Journal of Science and Transport Technology
Journal homepage: https://jstt.vn/index.php/en

Enhancing concrete structure maintenance through automated crack detection: A computer vision approach

Nha Huu Nguyen[1], Thuy-Hien Thi Nguyen[2,*], Linh Gia Bui[2], Hai-Bang Ly[2]
[1]Port Authority of Inland Waterway Area 1, VIWA, Vietnam
[2]University of Transport Technology, Hanoi 100000, Vietnam

Article info
Type of article:
Original research paper

DOI:
https://doi.org/10.58845/jstt.utt.2024.en.4.3.11-23

***Corresponding author:**
Email address:
hienntt82@utt.edu.vn

Received: 09/07/2024
Revised: 28/07/2024
Accepted: 30/07/2024

Abstract: This paper presents the development of an Artificial Intelligence (AI) and Machine Learning (ML) model designed to detect cracks on concrete surfaces. The objective is to enhance the automation, precision, and performance of crack detection using the computer vision algorithm. Employing a ML approach and the YOLOv9 algorithm, this study developed a system to accurately identify concrete cracks from a diverse dataset. A total of 16,301 images of concrete surfaces, balanced between those with and without cracks, were utilized. The dataset was split into various sets with different ratios to ensure comprehensive model training. A transfer-learning methodology was employed to optimize the model's performance. The accuracy of the model was measured in each experiment to determine the optimal result. The most successful experiment resulted in a model with a mean Average Precision (mAP) of 94.6%, a Precision of 94.1%, and a Recall of 88.4%. These results demonstrate the effectiveness of AI and ML in concrete crack detection.

Keywords: Artificial intelligence; Concrete; Crack detection; Machine learning; Computer vision.

1. Introduction

Concrete is a composite material consisting of aggregate bonded with fluid cement that hardens over time. As the second-most-used substance globally after water and the most widely used building material, concrete plays a crucial role in construction and infrastructure [1]. A crack in concrete refers to a complete or partial separation of the material into two or more parts due to breaking or fracturing [2]. Surface cracks in concrete structures are critical indicators of structural damage and compromise its durability [3]. Detecting these cracks is essential for the inspection, diagnosis, and maintenance of concrete structures [4]. However, automatic crack detection presents significant challenges [5]. The importance of crack detection in concrete structures is growing, as it is vital for effective inspection, evaluation, and maintenance.

Manual visual inspection, the most commonly employed method in practice, is inefficient in terms of cost, time, accuracy, and safety [3]. Inspectors typically rely on their experience, skill, and engineering judgment to visually assess defects in concrete structures. This method can involve determining the optimal binarization parameters of commonly used image binarization techniques and conducting comparative analyses to identify their characteristics in crack detection. Optimal parameters are obtained by minimizing the discrepancy between crack widths measured by digital cameras and optical microscopes [3]. Monitoring is usually conducted by regularly evaluating the onset of surface cracks using optical methods or extensometers [6]. However, this process is inherently subjective, labor-intensive, time-consuming, and complicated by the need to access to numerous parts of complex structures [7].

Advancements in automated inspection techniques, particularly through AI and ML, are addressing these challenges by providing more objective, efficient, and comprehensive assessments of concrete structures. Automated systems can analyze large volumes of data with high precision, reducing the reliance on manual inspection and enabling more proactive maintenance strategies. These technologies facilitate continuous monitoring, early detection of potential issues, and improved allocation of resources for maintenance, ultimately enhancing the safety and longevity of concrete structures.

AI image recognition has become widely applied, with object detection being a fundamental problem in computer vision involving the recognition and localization of objects within an image [8]. The YOLOv3 algorithm has been employed for real-time detection, calculating the sizes of detected cracks based on the positions of projected laser beams on structural surfaces [9]. A method for real-time logo identification using a deep learning network architecture has also been developed, customizing the YOLO algorithm to simultaneously detect and identify logos in input color images. Experimental results with the popular FlickrLogos-47 dataset demonstrate that this approach achieves high accuracy, with the added benefits of simplicity, effectiveness, and fast execution time, meeting the requirements of real-time computing for logo recognition systems [10]. Furthermore, a method using ResNet-50 for feature extraction and non-maximum suppression (NMS) for selecting high-quality suggestion boxes showed increased accuracy in detecting improperly worn masks. This method enhances practical applications and improves epidemic prevention by accurately detecting mask usage through feature extraction and prediction frame generation [11]. Additionally, a novel real-time object detection model based on the YOLOV2 framework has been developed for detecting tiny vehicles in Automatic Driving Systems (ADS) and Driver Assistance Systems [12]. An adaptive learning model based on the YOLOV3 deep learning network has been applied in self-driving vehicle systems, traffic management, and vehicle flow measurement at critical locations and routes [13]. The YOLO model has also been utilized to develop the HandGun Detector-C500, which recognizes handguns through surveillance cameras to provide early warnings related to crimes involving firearms [14]. Moreover, automatic crack detection and segmentation of masonry structures have been achieved using YOLOV9-Seg 2 and edge detection techniques [15]. Cha et al. proposed a crack detection algorithm using deep learning with a Convolutional Neural Network (CNN), as well as algorithms for detecting multiple types of damage, including steel delamination, steel corrosion, and bolt corrosion, using a Faster Region-based CNN (R-CNN) [9],[10]. These advancements highlight the versatility and effectiveness of AI and deep learning algorithms in various applications, from structural health monitoring to traffic management and security, demonstrating their potential to enhance efficiency and accuracy in numerous fields.

In the field of concrete crack identification, several deep learning approaches have been proposed. Choi et al. introduced SDDNet, a deep learning network optimized for crack detection in images with diverse background features [16]. Zhang et al. developed CrackU-net, a state-of-the-art pixelwise crack detection architecture utilizing advanced deep convolutional neural network technology. This "U"-shaped model architecture, involving convolution, pooling, transpose convolution, and concatenation operations, surpasses traditional methods as well as fully convolutional network (FCN) and U-net models for pixelwise crack detection [17]. Kim et al. proposed a method using R-CNN for crack identification and a square-shaped marker for measuring the size of detected cracks [18]. Beckman et al. suggested a Faster Region-based Convolutional Neural Network (Faster R-CNN) method for detecting concrete spalling damage. This approach automatically performs damage quantification by processing depth data, identifying surfaces, and isolating damage after merging outputs from the Faster R-CNN with the depth stream of the sensor [19]. Park et al. proposed a validated system capable of successfully detecting cracks and estimating their sizes without prior knowledge or any installation [9]. Various published studies have employed different algorithms for object detection, with many utilizing versions of the YOLO algorithm . Some research focused specifically on pavement crack images and bridge inspection tasks, such as the model proposed by Zhang et al., which was trained and validated using 3,000 pavement crack images (2,400 for training and 600 for validation) with the Adam algorithm. CrackU-net achieved a performance of loss = 0.025, accuracy = 0.9901, precision = 0.9856, recall = 0.9798, and F-measure = 0.9842 with a learning rate of 10−2 [17]. Research has also confirmed that, to date, no deep learning method has been able to automatically detect concrete spalling. The trained Faster R-CNN presented an average precision (AP) of 90.79%, with volume quantifications showing a mean precision error (MPE) of 9.45% at distances ranging from 100cm to 250cm between the element and the sensor [19]. These advancements highlight the ongoing efforts to enhance the accuracy and efficiency of concrete crack detection through deep learning methodologies.

Concrete cracks found in structures vary in terms of their width, depth, orientation, and complexity due to the impact of environmental conditions and structural loads. These variations present challenges in accurately detecting and classifying cracks. The YOLO (You Only Look Once) algorithm, recognized for its real-time object detection capabilities, is a potential solution due to its capacity to process diverse object types and scales within a unified framework. Prior research has shown the effectiveness of YOLO in various object detection applications, and its utilization for concrete crack detection offers a chance to enhance accuracy and efficiency in real-world construction site inspections.

This study developed an AI computer vision model using YOLOv9 based on a database of 16,301 images collected from various sources. The paper is structured into five sections: Introduction, Database Description and Analysis, Machine Learning (ML) Methods, Results and Discussion, and Conclusions and Future Research Directions. This study contributes to the field by developing a robust and efficient AI model for concrete crack detection, utilizing the YOLOv9 algorithm and a diverse dataset of 16,301 images, and demonstrating its practical applicability for enhancing infrastructure maintenance.

2. Database description and analysis

The dataset comprised 16,301 images of various types of concrete cracks collected from multiple sources, with 1,233 images sourced from the internet and the remaining 15,068 images obtained from an open database repository [20]. The training process aimed to diversify the dataset to encompass a wide range of scenarios. The selected images included various types of cracks on concrete structures, considering factors such as scale, observation location, observation direction, and illumination level. Each image in the synthetic dataset was precisely annotated with crack labels and bounding boxes using the Roboflow platform, which is designed for data management and preparation in computer vision problems. The dataset was divided into three subsets: training, validation, and test sets. By default, the dataset was split in a 70-15-15 ratio, used during training to tune the model's hyperparameters and to monitor and evaluate the model's performance after the training and tuning process. Examples of the collected dataset are shown in Fig. 1, whereas the labeling process of images is shown in Fig. 2.

Fig. 1. Illustration of images collected in the dataset

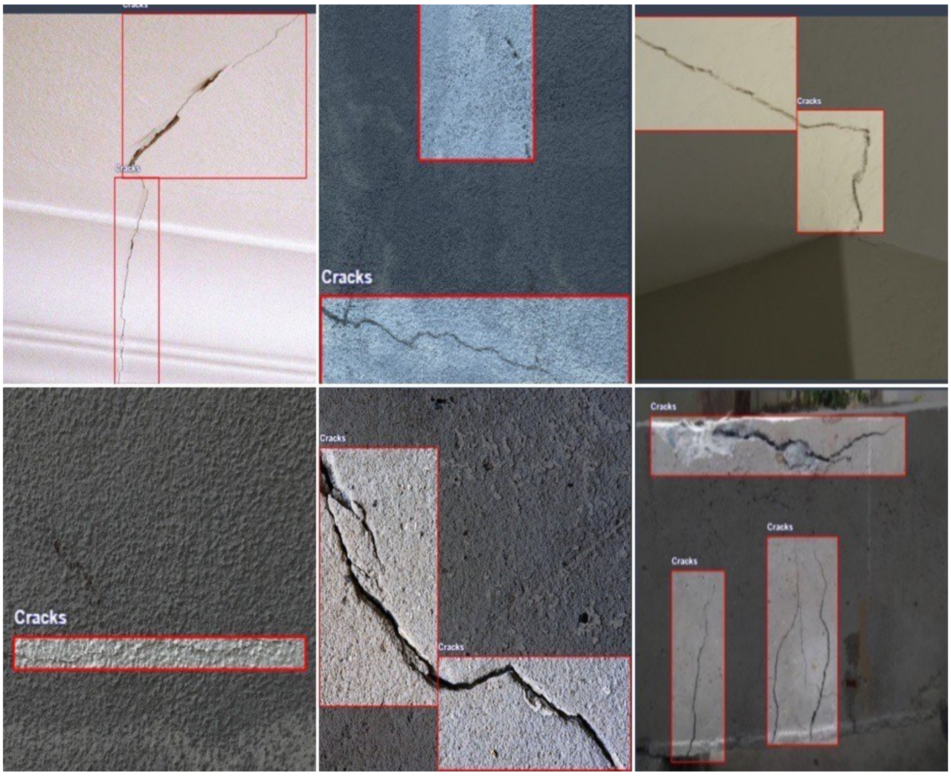


Fig. 2. Example of labeling process of images

3. Machine learning methods

3.1. YOLO overview

YOLO (You Only Look Once) is a unified, real-time object detector designed to address object detection as a regression problem, predicting spatially separated bounding boxes and associated class probabilities from full images in a single evaluation [8]. Since its introduction in 2015, YOLO has undergone several iterations, with the latest version, YOLOv9, released in 2024. This version has achieved high detection accuracy and fast inference times as a single-stage detector, surpassing many other object detection algorithms. Fig. 3 illustrates these advancements.

Developed by Chien-Yao Wang, I-Hau Yeh, and Hong-Yuan Mark Liao, YOLOv9 stands out as the fastest general-purpose object detector currently available, pushing the state-of-the-art in real-time object detection. Its ability to generalize well to new domains makes it ideal for applications requiring rapid and robust object detection [21]. Processing images with YOLO involves running a single convolutional network on the image and then thresholding the resulting detections based on the model's confidence. This process is illustrated in Fig. 4.

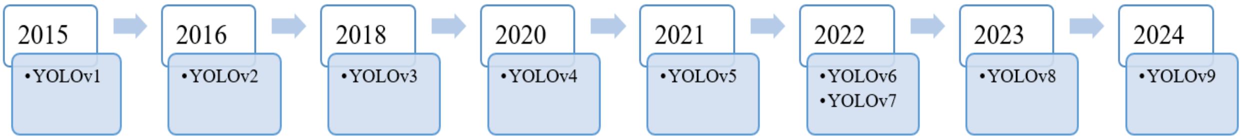


Fig. 3. Development process of YOLO

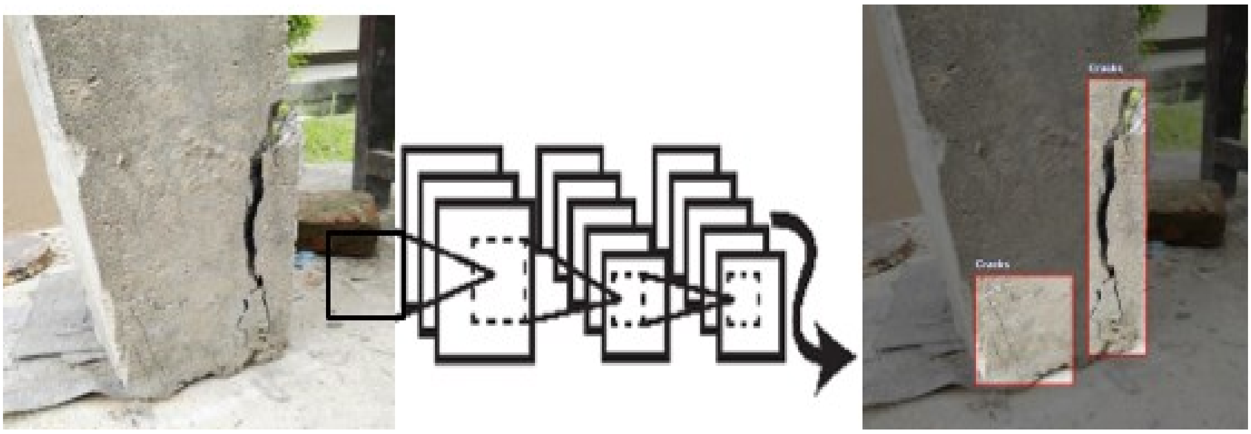


Fig. 4. The detection system of YOLO

3.2. Performance indices of model

Determining the training set, validation set, and test set is crucial in a ML project. The training set is fundamental for learning and model development, the validation set is essential for tuning and preventing overfitting during training, and the test set is critical for an unbiased final evaluation of the model's performance. A successful ML project requires all three sets to ensure the model is well-trained, properly tuned, and accurately evaluated [22].

In object detection, Average Precision (AP) summarizes the precision-recall curve into a single value representing the area under the curve, providing a single metric for evaluation. Mean Average Precision (mAP) is crucial for a balanced evaluation in tasks like object detection and multi-class classification, offering a comprehensive view of model performance. Additionally, Precision and Recall are critical metrics for evaluating model performance, with values that vary as the confidence threshold changes. Various loss components are used to train object detection models effectively by penalizing different types of prediction errors. Box loss evaluates the accuracy of the model's predictions regarding the location and size of objects within each image. Class loss measures the error in predicting the class labels of detected objects, ensuring the model accurately identifies the class of each object within the detected bounding boxes. Object loss, or confidence loss, evaluates the confidence score and error in predicting a bounding box. Each of these losses plays a crucial role in training a robust object detection model by ensuring accurate localization, classification, and confidence in object detection. The combination of these losses allows the model to learn to detect and classify objects in images effectively.

3.3. Methodology workflow

The methodology for the ML process in this study is illustrated in Fig. 5. The workflow for this paper encompasses several key stages, starting with data collection and preparation, where a comprehensive dataset of concrete crack images was compiled from various sources. This dataset was then divided into training, validation, and test sets. The training set was used to develop the model, the validation set was employed to tune the model and prevent overfitting, and the test set was utilized for unbiased evaluation of the model's performance.

Following data preparation, the images were annotated with crack labels and bounding boxes to ensure accurate detection and classification. The annotated dataset was processed through a convolutional neural network, specifically the YOLOv9 algorithm, to predict bounding boxes and class probabilities. Various loss components, including box loss, class loss, and object loss, were used to penalize prediction errors and refine the model. During training, the model's hyperparameters were tuned using the validation set to achieve optimal performance. The final model's accuracy and effectiveness were evaluated using the test set, considering metrics such as mean Average Precision (mAP), Precision, and Recall. This comprehensive workflow ensured the development of a robust and accurate model for detecting cracks in concrete structures.

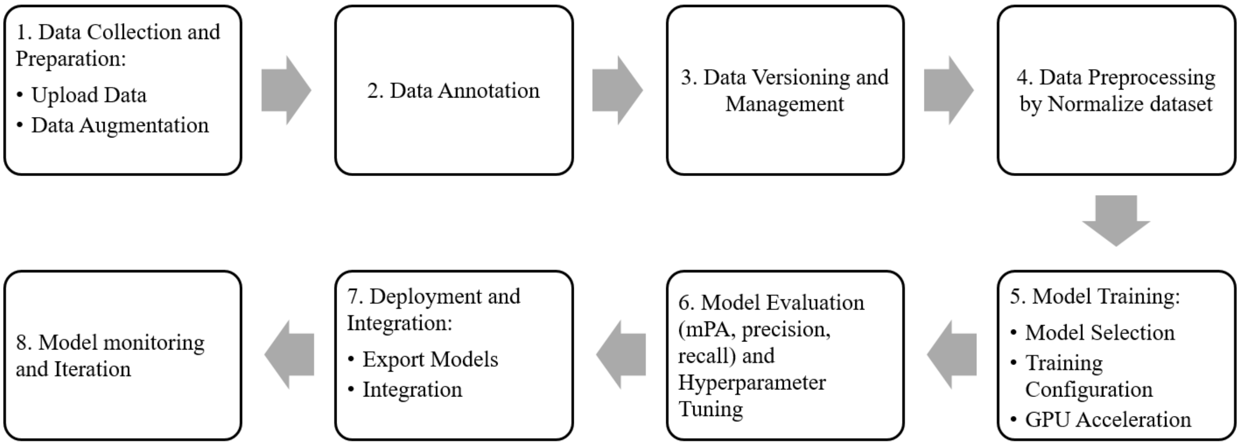


Fig. 5. Methodology workflow

4. Results and discussion

4.1. Model construction and development

Different versions were tested to verify the results. The training process of model V1 (Fig. 6), which used 343 images, involved a train/test/validation split of 60%/20%/20%. The mAP was plotted as a function of the training epoch. However, the training process was observed to be unstable, with low accuracy and lack of convergence after multiple iterations. This highlighted the need for adjustments in the training approach to achieve more reliable and accurate results. The training process of the V2 model, which utilized 16,293 images and was split into train/test/validation sets (60%/20%/20%), and the V4 model, which used 16,301 images with the same split, are shown in Figs. 7 and 9. The training process for both models stabilized after 60 iterations, with no further improvement observed in mAP beyond this point. The accuracy remained high, and the mAP converged to approximately 0.9 after 60 epochs. This indicates that the models achieved good performance and stability within a relatively short training period.

The training process for the V3 model, which used a dataset split of 70%/15%/15% and included 16,293 images resized to 640x640 pixels, is shown in Fig. 8. Similarly, the V5 model, which used 16,301 images and applied Histogram Equalization, is shown in Fig. 10. Both training processes demonstrated stability, with high accuracy achieved after only 15 epochs. This indicates that the preprocessing steps and dataset splits effectively contributed to the models' performance and rapid convergence.

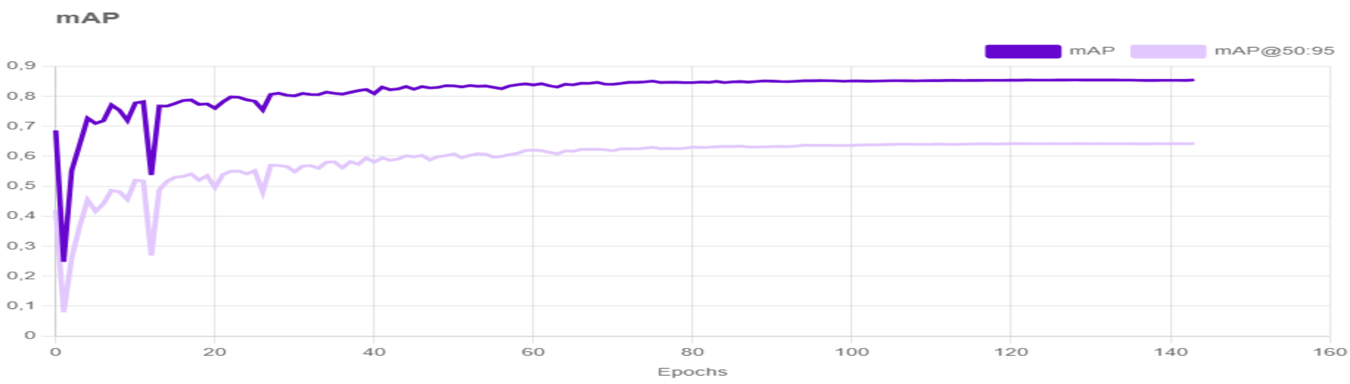
The training process of the V6 model, which utilized 16,301 images with a train/test/validation split of 80%/10%/10%, is illustrated in Fig. 11. The training process stabilized after a few iterations, with no further improvement observed, with mAP increasing from 50 to 95. High accuracy was achieved after just 10 epochs, with mAP converging to over 0.9 within this period. Using a dataset specifically labeled for cracks, the performance of the YOLOv9 model in crack detection was significantly enhanced. The mAP increased from 89% to 94.6%, and the recall improved notably from 78% to 88.4%. This demonstrates the effectiveness of the labeled dataset and the robust performance of the YOLOv9 model in detecting concrete cracks.

After removing low-resolution, mislabeled, and unlabeled images, the refined database of 16,301 images showed improved accuracy and performance. The results of different versions of the model are presented in Table 1 below.

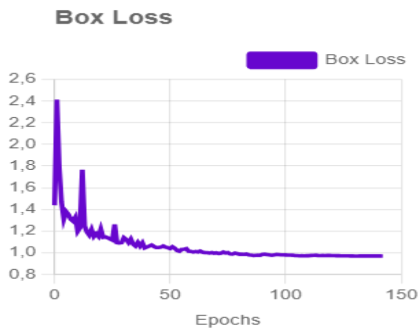
(a) mAP

(b) Box Loss (c) Class Loss (d) Object Loss

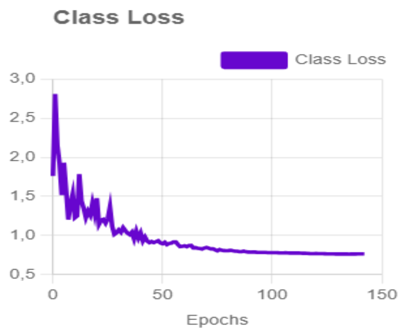
Fig. 6. Training process of V1 model



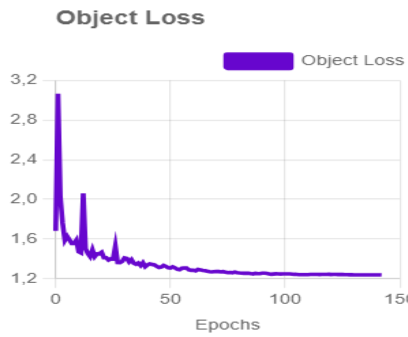
(a) mAP



(b) Box Loss

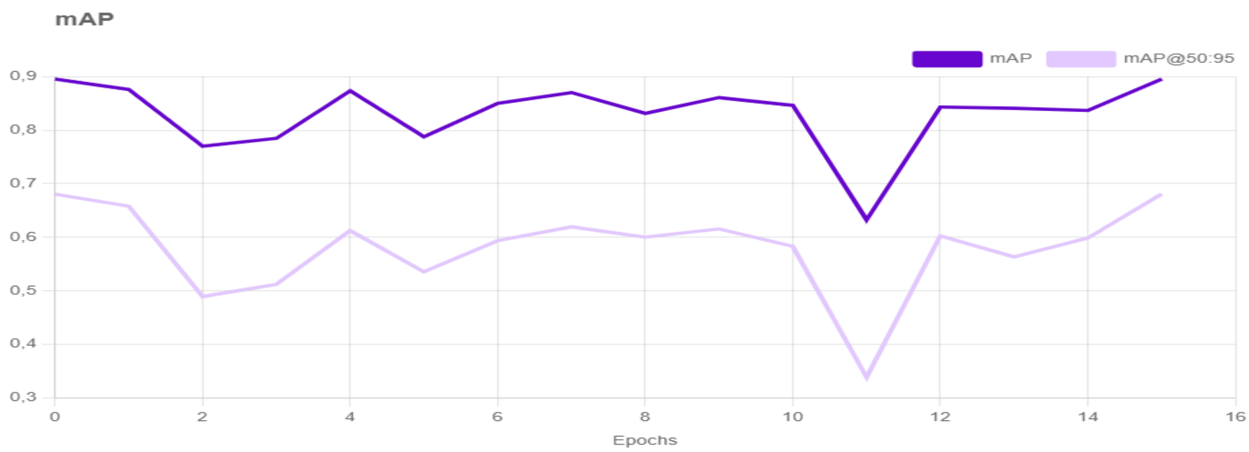


(c) Class Loss

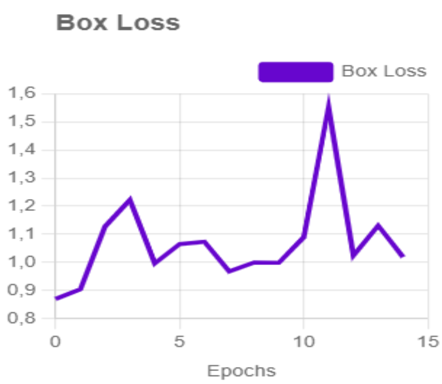


(d) Object Loss

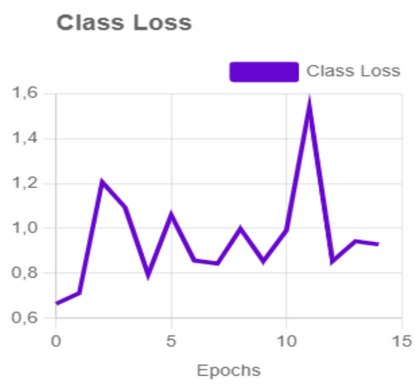
Fig. 7. Training process of V2 model



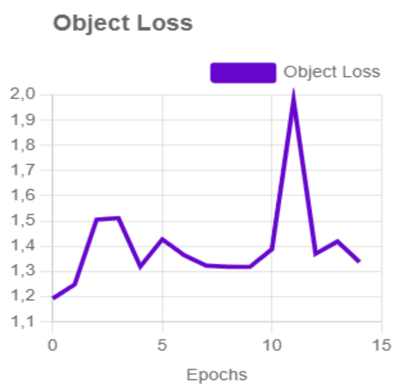
(a) mAP



(b) Box Loss

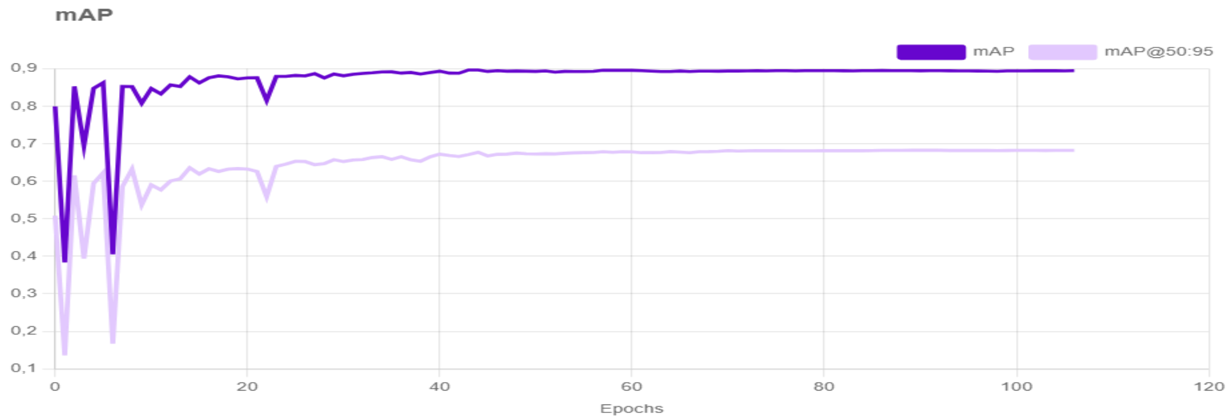


(c) Class Loss

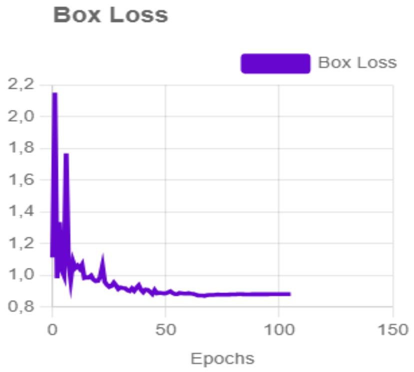


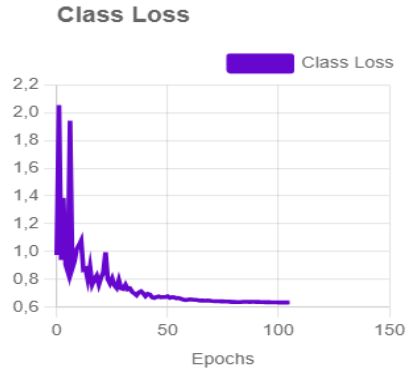
(d) Object Loss

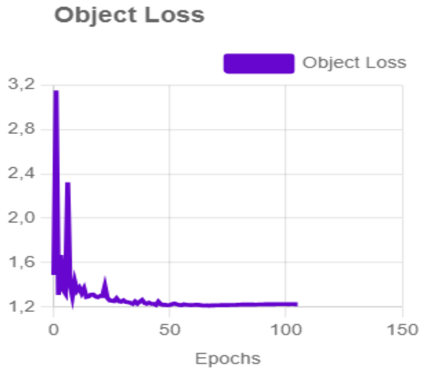
Fig. 8. Training process of V3 model



(a) mAP

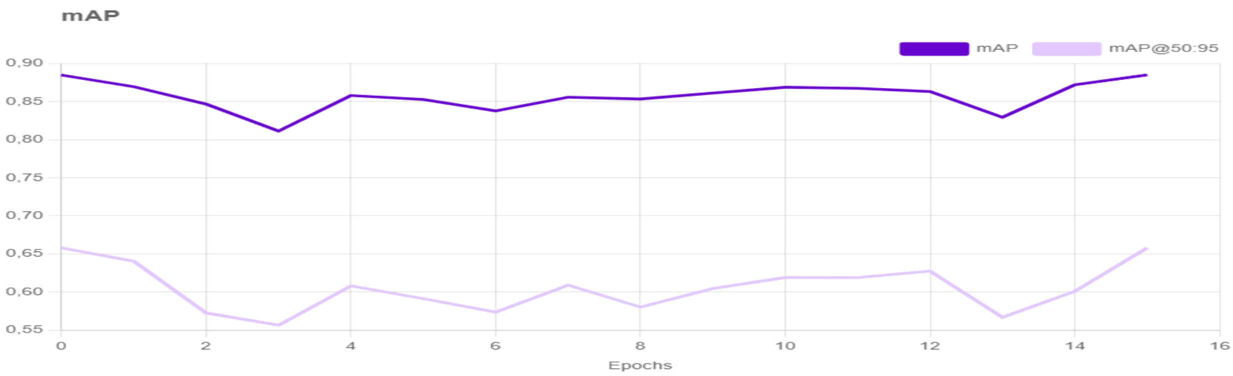




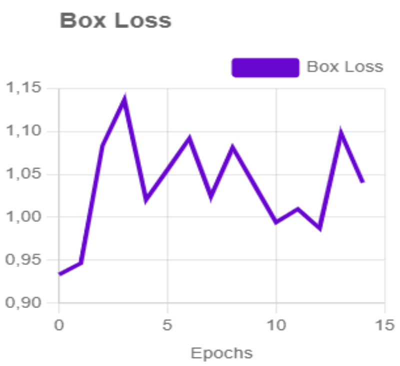


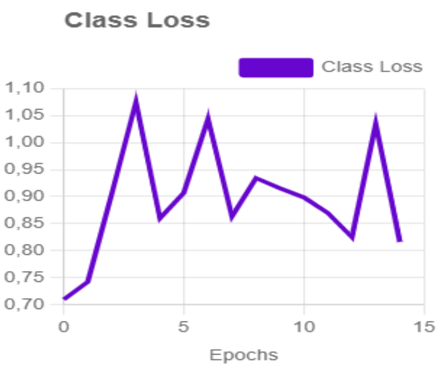
(b) Box Loss (c) Class Loss (d) Object Loss

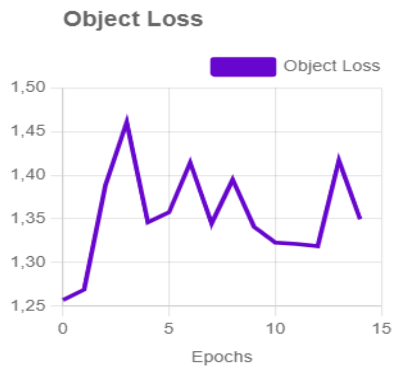
Fig. 9. Training process of V4 model



(a) mAP

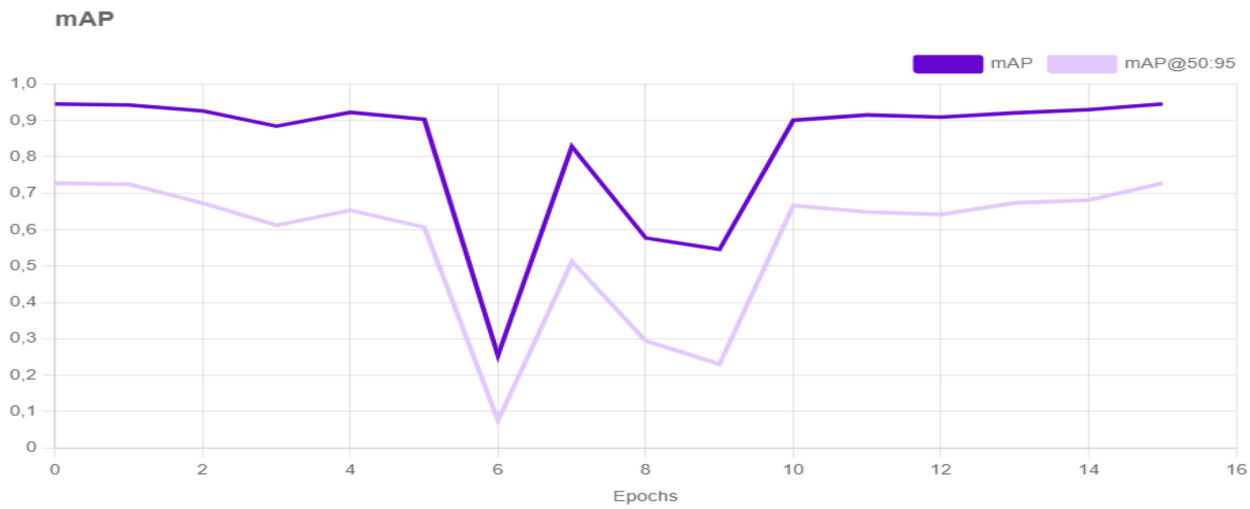




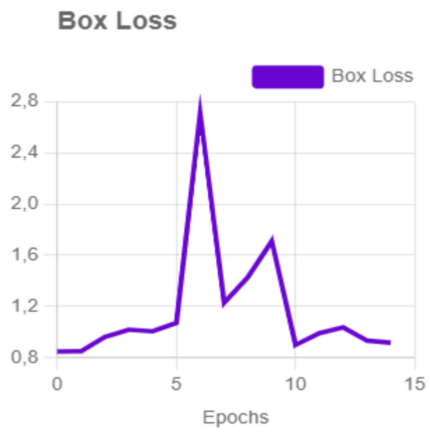


(b) Box Loss (c) Class Loss (d) Object Loss

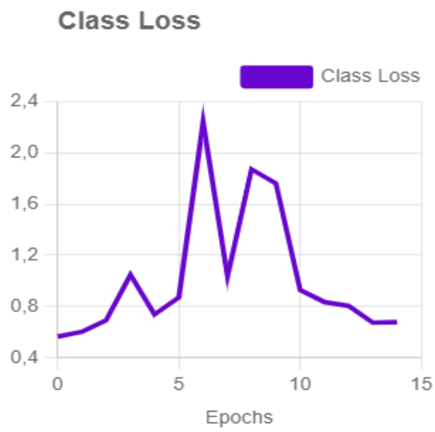
Fig. 10. Training process of V5 model



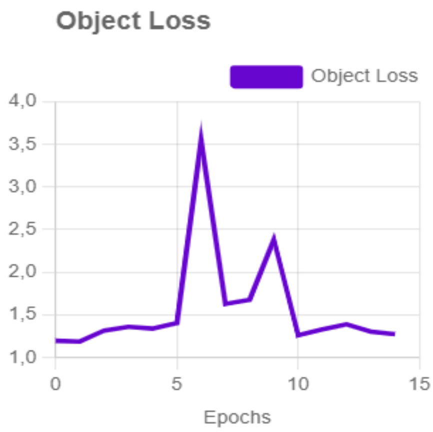
(a) mAP



(b) Box Loss



(c) Class Loss



(d) Object Loss

Fig. 11. Training process of V6 model

Table 1. Accuracy of different models

	Preprocessing	**Dataset Split**	**No. of images**	**mAP (%)**	**Precision (%)**	**Recall (%)**
V1	Orientation	60-20-20	343	21.9	63.5	19.9
V2	Orientation	70-20-10	16,293	85.4	92.6	74.4
V3	Orientation Resize: Stretch to 640x640	70-15-15	16,293	89.6	93.8	78.0
V4	Orientation	70-15-15	16,301	89.5	92.5	80.0
V5	Orientation Contrast: using Histogram Equalization	70-15-15	16,301	88.4	92.7	78.1
V6	Orientation	80-10-10	16,301	94.6	94.1	88.4

4.2. Model performance

This section presents the model performance, focusing on the best version used to predict the test set (Fig. 12). Several images are shown to illustrate the predictions and highlight the model's effectiveness. The results reveal that the model performs well in detecting various types of cracks. Comparing the performance across the training, validation, and test sets, the model demonstrates high accuracy and consistency, with minimal differences between these sets. The results indicate that the model effectively detects cracks in concrete structures, suggesting its suitability for this type of problem. ML models like YOLOv9 offer significant advantages, including high accuracy, real-time detection capabilities, and robustness against diverse background features. However, challenges remain in handling low-resolution images and ensuring comprehensive labeling.

Comparing these findings with previous studies, such as those by Choi et al. [16] and Zhang et al. [19], the model's performance aligns closely with state-of-the-art results. Specifically, the results in this work show higher mAP and recall rates, demonstrating superior performance in certain aspects. These improvements underscore the effectiveness of the refined dataset and the robust capabilities of the YOLOv9 model in concrete crack detection.

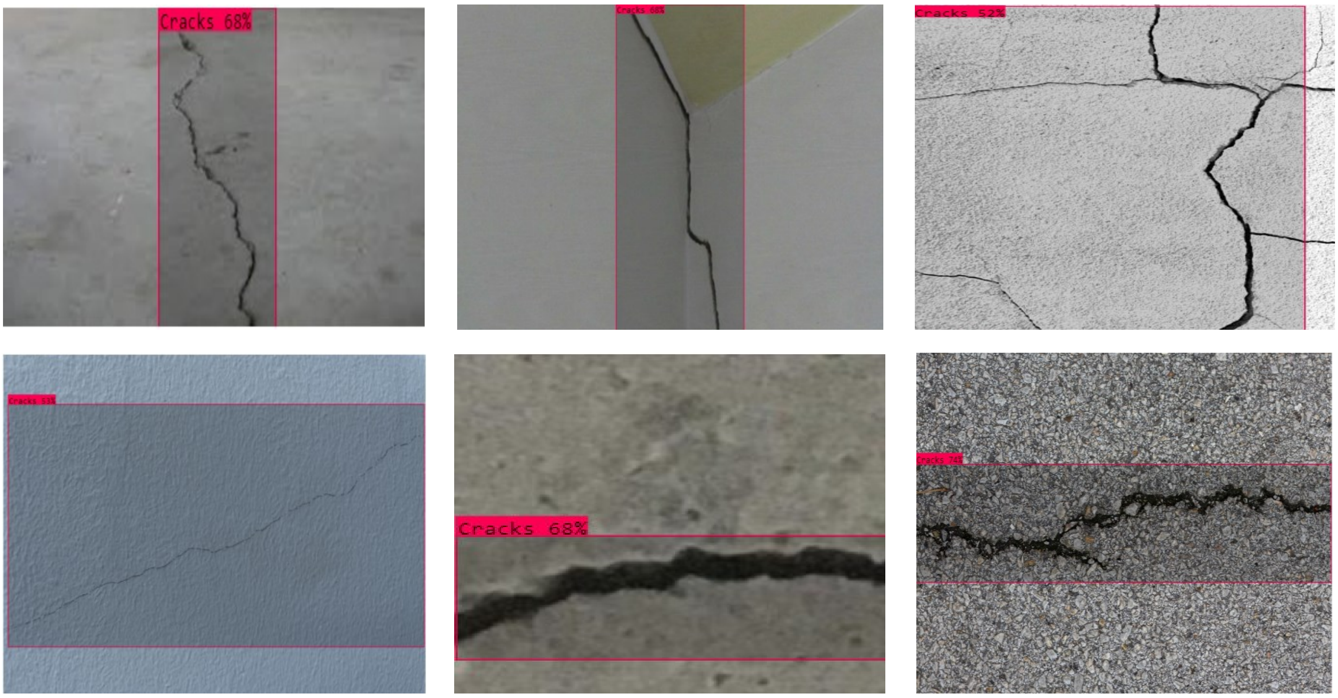


Fig. 12. Illustration of model performance on crack concrete

4.3. Model deployment

This section details the deployment of the crack detection model on concrete datasets. The model was built, debugged, and trained to achieve high accuracy and performance. A QR code and a link are provided (Fig. 13) below to enable users to test the model on real images using a computer or smartphone. The best-performing model was deployed to predict new images, with several examples presented to demonstrate its capabilities. The model consistently identifies cracks with high accuracy across various types of concrete surfaces. The comparison between the training, validation, and test sets shows that the model maintains excellent performance and robustness, indicating a successful deployment. The practical application of this model confirms its effectiveness for real-world crack detection problems. The model offers high precision, ease of use, and adaptability to different scenarios. However, potential limitations include the need for high-quality images and accurate labeling.

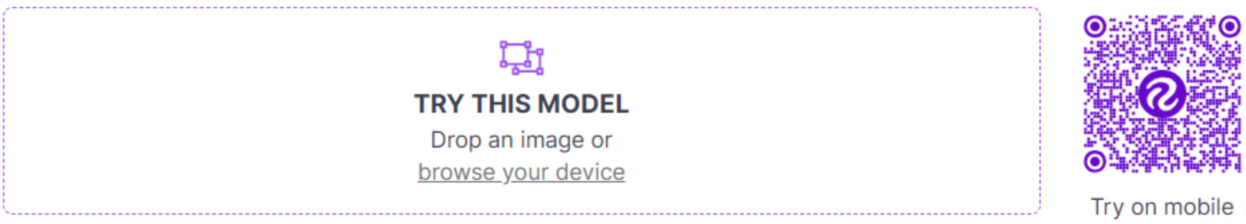


Fig. 13. Model deployment using QR code

5. Conclusions and Perspectives

This research presents a ML model developed to automatically identify various types of cracks in concrete structures. The study utilized a substantial dataset comprising 16,301 images of concrete cracks, annotated for accurate labeling during the training phase. The dataset was divided into training, validation, and test sets, adhering to a typical ML workflow. The training process utilized the YOLOv9 architecture, known for its high accuracy and speed in real-time object detection. Various hyperparameters were tuned to optimize the model's performance. The model's performance was evaluated based on mAP, precision, and recall. The model achieved an mAP of 94.6%, precision of 94.1%, and recall of 88.4%, demonstrating high reliability and accurate predictive capabilities. The findings from this study suggest significant potential for practical applications in real-time crack detection within concrete structures. Implementing this technology enables continuous and automated monitoring, facilitating timely assessment and intervention. This capability can significantly improve the efficiency of inspection, diagnosis, evaluation, and maintenance procedures for concrete structures.

This study demonstrates the potential of AI and ML in automating concrete crack detection, but acknowledges limitations due to factors like image quality, lighting, and the diversity of crack types in the training data. Future research could address these limitations by incorporating more diverse datasets, exploring advanced image preprocessing techniques, and investigating the model's generalizability to different concrete structures. Integrating this model with other non-destructive testing methods could provide a more comprehensive assessment of concrete structure health. Additionally, future work could focus on incorporating crack classification capabilities into the model, utilizing additional labeling and training data that includes specific crack attributes such as shape, size, and depth, aligning with allowable limits defined by relevant standards. While the current model focuses on general crack detection, this initial step is practically valuable as a screening tool.

References

[1]. J.S. Damtoft, J. Lukasik, D. Herfort, D. Sorrentino, E.M. Gartner. (2008). Sustainable development and climate change initiatives. *Cement and Concrete Research*, 38(2), 115-127.

[2]. American Concrete Institute. (2024). Cracking in Concrete Topic, https://www.concrete.org/topicsinconcrete/topicdetail/cracking%20in%20concrete?search=cracking%20in%20concrete (accessed 2nd June 2024).

[3]. H. Kim, E. Ahn, S. Cho, M. Shin, S.-H. Sim. (2017). Comparative analysis of image binarization methods for crack identification in concrete structures. *Cement and Concrete Research*, 99, 53-61.

[4]. T. Yamaguchi, S. Nakamura, R. Saegusa, S. Hashimoto. (2008). Image-Based Crack Detection for Real Concrete Surfaces. *IEEJ Transactions on Electrical and Electronic Engineering*, 3(1), 128-135.

[5]. Y. Fujita, Y. Mitani, Y. Hamamoto. (2006). A method for crack detection on a concrete structure. *18th International Conference on Pattern Recognition (ICPR'06)*, pp 901-904. IEEE (2006).

[6]. A. Arbaoui, A. Ouahabi, S. Jacques, M. Hamiane. (2021). Concrete Cracks Detection and Monitoring Using Deep Learning-Based Multiresolution Analysis. *Electronics*, 10(15), 1772.

[7]. M. Flah, A.R. Suleiman, M.L. Nehdi. (2020). Classification and quantification of cracks in concrete structures using deep learning image-based techniques. *Cement and Concrete Composites*, 114, 103781.

[8]. È. Quintana Aguasca. (2021). Tracking by detection using YOLO and Kalman filtering,

https://upcommons.upc.edu/handle/2117/357892.

[9]. S.E. Park, S.-H. Eem, H. Jeon. (2020). Concrete crack detection and quantification using deep learning and structured light. *Construction and Building Materials*, 252, 119096.

[10]. L.D. Nghiep, P.V Binh, D.N. Toan, P.T. Ha, T.V. Huy. (2019). Custumized YOLO architecture for Logo recognition. TNU Journal of Science and Technology. ISSN: 1859-2171. 200(07), 199-205.

[11]. Y. Ding, C. Liu, H. Wang, Z. Chang. (2022). Research on Mask Wearing Detection Based on Faster RCNN. *2022 5th International Conference on Advanced Electronic Materials, Computers and Software Engineering (AEMCSE)*, pp 629-633. IEEE, Wuhan, China.

[12]. X. Han, J. Chang, K. Wang. (2021). Real-time object detection based on YOLO-v2 for tiny vehicle object. *Procedia Computer Science*, 183, 61-72.

[13]. N.X. Thang (2023). Applying adaptive deep learning model for detecting traffic vehicles. Journal of Science and Transport Technology (JSTT), 3(2), 38-46.

[14]. D.T. Son, N.T.K. Tram. (2020). Application of YOLO model in firearm detection. The 9th International Conference on Information Technology and Its Applications, pp.152-157. CITA 2020.

[15]. W. Zhang, Y. Zhou, Y. Wang, R. Wang, H. Yang. (2024). Automatic Crack Detection and Segmentation of Masonry Structure Based on Yolov9-Seg and Edge Detection. SSRN, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4812249 (accessed 2nd June 2024).

[16]. W. Choi, Y.-J.Cha. (2020). SDDNet: Real-Time Crack Segmentation. *IEEE Trans. Ind. Electron*, 67(9), 8016-8025.

[17]. J. Huyan, W. Li, S. Tighe, Z. Xu, J. Zhai. (2020). CrackU-net: A novel deep convolutional neural network for pixelwise pavement crack detection. *Structural Control and Health Monitoring*, 27, e2551.

[18]. H. Kim, H. Jeon, S.-C. Baek, W.-H. Hong and H.-J. Jung. (2018). Application of Crack Identification Techniques for an Aging Concrete Bridge Inspection Using an Unmanned Aerial Vehicle. *Sensors*, 18(6), 1881.

[19]. G.H. Beckman, D. Polyzois, Y.-J. Cha. (2019). Deep learning-based automatic volumetric damage quantification using depth camera. *Automation in Construction*, 99, 114-124.

[20]. N.T.T. Hien. (2024). Roboflow_Hien_Crack_Concrete. https://app.roboflow.com/nguyen-thi-thuy-hien-3ea5t/bac_hien_crack_concrete_2024/deploy

[21]. J. Redmon, S. Divvala, R. Girshick, A. Farhadi. (2016). You Only Look Once: Unified, Real-Time Object Detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 779-788. IEEE, Las Vegas, NV, USA.

[22]. H. Li, L. Deng, C. Yang, J. Liu, Z. Gu. (2021). Enhanced YOLO v3 Tiny Network for Real-Time Ship Detection From Visual Image. *IEEE Access*, 9, 16692-16706.