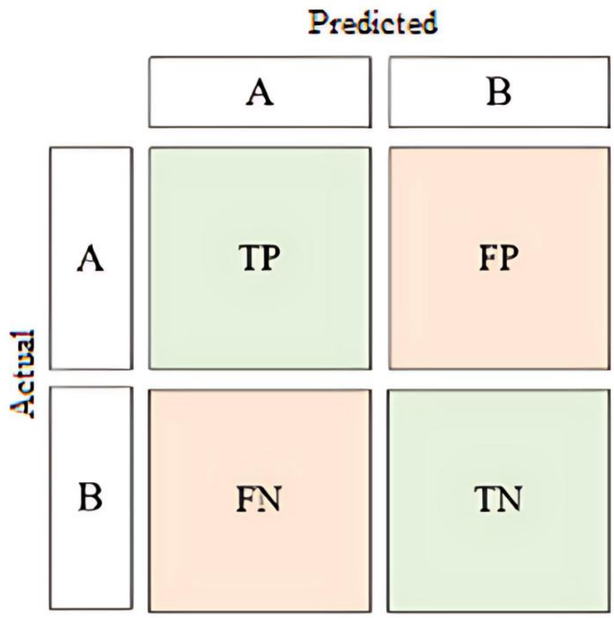


Figure 2. Confusion matrix structure

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

3. RESULTS AND DISCUSSION

In this study, concrete surface cracks were examined through two distinct approaches. Initially, the research focused on visual crack detection using vision techniques. Subsequently, pre-trained methods were combined with a voting ensemble model to investigate automatic crack prediction capabilities and determine prediction accuracy rates. The deep learning analyses were programmed and trained using Python 3.6 on a computer with 8GB RAM, 3.8-GHz Core-i7 embedded processor and CPU (central processing units). The results obtained from these analyses are presented under the following headings.

3.1. Results of the Image Crack Detection

In addition to normal basic visualization, Canny and threshold vision techniques were employed for detecting concrete cracks. Through these image processing techniques, attempts were made to identify cracks on concrete surfaces. Figure 3 displays 16 non-crack concrete surface images from the obtained dataset, with dimensions of (227, 227, 3) pixels. Figure 4 shows 16 crack concrete surface images with the same pixel dimensions. As evident in these figures, pixels containing cracks exhibit distinct color variations. Using Canny and threshold techniques, it became possible to highlight these color differences and detect cracks. All techniques were implemented using the OpenCV library for image processing and Matplotlib for data visualization.

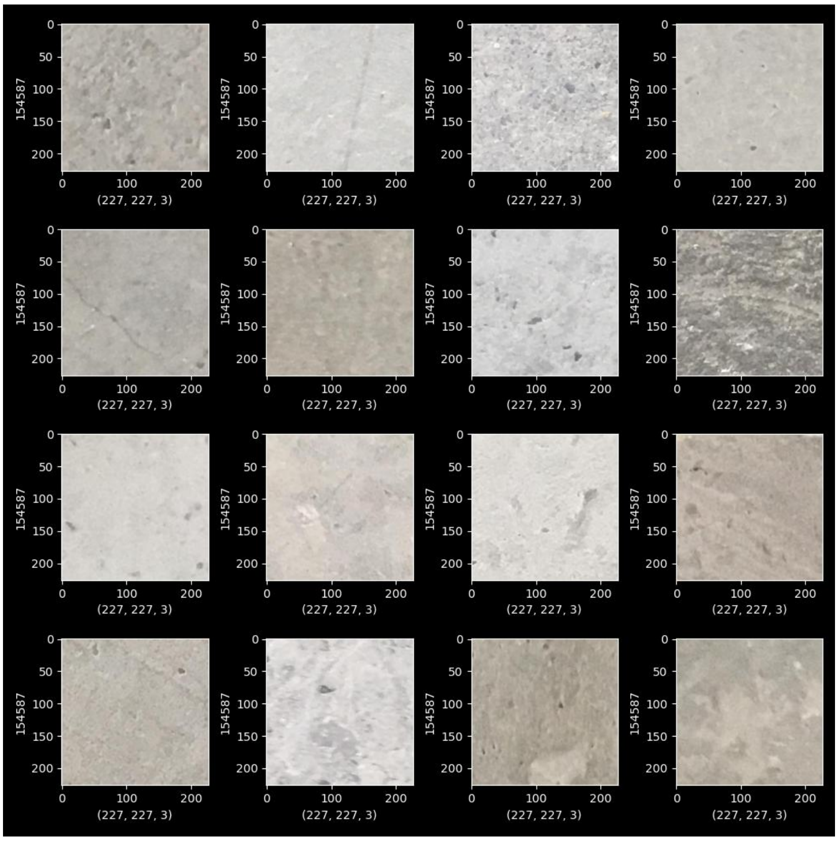


Figure 3. Simple vision negative (non-crack) samples

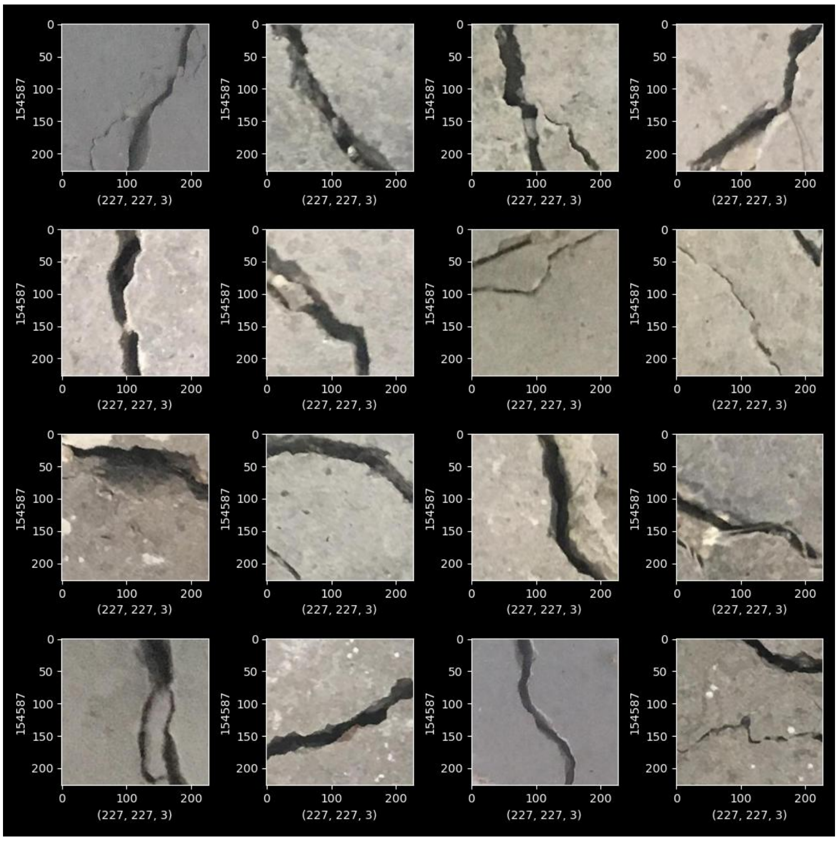


Figure 4. Simple vision positive (crack) samples

The Canny method, developed by John F. Canny in 1986, is a widely recognized technique for edge detection in image processing. This method is demonstrated in Figures 5 and 6, which depict crack-free concrete and concrete with cracks, respectively. As detailed in the methodology section, the Canny method necessitates the use of distinct upper and lower thresholds. In this study, after evaluating various combinations, we established a fixed lower threshold of 90 and an upper threshold of 100 as the most effective for detecting cracks in concrete. These thresholds were compared against those determined by the Otsu method. Values that fall below the lower threshold are capable of detecting less prominent edges, whereas values exceeding the upper threshold identify only more pronounced edges. Pixels that lie between the lower and upper thresholds are classified as edges only if they are adjacent to strong edge pixels. For the purposes of this study, the fixed thresholds of 90 and 100 were implemented. The resulting figures illustrate that, while some pixel variations may be evident on non-cracked concrete surfaces, these variations are not classified as cracks. Conversely, in images of cracked concrete, various edges are detected based on specific threshold lengths. Further analysis was conducted using Otsu's thresholding method to optimize the threshold values. Otsu's method automatically calculates the optimal threshold by minimizing the intra-class variance between foreground and background pixels. After applying this optimization technique, the new threshold values were determined to be 133 for the lower threshold and 265 for the upper threshold. These optimized values provide more accurate edge detection by better distinguishing between actual cracks and background noise in the concrete surface images. Comparative images using both the initial (90, 100) and optimized (133, 265) threshold values are presented in Figure 5, demonstrating the enhanced detection capability achieved through this optimization. The results indicate that the Otsu-optimized thresholds significantly improve the accuracy of crack detection while reducing false positives in non-crack surfaces.

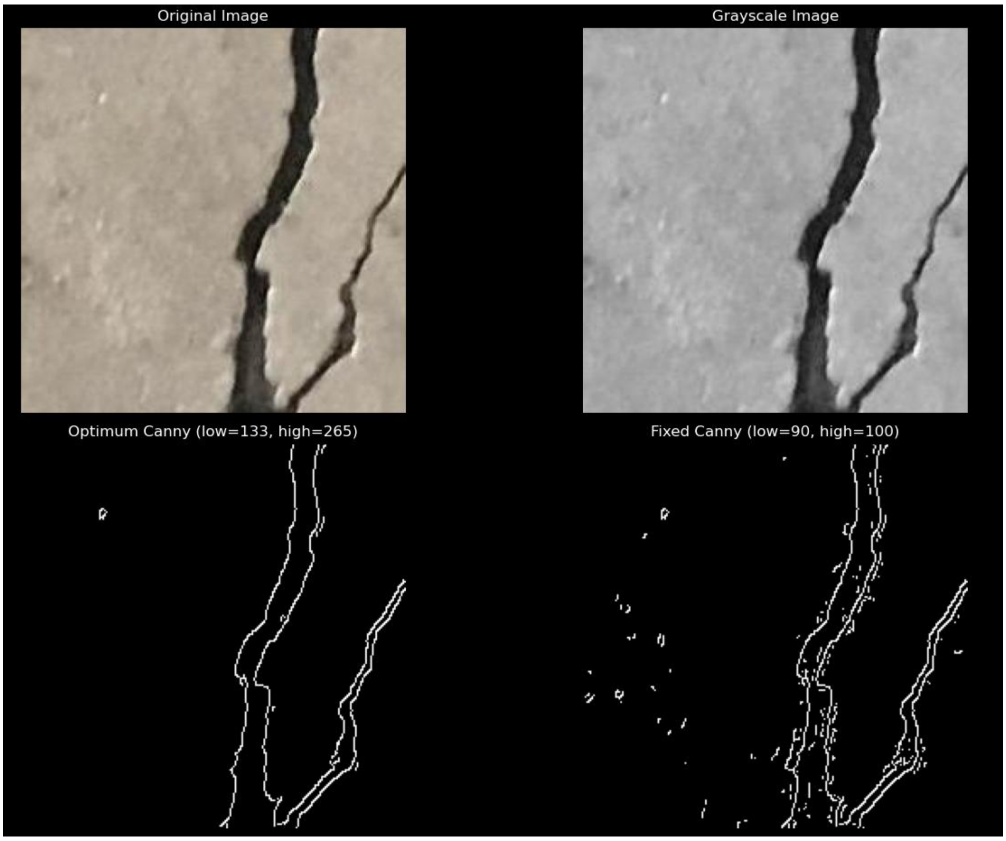


Figure 5. Comparison of Canny vision positive (cracked) samples with the Fixed and Otsu Method

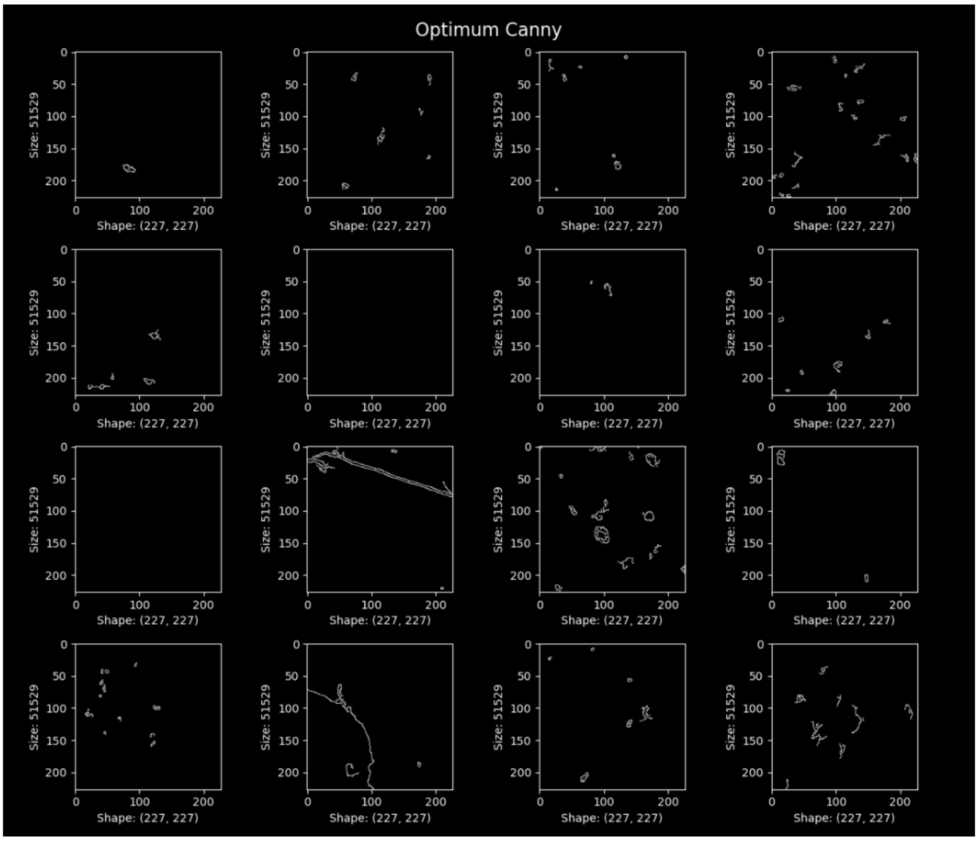


Figure 6. Canny vision negative (non-crack) samples

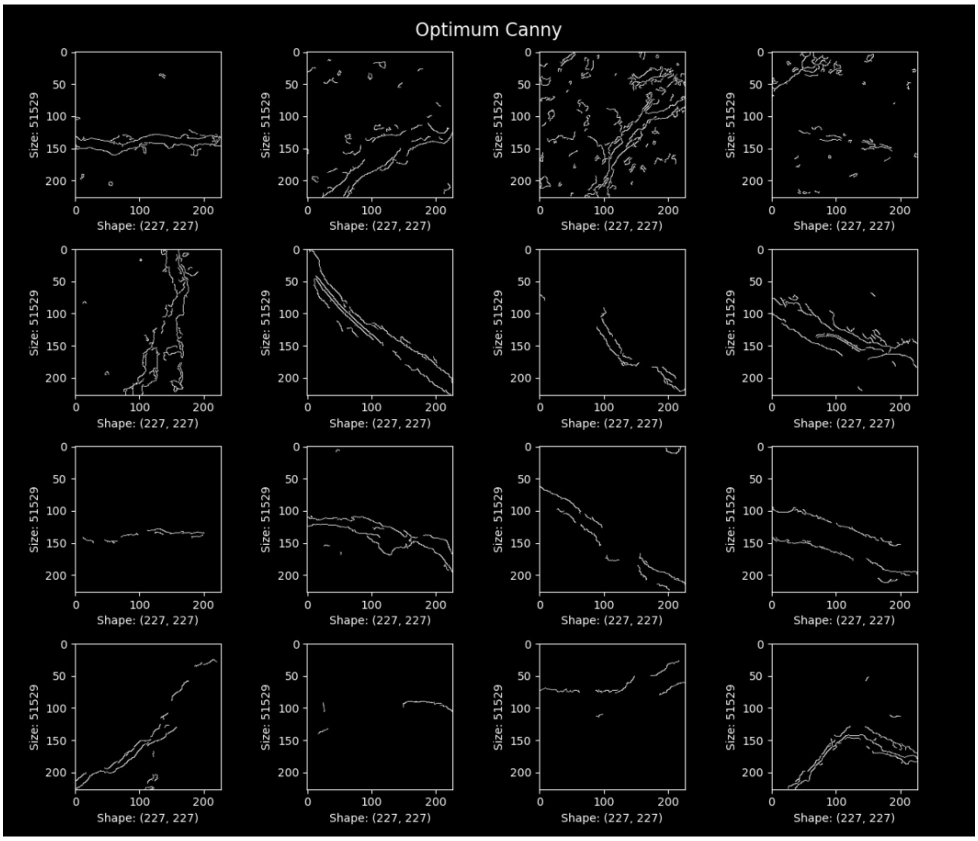


Figure 7. Canny vision positive (crack) samples

The threshold method, a widely used technique in digital image processing, simplifies an image by dividing it into two levels based on a specific threshold value. The results obtained using this method are shown in Figures 9 and 10 for non-crack and crack images, respectively. As detailed in the methodology section, implementing the threshold method requires determining a threshold value. In this study, after testing various combinations, the threshold value was set to 130 for concrete crack detection. Conversely, the fixed threshold value was validated through a comparative analysis with the Otsu optimization results. The Otsu optimization analysis indicated a threshold value of approximately 130. The findings of this comparison are illustrated in the accompanying Figure 8. This value can be used to separate objects from the background based on pixel intensity in the image. As demonstrated in the resulting figures, while some pixel variations are clearly visible on non-crack concrete surfaces, they are not identified as cracks. In crack images, pixels appear in white crack formations. However, the threshold value of 130 could not fully identify some crack images, appearing only as a white surface, as shown in Figure 8. Therefore, crack detection was not limited to vision techniques alone in this study. In the subsequent phase, crack prediction was achieved by combining pre-trained artificial intelligence methods using an ensemble method.

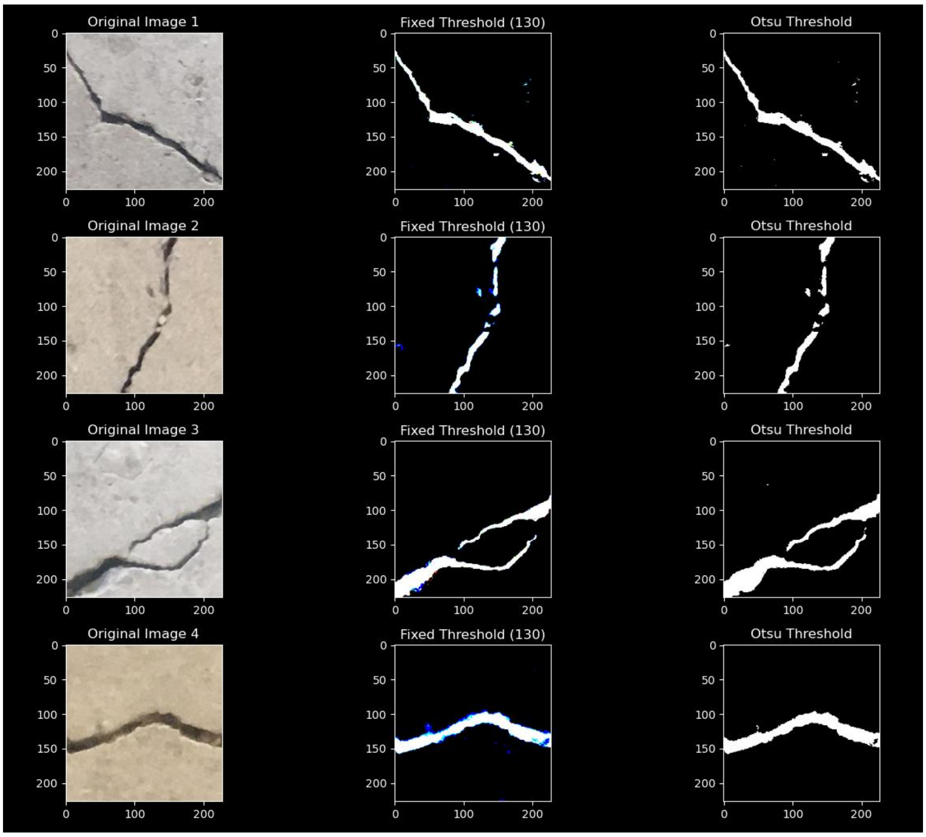


Figure 8. Comparison of Threshold vision positive (cracked) samples with the Fixed and Otsu Method

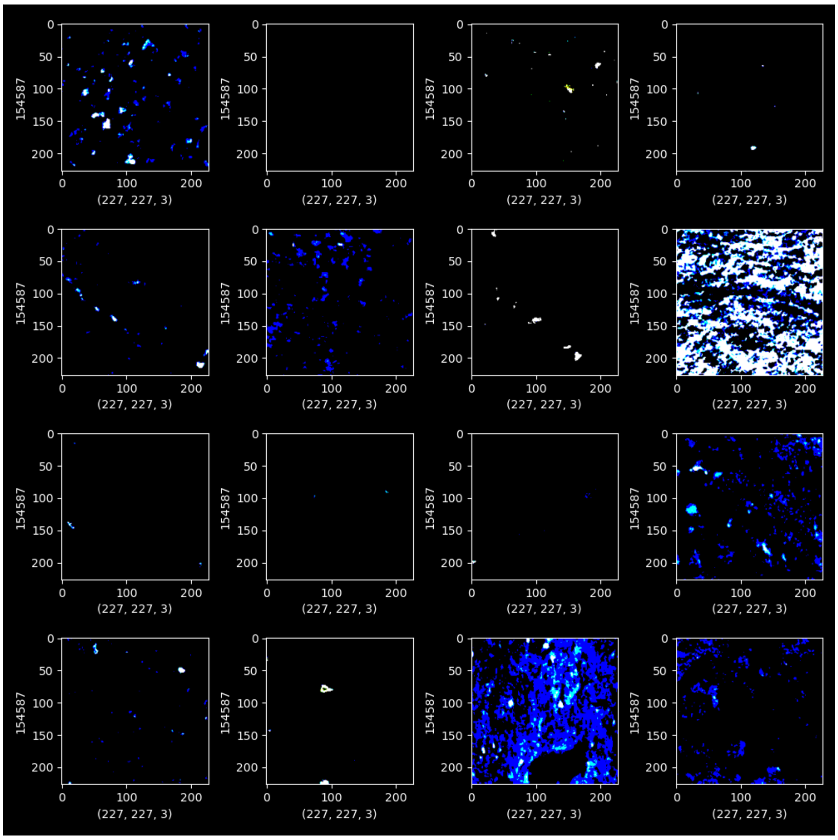


Figure 9. Threshold vision negative (non-crack) samples

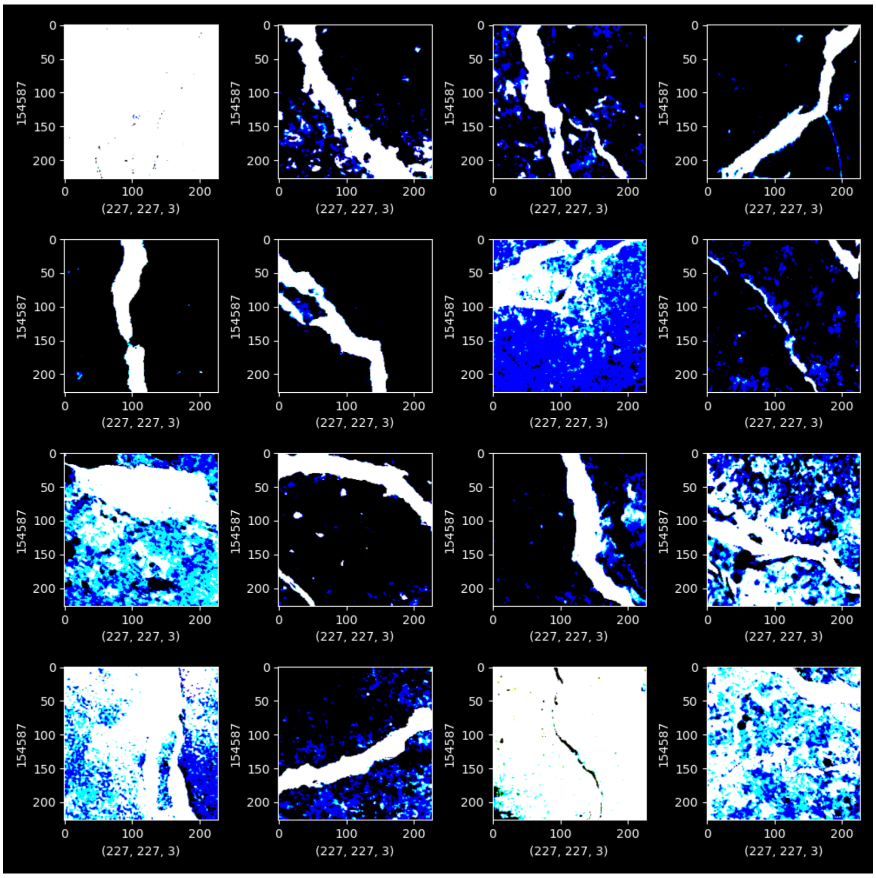


Figure 10. Threshold vision positive (crack) samples

3.2. Crack Prediction using Voting Ensemble Model

In this study, a voting ensemble model was created and analyzed by combining deep learning models such as VGG16, ResNet50, Xception, and MobileNet from pre-trained models. This approach integrated predictions from 4 models, resulting in improved accuracy and robustness. This method combines the strengths of individual models while minimizing their weaknesses, providing a more robust and accurate solution for concrete crack prediction. According to the analysis results, the accuracy value achieved in predicting concrete cracks was 99.75% with a loss value of 0.00618. Other evaluation metrics were precision 0.995, recall 1 and f1-score 0.9975. The confusion matrix of this highly accurate model is shown in Figure 11. Based on 400 test samples, only one image was incorrectly classified as having a crack when it did not, while all other images were correctly predicted. Figure 12 displays the outputs of the images analyzed using the Voting ensemble model. When images were input into this model, both the actual class and predicted class were simultaneously provided to determine the accuracy of predictions. To mitigate overfitting effectively, our study utilized pre-trained models. When used with a transfer learning approach, models pre-trained on large datasets such as ImageNet tend to demonstrate better generalization performance even under limited data conditions, as they have already learned general image features. These models were fine-tuned by adjusting only the upper layers when retrained to identify cracks on concrete surfaces. Consequently, since the model had previously learned general visual features, the risk of overfitting our specific dataset was reduced. Additionally, pre-trained models naturally mitigate overfitting in limited data conditions as they provide effective results with less data. Furthermore, our voting ensemble model provided more reliable results by balancing the overfitting tendency of any single model. While individual models might exhibit overfitting in different ways, collective predictions minimized such errors.

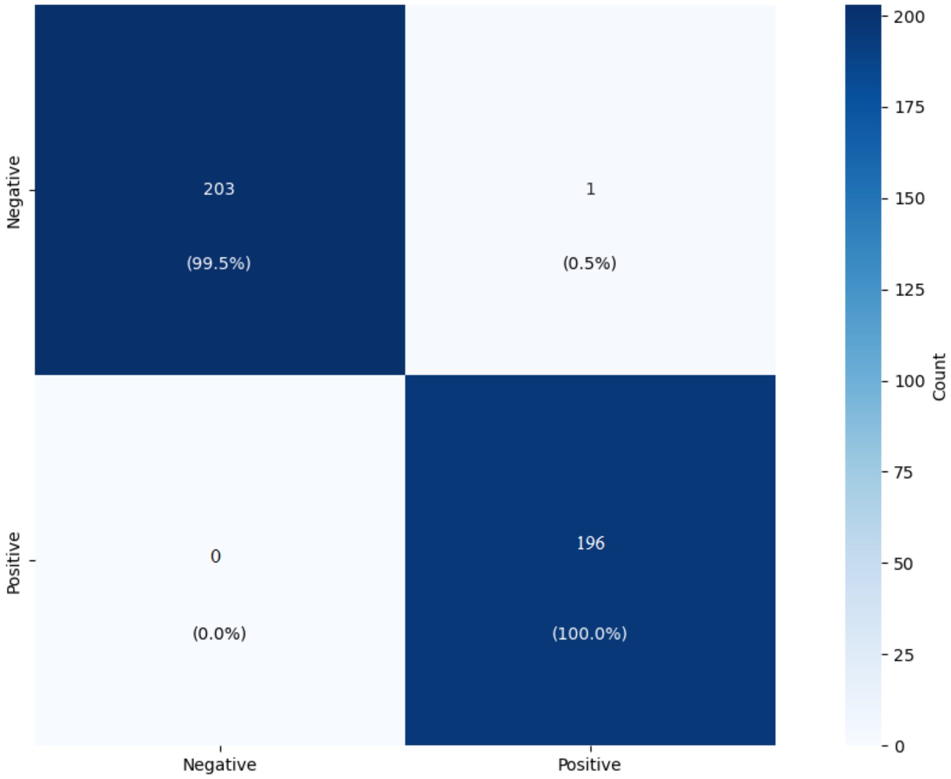


Figure 11. Voting Ensemble Model Confusion Matrix

Figure 12. Results of the Voting ensemble analysis

3.3. Discussion

With technological advancements, image processing and computer vision applications have become increasingly prevalent in the construction industry, accompanied by extensive scientific research. Previous studies on concrete crack detection have conducted separate analyses using deep learning models such as VGG16, ResNet50, Xception, and MobileNet. The prediction success rates of these analyses range between 80% and 99% [1,3,4,17,22]. Upon reviewing the literature, various deep learning models have been utilized for concrete crack detection. Studies by Khlifati et al. [1] and Abualigah et al. [4] examined automatic methods for detecting cracks in asphalt and concrete structures, evaluating their success rates. Altun and Altun [3] detected concrete surface cracks using the YOLOv8 deep learning algorithm. In research conducted by Karthik et al. [17], crack detection was performed using transfer learning-based deep feature extraction and support vector machines. The prediction success rates of deep learning models such as VGG16, ResNet50, Xception, and MobileNet used in these studies range between 80% and 99%. Sermet and Pachal [23] achieved approximately 99% prediction accuracy in their study on the METU dataset using the MobileNetV2 140 model. However, their analyses were processed rapidly on large datasets with GPU support. On CPU-based computers, the analysis of this model would be considerably slower. Similarly, Sevinc and Özyurt [24] utilized the METU dataset for crack detection, employing various deep learning architectures including MobileNet, VGG-16, ResNet-50, EfficientNet, Xception, Inception-V3, and ShuffleNet, and obtained prediction accuracies of approximately 99% across all architectures. The highest prediction accuracy was achieved with the MobileNet architecture. Our study demonstrated high prediction accuracy using 2,000 randomly selected samples from the METU dataset. Notably, by employing the Voting ensemble model, analyses can be performed quickly without requiring high-performance computing capabilities. Building upon these previous findings, the present study achieved a high accuracy rate of 99.75% by combining these models, which were used separately in previous research, with a Voting Ensemble model. This result provides a higher success rate compared to previous studies that used a single model. Additionally, this study differentiates itself from other literature by enabling both detection and prediction of cracks on concrete surfaces through the combined use of image processing techniques (Canny and Threshold) with deep learning methods. Cracks that form when concrete materials are not regularly maintained significantly reduce material strength and adversely affect the performance of reinforced concrete structures, particularly during earthquakes. Therefore, the detection and monitoring of cracks on concrete surfaces are extremely important for structural performance and safety.

4. CONCLUSIONS

In this study, Canny and threshold techniques were employed for crack detection, while a voting ensemble model was analyzed by combining deep learning models such as VGG16, ResNet50, Xception, and MobileNet from pre-trained models for crack prediction. An open-source image dataset containing two classes (non-crack and crack) was utilized. The image datasets were divided into training, validation, and test data for analysis.

Initially, Canny and Threshold methods were utilized for crack detection. Threshold values were established in these methods for clear crack detection. For the Canny method, the lower threshold value was set to 133, and the upper threshold value to 265. The resulting images are presented in the results section, demonstrating successful crack detection. Conversely, for the Threshold method, the threshold value was set to 130, and the generated images are presented accordingly. The condition and location of cracks were successfully detected based on pixel variations. However, it was determined that very fine cracks could not be detected in some cases. Therefore, crack prediction was performed using deep learning methods.

In this study, a more robust and reliable model was created by combining pre-trained deep learning models such as VGG16, ResNet50, Xception, and MobileNet. The training results of the new model created using the voting ensemble method achieved an accuracy value of 99.75% and a loss value of

0.00618. According to the results, cracks occurring on concrete surfaces can be predicted with minimal error.

The findings from this research have several significant practical implications for the construction industry and concrete quality control processes:

1. Implementation in quality control systems: The developed voting ensemble model can be integrated into existing quality control workflows in precast concrete factories and on-site concrete inspections, providing real-time automated crack detection with minimal human intervention.
2. Early detection benefits: By implementing this system during initial concrete curing and throughout the lifecycle of concrete structures, potential structural issues can be identified much earlier, significantly reducing repair costs and extending service life.
3. Specialized applications: The methodology is particularly valuable for critical infrastructure such as dams, bridges, and high-rise buildings, where crack detection is crucial for public safety and where manual inspection may be dangerous or impractical.
4. Integration with mobile technologies: The relatively lightweight computational requirements of the optimized model enable potential deployment on mobile devices, allowing field engineers to conduct immediate assessments using standard tablets or smartphones.

Based on our findings, we propose the following recommendations for practical implementation: Organizations should adopt a phased implementation approach, beginning with controlled environments such as prefabricated facilities before expanding to more variable field conditions; despite the model demonstrating 99.75% accuracy, personnel should receive training to understand the system's capabilities and limitations; standardized imaging protocols (lighting, distance, angle) should be developed to enhance field reliability; and future applications should consider integration with Building Information Modeling (BIM) systems to document crack detection history. Additionally, the applicability of image processing models can be investigated for detecting other defects in the concrete quality control process, such as corrosion, porosity, voids, and pitted surfaces.

Declaration of Ethical Standards

The paper is conducted in accordance with ethical standards.

Credit Authorship Contribution Statement

The author made a whole contribution to this study

Declaration of Competing Interest

The author declares that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding / Acknowledgements

No dedicated funding was received for this research from commercial, public or non-profit entities.

Data Availability

All data generated or analysed during this study are cited in this published paper.

5. REFERENCES

[1] O. Khlifati, K. Baba, and B. A. Tayeh, "Survey of automated crack detection methods for asphalt and concrete structures," Innov. Infrastruct. Solut., vol. 9, no. 11, 2024, doi: 10.1007/s41062-024-

01733-w.
[2] N. Karimi, M. Mishra, and P. B. Lourenço, "Automated Surface Crack Detection in Historical Constructions with Various Materials Using Deep Learning-Based YOLO Network," International Journal of Architectural Heritage Conservation, Analysis, and Restoration, 2024, doi: 10.1080/15583058.2024.2376177.
[3] M. G. Altun and A. H. Altun, "Beton Yüzey Çatlaklarının YOLOv8 Derin Öğrenme Algoritması ile Tespit Edilmesi," Çukurova Üniversitesi Mühendislik Fakültesi Dergisi, vol. 39, no. 3, pp. 667–678, 2024, doi: 10.21605/cukurovaumfd.1560104.
[4] S. M. Abualigah, A. F. Al-Naimi, G. Sachdeva, O. AlAmri, and L. Abualigah, "IDSDeep-CCD: intelligent decision support system based on deep learning for concrete cracks detection," Multimed Tools Appl, 2024, doi: 10.1007/s11042-024-18998-z.
[5] B. Kim and S. Cho, "Automated Vision-Based Detection of Cracks on Concrete Surfaces Using a Deep Learning Technique," Sensors (Basel, Switzerland), vol. 18, no. 10, 2018, doi: 10.3390/s18103452.
[6] R.-S. Rajadurai and S.-T. Kang, "Automated Vision-Based Crack Detection on Concrete Surfaces Using Deep Learning," Applied Sciences, vol. 11, no. 11, p. 5229, 2021, doi: 10.3390/app11115229.
[7] Q. Zhou, S. Ding, G. Qing, and J. Hu, "UAV vision detection method for crane surface cracks based on Faster R-CNN and image segmentation," J Civil Struct Health Monit, vol. 12, no. 4, pp. 845–855, 2022, doi: 10.1007/s13349-022-00577-1.
[8] C. Feng, H. Zhang, H. Wang, S. Wang, and Y. Li, "Automatic Pixel-Level Crack Detection on Dam Surface Using Deep Convolutional Network," Sensors (Basel, Switzerland), vol. 20, no. 7, 2020, doi: 10.3390/s20072069.
[9] M. R. Bandi, L. N. Pasupuleti, T. Das, and S. Guchhait, "Deep learning based damage detection of concrete structures," Asian J Civ Eng, 2024, doi: 10.1007/s42107-024-01106-9.
[10] F. Özgenel, Concrete Crack Images for Classification. Mendeley Data 2019, V2.
[11] Y. Ding et al., "Innovative computer vision-based full-scale timber element cracks detection, stitching, and quantification," Structural Health Monitoring, 2024, doi: 10.1177/14759217241258682.
[12] S.-M. Choi, H.-S. Cha, and S. Jiang, "Hybrid Data Augmentation for Enhanced Crack Detection in Building Construction," Buildings, vol. 14, no. 1929, pp. 1–32, 2024, doi: 10.3390/buildings14071929.
[13] R. Kirthiga and S. Elavenil, "A survey on crack detection in concrete surface using image processing and machine learning," J Build Rehabil, vol. 9, no. 1, 2024, doi: 10.1007/s41024-023-00371-6.
[14] S. Lee, M. Jeong, C.-S. Cho, J. Park, and S. Kwon, "Deep Learning-Based PC Member Crack Detection and Quality Inspection Support Technology for the Precise Construction of OSC Projects," Applied Sciences, vol. 12, no. 19, p. 9810, 2022, doi: 10.3390/app12199810.
[15] F. Song, B. Liu, and G. Yuan, "Pixel-Level Crack Identification for Bridge Concrete Structures Using Unmanned Aerial Vehicle Photography and Deep Learning," Structural Control and Health Monitoring, vol. 2024, no. 1, pp. 1–14, 2024, doi: 10.1155/2024/1299095.
[16] F. Chollet, Ed., Xception: Deep Learning With Depthwise Separable Convolutions, 2017.
[17] K. S. Bhalaji Kharthik et al., "Transfer learned deep feature based crack detection using support vector machine: a comparative study," Scientific reports, vol. 14, no. 1, p. 14517, 2024, doi: 10.1038/s41598-024-63767-5.
[18] H. Polat, S. Alpergin, and M. S. Özerdem, "Beton çatlakların derin öğrenme tabanlı semantik segmentasyonunda kodlayıcı değişkenlerinin karşılaştırmalı analizi," DÜMF MD, 2024, doi: 10.24012/dumf.1465724.
[19] A. Geron, Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems,O'Reilly, CA. 2017.
[20] I. Karatas and A. Budak, "Development and comparative of a new meta-ensemble machine

learning model in predicting construction labor productivity," ECAM, vol. 31, no. 3, pp. 1123–1144, 2024, doi: 10.1108/ECAM-08-2021-0692.

[21] M. M. Hossain, M. B. Arefin, F. Akhtar, and J. Blake, “Combining state-of-the-art pre-trained deep learning models: A noble approach for skin cancer detection using max voting ensemble”. Diagnostics, 14(1), 89, 2023.

[22] M. Mishra, V. Jain, S. K. Singh, and D. Maity, "Two-stage method based on the you only look once framework and image segmentation for crack detection in concrete structures," Archit. Struct. Constr., vol. 3, no. 4, pp. 429–446, 2023, doi: 10.1007/s44150-022-00060-x.

[23] F. Sermet, I. Pachal, “Deep Learning-Based of Concrete Cracks Using Mobilenet Architectures,” Süni İntellekt: Nəzəriyyədən Praktikaya, 259, 2024.

[24] A. Sevinç, F. Özyurt, “Detection of concrete surface cracks with deep learning architectures”. International Journal of Innovative Engineering Applications, 6(2), 2022, doi: 10.46460/ijiea.1098046