PORT SAID ENGINEERING RESEARCH JOURNAL
Faculty of Engineering - Port Said University
Volume 28 No. 2 pp: 18 : 30

YOLOv7 Deep Learning Model for Pavement Crack Detection Using Close Range Photogrammetry Dataset

F.Ashraf[1,], M. Rabah[2], Essam. Ghanem[3]*

[1] *Department of Civil Engineering, Higher Technological Institute, Tenth of Ramadan City, Egypt, email: fekry.ashraf@hti.edu.eg*
[2] *Department of Civil Engineering, Benha Faculty of Engineering, Benha University, Egypt, email: m.rabah@bhit.bu.edu.eg*
[3] *Department of Civil Engineering, Benha Faculty of Engineering, Benha University, Egypt, email: essam.ghanem@bhit.bu.edu.eg*
**Corresponding author,* *DOI: 10.21608/PSERJ.2024.246447.1276*

Received 5 -11 2023,
Revised 29-12-2023,
Accepted 10-1-2024

ABSTRACT

Developing an effective system for detecting and classifying pavement cracks is crucial for ensuring traffic safety. However, the procedure of manual inspection for identifying these cracks can be hazardous and time-consuming. Thus, it's essential to implement an automated approach to make the detection process more efficient. Overcoming challenges like varying intensity levels, inconsistent data availability, and ineffective traditional methods make this task complicated. This research's aim is to contribute to the development of an efficient system for detecting pavement cracks. Pavement crack detection using close range photogrammetry is a process for identifying, characterizing and evaluating pavement surface cracks that is revolutionizing the speed, accuracy and cost of assessing the structural integrity of pavements. These images are used by analysis software to generate detailed digital maps of the pavement surface. These digital maps can then be used to identify and measure pavement cracking. The use of close-range photogrammetry for pavement crack detection offers numerous advantages over traditional pavement inspection methods, including improved accuracy and flexibility in the analysis of pavement cracks and the ability to analyze large areas of pavement quickly. The quality of the images captured depends on the type of camera used, but most cameras offer high-resolution imaging at close range. The data was intentionally gathered via iphone14promax camera in Tenth of Ramdan City. The customized YOLOv7 model, which is a state-of-the-art deep learning algorithm, was used in this study. The difference between ground truth and boundary box is 0.016, the class probability loss is 0.021 and the objectless loss is 0.009. The precision of the outcome reports is 0.854 and recall from the custom dataset is 0.755. The results of the suggested system were satisfactory compared to the results of reference studies.

Keywords: Deep Learning YOLOv7- Pavement Crack Detection-close range photogrammetry.

1 INTRODUCTION

Damage to asphalt pavement can be caused by both environmental hazards, like prolonged sun exposure, erosion caused by rain, and the effects of natural weathering, as well as human factors, such as rolling of vehicles, the quality of pavement materials used, the level of construction quality, and subsequent maintenance. These factors have varying levels of influence on the performance of pavement [1]. Failure to detect and repair damaged roads in a timely manner can lead to a reduction in the service life and quality of the highway, and even increase the risk of traffic accidents [2].Currently, pavement detection is mostly done manually, which has several drawbacks, including being time-consuming, requiring a significant amount of manpower, obstructing the highway, posing safety risks to inspectors, and potentially affecting detection results due to human factors. As highways continue to undergo rapid development, meeting the detection demands of large-scale projects has become increasingly challenging. As a result, current approaches have fallen significantly short in fulfilling the requirements for the continued growth and expansion of highways [3].

To enhance the service quality of highways and achieve automated detection of damaged pavement, researchers have suggested exploring the use of visual technology for paved road detection. Previously, Technology for processing digital images was employed for identifying cracks [4]. There has been significant focus on image-based crack detection algorithms in recent decades. Early research methods primarily involved refining or integrating conventional methods for digital image processing like edge detection, geometric morphology, and thresholding [5]. These techniques were based on photometric and geometric principles and were used to analyze crack images [6]. Crack pixels were identified as the prominent photometric feature because they appeared darker in an image. Crack and background segmentation was achieved by determining a global or local threshold value [7]. However, these techniques were sensitive to noise due to their implementation at the pixel level. To address this, some solutions used geometric information, such as limiting false detection by analyzing crack continuity [8]. A local operator for binary patterns was also used to recognise whether individual pixels belonged to cracks by considering a local orientation [9]. With the implementation of multiscale analysis, wavelet transform has been used to distinguish between areas that contain cracks and those that do not, and although these methods are successful in detecting cracks, they may not always be precise in their identification of all cracks present in an image. As a response to this issue, researchers have been exploring computer vision and Artificial Intelligence (AI) technologies to develop automatic crack detection methodologies [10]. AI and machine vision are now employed to solve various problems across multiple domains, from banking and healthcare to engineering and other technical challenges [11]. Owing to the extensive use of machine learning techniques, especially deep learning, in research and industry, deep learning models can be utilized to automatically identify and categorize pavement cracks. Several approaches that rely on pattern recognition and feature extraction have been proposed for crack recognition since the arrival of machine learning. While these models have shown outstanding performance, their effectiveness is highly dependent on the extracted features and may not be practical for all types of pavements given the complexity of pavement conditions. Despite this, Deep Learning has the potential to significantly enhance road maintenance performance [12].

2 RELATED WORKS

2.1. Automatic Detection Based on Image Processing

Image processing techniques have been used to automatically detect pavement cracks. Early methods relied on threshold segmentation algorithms to extract crack areas, which assume that crack pixels are generally darker than their surroundings. Li and Liu [13] suggested a new technique based on thresholding based on adjacent difference histograms, which achieved better results than traditional methods. Other researchers combined threshold segmentation with the connected domain algorithm or the neighborhood difference histogram to improve detection accuracy. However, these methods are sensitive to noise and can produce false positives, especially when non-crack features are present. To achieve a more precise detection, Gavilán et al [14] suggested estimating the mean gray value of pixels in the inside and outside contours of linear objects present in the image. Another approach, used by Li and Mao [15], involved dividing the image into multiple sub-regions, computing the neighborhood difference histogram for each region, and then fusing the resulting crack information. However, these methods are still limited by their inability to describe global information, their sensitivity to noise, and their dependence on the selection of a threshold for detection. In practical applications, more advanced algorithm models are needed to handle the complexity of road backgrounds and the presence of noise.

2.2. Automatic Crack Detection Based on Machine Learning Technique

The use of machine learning depending on functional engineering has proved successful in numerous fields, including automatic detection of pavement cracks. Researchers have identified significant texture characteristics of the crack pavement and used image classification technologies to detect them automatically. Hu et al [16] proposed approach based on analysis of texture and shape explanation, which uses six attributes of texture and two translation invariant shape identifiers to classify images into cracks and non-cracks using support vector machine (SVM). Cord and Chambon [17] suggested a supervised learning approach based on linear and nonlinear filters to describe texture features of the crack pavement at different scales. They used the AdaBoost classifier to learn and classify information from the filters and obtain the pavement damage area. Shi et al [18] proposed a method called Crack Forest, which combined a framework based on overall channel features with random forest classification. To represent the cracks and remove noise that was incorrectly labeled as cracks, they utilized two feature histograms. Although these methods have shown some improvements in crack detection, they still face challenges in fine extraction of cracks.

Despite some success in overcoming road noise interference, current machine learning-based methods for crack extraction struggle to meet the challenges of complex backgrounds. Using a data set comprising of projective integral and fracture properties, Hoang and Nguyen [19] tested the efficacy of a machine learning algorithm trained on the support vector machines (SVM), artificial neural networks (ANN), and random forest (RF) algorithms. However, the accuracy of these algorithms is highly dependent on manually extracted image features such as color and texture. Therefore, developing separate feature models to suit different lighting scenarios and conditions is necessary. The diverse road environment, with its numerous debris and noises, makes it challenging to extract practical attributes with a single feature model, resulting in poor performance of the detection model. Therefore, the machine learning-based detection model may only be effective within a small range of conditions and is not universally applicable.

2.3. Automatic Crack Detection Based on Deep Learning Technique

Automatic crack detection based on deep learning has improved significantly over the past few years. Object detection H. Maeda, et al [20] and image segmentation X. Wang and Z. Hu [21] have been used to extract cracks, but they cannot complete pixel-level detection or accurately determine damage severity. To address this, Zhang et al [22] introduced a system for pavement detection called CrackNet that uses convolutional neural networks (CNN) and pixel-level extraction to automatically detect 3D asphalt cracks. Unlike traditional CNNs, Crack Net doesn't use pooling layers to decrease the output of the preceding layer and uses maintain a consistent image width and height technique across all layers of the network. Fei et al [23] improved upon Crack Net with a deeper and more efficient deep network, Crack Net-V, while maintaining the same learning at the pixel level. Meanwhile, Zou et al [24] developed a trainable deep CNN, Deep Crack that learns advanced crack representations and is end-to-end trainable. These studies demonstrate the effectiveness of technologies for deep learning in automatic detecting cracks in pavement at the pixel level and for identifying different types of damage accurately. To enhance crack detection, the deep crack network was developed using multiscale deep convolution features learned from various layers to create a linear structure. This network, built on the encoder-decoder architecture of SEG net, integrated convolution features from both encoder and decoder networks at the same scale to automatically identify cracks at the pixel level with a detailed description of expansive feature maps and thorough representation of compact feature maps. Compared to machine learning models that rely on feature engineering, the deep learning-based detection model exhibited much better detection performance. Combining the deep crack network with the most effective current model for crack pavement object detection promises a significant improvement in the pavement detection process.

3 RESEARCH METHODOLOGY

The current study presents a pavement crack detection model wherein data from a camera are used. To prepare the collected data for analysis, several preprocessing steps were involved, such as extracting image frames, labeling, augmentation, and data resizing. Data is then divided into a train, validation, and test samples, which are trained using the YoloV7 algorithm. Once the model training is completed, the performance is evaluated through testing. The proposed research approach is illustrated in Figure 1.

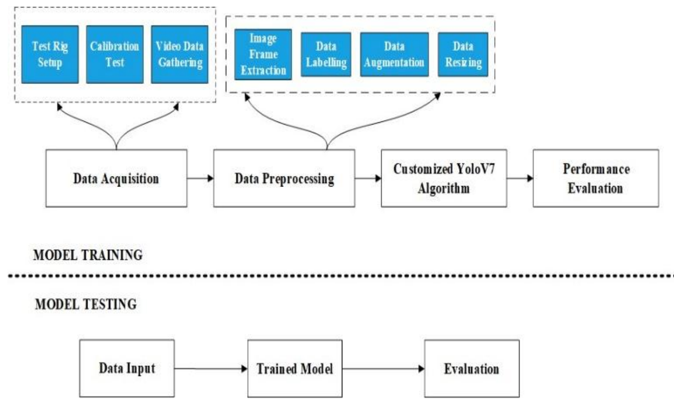

Figure 1: Suggested Research Methodology

4 COLLECTION DATA

The data was intentionally gathered via iphone14promax camera in Tenth of Ramdan City. The primary camera of the iPhone 14 Pro Max has seen the biggest update. It now uses a 48MP 1/1.28" sensor with a Quad-Bayer color filter, a first for the iPhone. The camera has a 1.22µm pixel size before binning - 2.44µm with binning. It's coupled with a 24mm f/1.78 lens. There is also second-gen sensor-shift stabilization, as well as full-focus pixels. The number of images is 250 images (200 for train and 50 for test) from height 1.5m to 2m with angle from 45 degree to 90 degree. Figure 2 shows the location, the number of images collected, the specification of apple iPhone 14promax camera and some raw images.

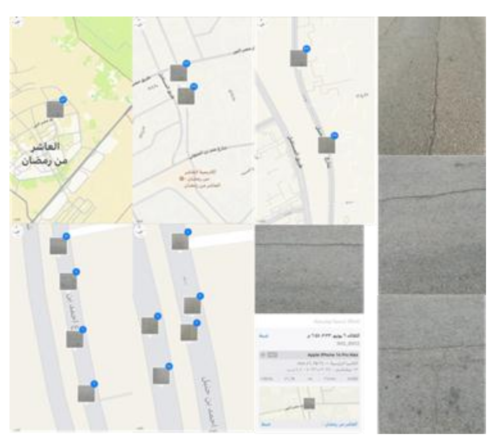

Figure 2: Data collection location, number, specification, and some raw images.

5 DATA PREPROCESSING

Prior to instructing the deep learning algorithm, the research data undergoes various preprocessing techniques. These include extracting image frames from the recorded video clips, data labelling using the Roboflow data annotation tool, and data augmentation and resizing to enhance the number and variety of training data. The quantity and variety of training data are crucial for the accuracy of supervised deep learning models. However, obtaining enough data can become a challenge, necessitating the use of data augmentation techniques that apply very little data changes or use machine learning frameworks to generate more data elements. Overall, an proposed preprocessing steps are crucial in preparing high-quality training data for deep learning models. The study utilized various data augmentation techniques, including blurring and brightness adjustments. Blurring refers to a visual effect that makes edges of text or images appear fuzzy or unfocused. Brightness, on the other hand, refers to the overall luminance or darkness of an image. After implementing data augmentation, the image samples underwent resizing, which is a crucial pre-processing step for computer vision. This involves adjusting the visible dimensions of the image, often to smaller sizes to improve deep learning algorithm efficiency. In this study, the images from the datasets were resized to 640 x 480 pixels.

6 YOLOV7 MODEL

Deep Learning is a subfield of machine learning that revolves around algorithms inspired by the design and operation of the brain's artificial neural networks. The central concept of Deep Learning is to train computer systems to learn through examples, emulating natural human learning. Just now, there has been a surge of interest in Deep Learning, as it can produce outcomes previously thought to be unattainable. Deep Learning models use multi-layered neural network architectures that learn right from images, text, or voice to tackle detection and classification duties. These models can achieve precision levels that occasionally surpass human performance, thanks to the massive amounts of labeled data available for training.

YOLOv7 is the deep learning algorithm utilized in this study. belonging to the family of real-time object identification techniques that go by the name of "You Only Look Once", or YOLO for short. YOLO models use a single-stage approach for object detection, involving the head, neck, and spine/backbone architecture. In this approach, the picture frames are distinguished by the spine, which are combined and fine-tuned in the network's neck before being transmitted to the head. At this stage, YOLO expects the positions and groups of objects that require bounding boxes. Finally, YOLO uses non-maximum suppression (NMS) during post-processing to arrive at its conclusive output. Figure 3 depicts the general architecture underlying YOLO.

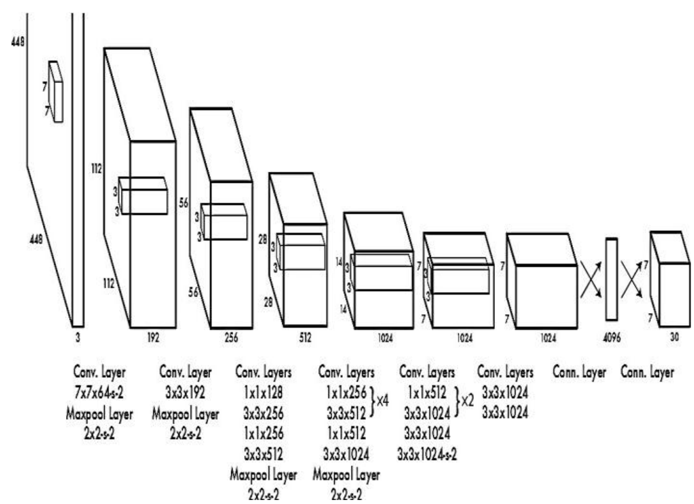

Figure 3: YOLOv7 Architecture [25]

The preprocessed image data of size 640x480 pixels was used as input for the YOLOv7 model in this study. The customized hyperparameters for the algorithm, including batch size 16, number of epochs 22, initial and final learning rates 0.010 and 0.10, weight decay 0.00050, box loss gain 0.050, cross-entropy loss 0.30, and momentum 0.937. YOLOv7 was chosen as the model for this research due to its advanced features, including Extended Efficient Layer Aggregation, Model Scaling Techniques, Re-parameterization Planning, and Auxiliary Head Coarse-to-Fine. These features were deemed crucial to achieving accurate results in crack detection.

It is crucial for the network's convolutional layers to be highly efficient in the backbone, enabling fast inference. YOLOv7's developers have improved upon previous work in this domain by considering both the distance a gradient must travel through the layers for back-propagation and the memory required to store these layers. Smaller gradients lead to faster network learning rates, after all. Finally, the YOLOv7 model utilizes E-ELAN, an improved variant of the ELAN computational block, which serves as the preferred layer aggregation method. Figure 4 shows this mechanism in more detail.

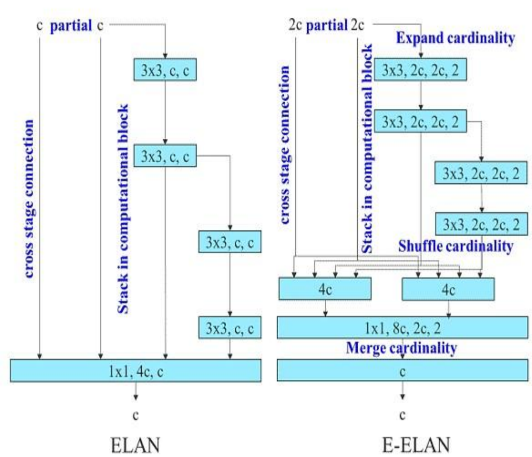

Figure 4: Evolution of layer aggregation strategies in YOLOv7 [26]

Object detection algorithms often consider metrics like network depth, breadth, and resolution that were employed in the instruction. In the case of YOLOv7, the developers take a unique approach by scaling the network's depth and breadth simultaneously through the concatenation of layers as shown in Figure 5. This method is backed by research demonstrating that it maintains optimal model creation when scaling up or down.

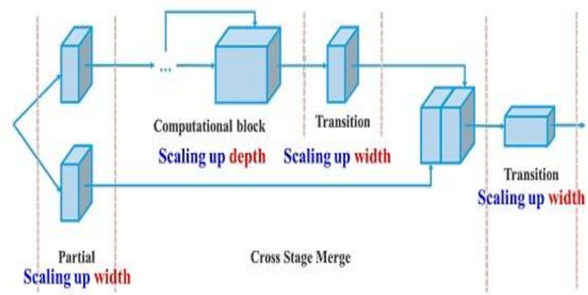

Figure 5: YOLOv7 Scaling [26]

Reparameterization techniques often involve averaging the weights of a set of models to create a more robust model capable of handling general patterns. Recent research has focused on reparameterization at the module level, where individual nodes within the network employ distinct strategies. To identify which network modules, require reparameterization, YOLOv7 examines the pathways of gradient flow. Although the YOLO network's forecasts originate from the "head" module, located at a considerable distance down the node chain, adding an auxiliary head nearer to the network center can be beneficial. While in training, supervision is applied to both the detection and prediction heads. Given the shorter network distance between the auxiliary and prediction heads, YOLOv7 developers experimented with various levels of oversight for this head, before ultimately deciding on coarse-to-fine oversight with increasing levels of granularity, passed back from the leading head to enhance training efficiency, as illustrated in Figure 6.

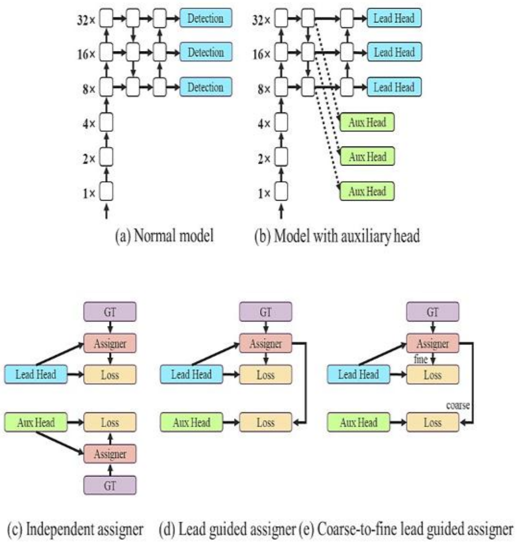

Figure 6: Auxiliary head supervision in the YOLOv7 from coarse to fine.[26]

7 RESULTS AND DISCUSSION

7.1. Evaluation Matrices

Performance evaluation of the developed model involves utilizing various metrics within the machine and deep learning domains, including accuracy, precision, recall, the confusion matrix, F1-score, and others. For classification models, the confusion matrix is a vital component of statistical analysis, providing a two-dimensional table of both approximated and actual values, as illustrated in Figure 7.

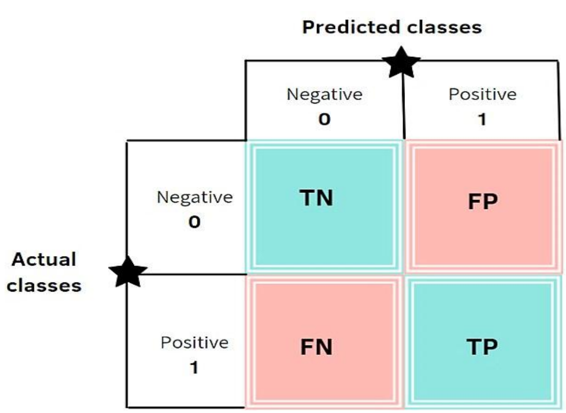

Figure 7: Confusion Matrix

The equation for accuracy, precision, recall, and F1 score is as follows:

$$\text{Accuracy} = \frac{TP+TN}{TS} \quad \textbf{(1)}$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad \textbf{(2)}$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad \textbf{(3)}$$

$$\text{F1- Score} = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad \textbf{(4)}$$

7.2. Experimental Results

We trained our YOLOv7 model on a custom dataset using a batch size of 16 and 22 epochs. Table 1 provides the box loss, segmentation loss, and objectness loss for each epoch, while Table 2 and 3 show the precision, recall, and mean average precision scores for the model with IoU thresholds of 0.5 and IoU thresholds ranging from 0.5 to 0.95 for boxes and masks, respectively. Results are depicted in Figures 8, 9, and 10. From figure 8 the difference between ground truth and boundary box is 0.016, the class probability loss is 0.021 and the objectless loss is 0.009. The highest box precision of 0.899 was achieved on epoch 14 and the highest mask precision of 0.777 on epoch 20. Additionally, the highest box recall of 0.779 was attained on epoch 17, and the maximal mask recall of 0.691 was seen on epoch 17. These outcomes are particularly noteworthy as the model detected both longitudinal and transverse cracks. Overall, the results were highly encouraging, demonstrating the effectiveness of a single model for crack detection.

After testing the developed model on a custom dataset, its accuracy was determined to be 85%. This level of accuracy is quite promising, especially with respect to other existing models. Notably, Figures 11, 12, and 13, show the testing predictions made by the developed system and the accuracy of crack prediction on all figures.

Table 1. The box loss, segmentation loss and objectless loss for custom data

Epoch	Box loss	Segmentation loss	objectless loss
1	0.071	0.042	0.025
2	0.049	0.025	0.018
3	0.047	0.024	0.014
4	0.043	0.024	0.014
5	0.039	0.024	0.013
6	0.035	0.024	0.013
7	0.033	0.023	0.013
8	0.031	0.023	0.012
9	0.029	0.023	0.012
10	0.028	0.023	0.012
11	0.026	0.023	0.011
12	0.026	0.022	0.012
13	0.024	0.022	0.011
14	0.023	0.022	0.011
15	0.022	0.022	0.010
16	0.021	0.021	0.010
17	0.020	0.021	0.010
18	0.019	0.021	0.010
19	0.018	0.021	0.010
20	0.017	0.021	0.009
21	0.017	0.021	0.009
22	0.016	0.021	0.009

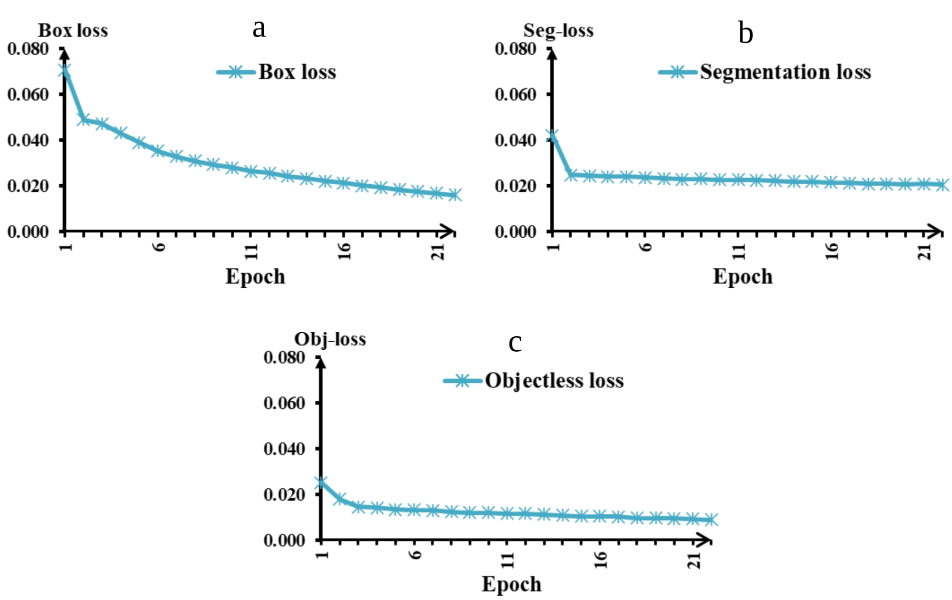

Figure 8: Performance metrics; a) Box loss, b) Segmentation loss, and c) Objectless loss

Table 2. The Box precision, Box recall and mean average precision.

Epoch	Box precision	Box recall	MAP@0.5	MAP@0.5:0.95
1	0.716	0.55	0.595	0.247
2	0.734	0.609	0.615	0.28
3	0.486	0.438	0.4	0.166
4	0.63	0.659	0.565	0.238
5	0.686	0.685	0.625	0.308
6	0.752	0.696	0.688	0.366
7	0.747	0.695	0.698	0.399
8	0.796	0.739	0.764	0.445
9	0.795	0.716	0.718	0.487
10	0.716	0.775	0.74	0.495
11	0.77	0.727	0.749	0.483
12	0.887	0.731	0.797	0.555
13	0.847	0.71	0.765	0.539
14	0.899	0.707	0.785	0.556
15	0.861	0.727	0.8	0.566
16	0.853	0.767	0.808	0.569
17	0.823	0.779	0.807	0.579
18	0.823	0.727	0.808	0.604
19	0.856	0.755	0.831	0.622
20	0.887	0.751	0.821	0.624
21	0.816	0.755	0.8	0.615
22	0.884	0.738	0.807	0.628

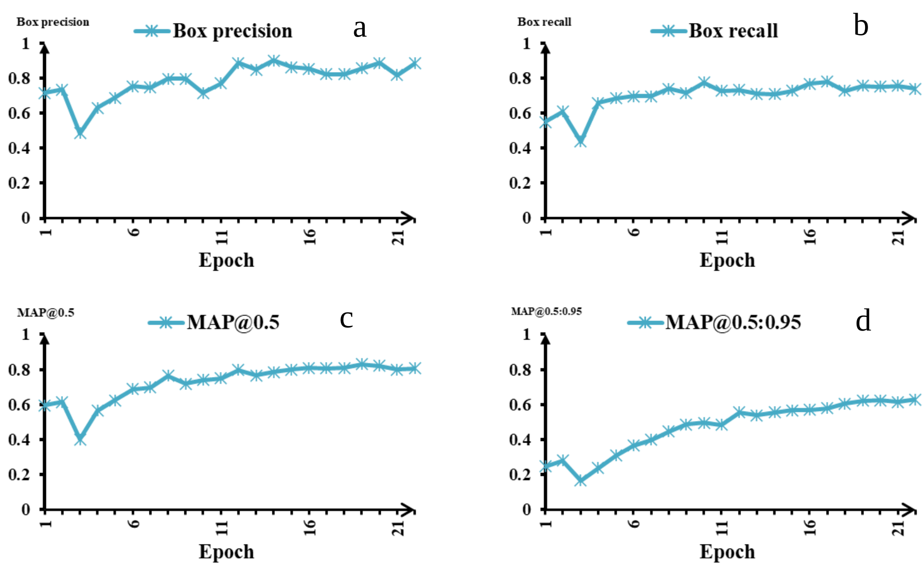

Figure 9: Performance metrics; a) Box precision, b) Box recall, c) MAP@0.5 and d) MAP@0.5:0.95

Table 3. The Mask precision, Mask recall and mean average precision.

Epoch	Mask precision	Mask recall	MAP@0.5	MAP@0.5:0.95
1	0.448	0.373	0.306	0.0767
2	0.613	0.526	0.432	0.111
3	0.344	0.355	0.245	0.0642
4	0.45	0.522	0.306	0.0893
5	0.585	0.578	0.464	0.125
6	0.612	0.542	0.447	0.13
7	0.568	0.582	0.475	0.135
8	0.63	0.586	0.477	0.136
9	0.68	0.613	0.53	0.162
10	0.607	0.655	0.552	0.177
11	0.696	0.614	0.586	0.184
12	0.776	0.653	0.628	0.197
13	0.763	0.643	0.632	0.186
14	0.762	0.618	0.593	0.192
15	0.751	0.627	0.6	0.182
16	0.71	0.64	0.593	0.199
17	0.73	0.691	0.654	0.221
18	0.759	0.655	0.665	0.218
19	0.741	0.69	0.666	0.229
20	0.777	0.659	0.66	0.219
21	0.724	0.671	0.652	0.205
22	0.745	0.621	0.604	0.204

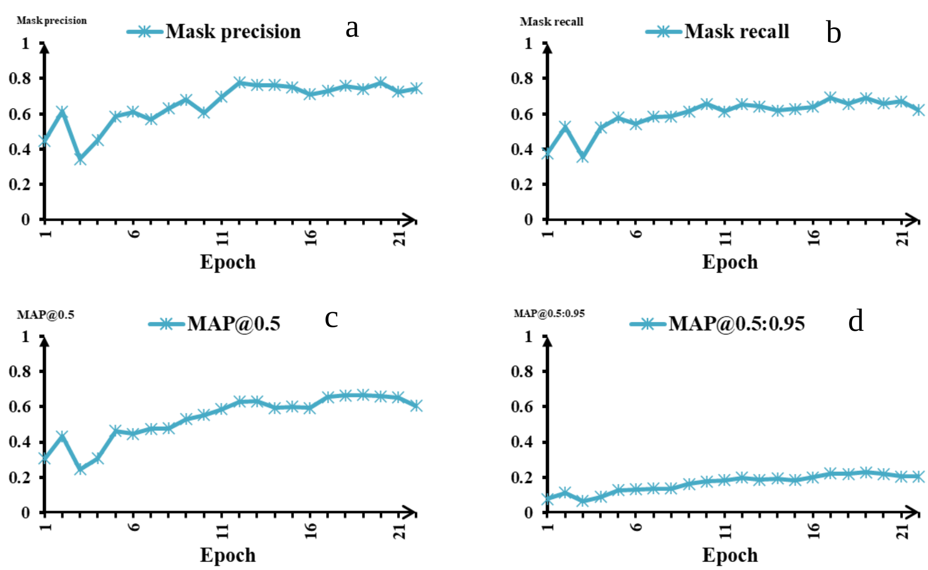

Figure 10: Performance metrics; a) Mask precision, b) Mask recall, c) MAP@0.5 and d) MAP@0.5:0.95

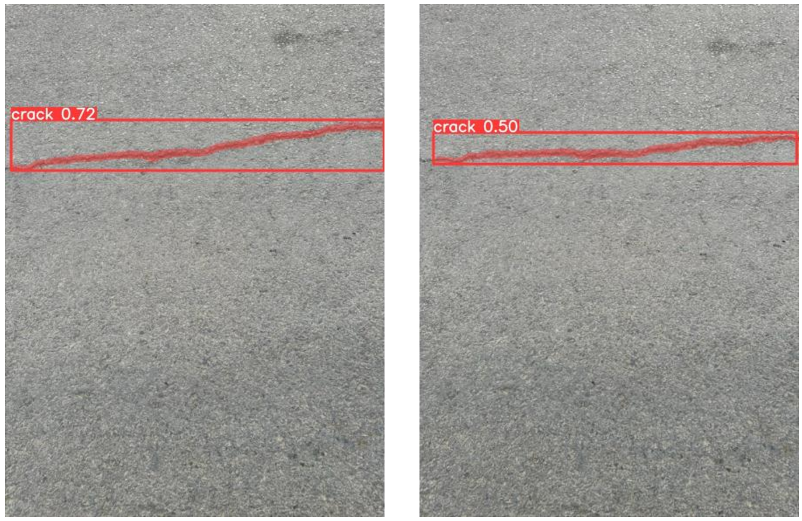

Figure 11: Predictions on Testing Data.

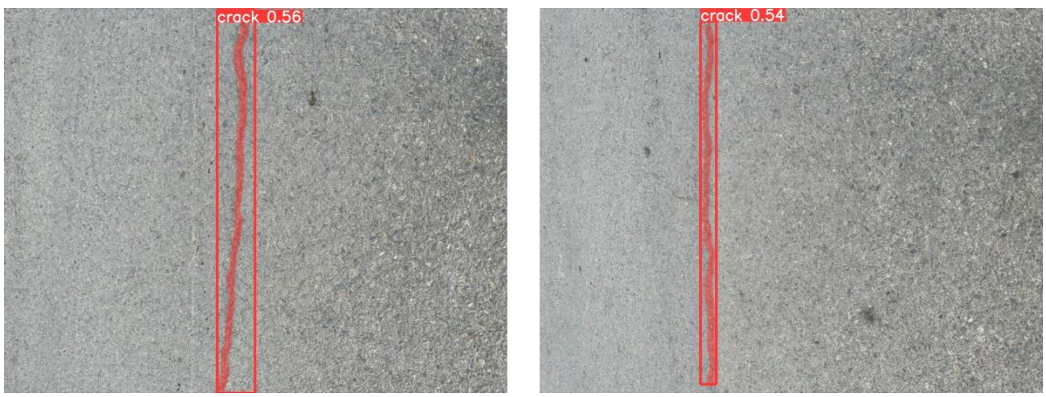

Figure 12: Predictions on Testing Data.

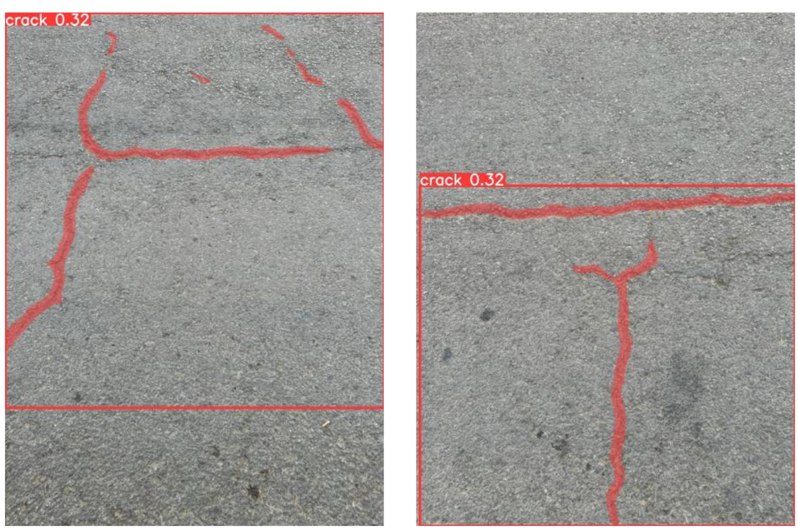

Figure 13: Predictions on Testing Data.

8 COCLUSION

Road and traffic safety are crucial, and detecting cracks in pavement plays a vital role in maintaining them. To tackle this issue, we propose using YOLOv7, regarded as one of the cutting-edge object identification models, to detect and classify pavement cracks. In order to ensure that our training data is clean and balanced, we applied various pre-processing methods like augmentations, resizing, and blurring. Our experimental results show off that the YOLOv7 model achieved an impressive detection accuracy of 90%. We conducted experiments using custom datasets and achieved precision and recall values of 0.854 and 0.755, respectively. Our research benchmarked recent similar studies and the proposed system yielded promising results, surpassing the benchmarks in several areas.

Author contributions

Fekry Ashraf did the following:

(design of the work, Data collection and tools, Data analysis and interpretation, Funding acquisition, Resources, Methodology, Drafting the article, Critical revision of the article, Final approval of the version to be published)

Mostafa Rabah did the following:

(Conception or design of the work, Data collection and tools, Data analysis and interpretation, Project administration, Supervision, Final approval of the version to be published)

Essam Ghanem did the following:

(Project administration, Supervision, Final approval of the version to be published)

FUNDING STATEMENT

No financial support was received.

Conflict of interest

The author declared that there are no potential conflicts of interest regarding the research authorship or publication of this article.

9 REFERENCES

[1] V. P. Tran, T. S. Tran, H. J. Lee, K. D. Kim, J. Baek, and T. T. Nguyen, “One stage detector (RetinaNet)-based crack detection for asphalt pavements considering pavement distresses and surface objects,” Journal of Civil Structural Health Monitoring, vol. 11, no. 1, pp. 205–222, 2020.

[2] X. Z. Xiang, Y. Q. Zhang, and A. El Saddik, “Pavement crack detection network based on pyramid structure and attention mechanism,” IET Image Processing, vol. 14, no. 8, pp. 1580– 1586, 2020.

[3] H. N. Nguyen, T. Y. Kam, and P. Y. Cheng, “An automatic approach for accurate edge detection of concrete crack utilizing 2D geometric features of crack,” Journal of Signal Processing Systems, vol. 77, no. 3, pp. 221–240, 2014.

[4] Z. Y. Yang, “The design of glass crack detection system based on image preprocessing technology,” in 2014 IEEE 7th Joint International Information Technology and Artificial Intelligence Conference, pp. 39–42, Chongqing, China, 2014.

[5] Y. Yi, J. Wang, W. Zhou, Y. Fang, J. Kong, and Y. Lu, “Joint graph optimization and projection learning for dimensionality reduction,” Pattern Recognition, vol. 92, pp. 258–273, 2019.

[6] S. Chambon and J.-M. Moliard, “Automatic Road Pavement Assessment with Image Processing: Review and Comparison,” International Journal of Geophysics, vol. 2011, pp. 1–20, 2011, doi: 10.1155/2011/989354.

[7] J. Tang and Y. Gu, “Automatic Crack Detection and Segmentation Using a Hybrid Algorithm for Road Distress Analysis,” in 2013 IEEE International Conference on Systems, Man, and Cybernetics, Oct. 2013, pp. 3026–3030. doi: 10.1109/SMC.2013.516.

[8] Naoki Tanaka and Kenji Uematsu, “A Crack Detection Method in Road Surface Images Using Morphology,” in MVA, 1998.

[9] H. Yong and Z. Chun-Xia, “A local binary pattern-based methods for pavement crack detection,” Journal of Pattern Recognition Research, vol. 5, no. 1, pp. 140–147, 2010.

[10] L. Zhang, F. Yang, Y. Daniel Zhang, and Y. J. Zhu, “Road crack detection using deep convolutional neural network,” in 2016 IEEE International Conference on Image Processing (ICIP), Sep. 2016, pp. 3708–3712. doi: 10.1109/ICIP.2016.7533052.

[11] A. Ghimire, S. Thapa, A. K. Jha, A. Kumar, A. Kumar, and S. Adhikari, “AI and IoT Solutions for Tackling COVID-19 Pandemic,” in 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), Nov. 2020, pp. 1083–1092. doi: 10.1109/ICECA49313.2020.9297454.

[12] D. Ma, H. Fang, B. Xue, F. Wang, M. A. Msekh, and C. Ling Chan, “Intelligent Detection Model Based on a Fully Convolutional Neural Network for Pavement Cracks,” Computer Modeling in Engineering & Sciences, vol. 123, no. 3, pp. 1267–1291, 2020, doi: 10.32604/cmes.2020.09122.

[13] Q. Q. Li and X. L. Liu, “Novel approach to pavement image segmentation based on neighboring diff erence histogram method,” in 2008 Congress on Image and Signal Processing, pp. 792–796, Sanya, China, 2008.

[14] M. Gavilán, D. Balcones, O. Marcos et al., “Adaptive Road crack detection system by pavement classification,” Sensors, vol. 11, no. 10, pp. 9628–9657, 2011.

[15] Q. Q. Li and Q. Z. Mao, “Land-borne pavement rapid test and measurement,” Journal of Transport Information and Safety, vol. 27, no. 1, pp. 7–10, 2009.

[16] Y. Hu, C. X. Zhao, and H. N. Wang, “Automatic pavement crack detection using texture and shape descriptors,” IETE Technical Review, vol. 27, no. 5, pp. 398–405, 2010.

[17] A.Cord and S. Chambon, “Automatic Road defect detection by textural pattern recognition based on AdaBoost,” Computer Aided Civil and Infrastructure Engineering, vol. 27, no. 4, pp. 244–259, 2012.

[18] Y. Shi, L. Cui, Z. Qi, F. Meng, and Z. Chen, “Automatic Road crack detection using random structured forests,” IEEET rans actions on Intelligent Transportation Systems, vol. 17, no. 12, pp. 3434–3445, 2016.

[19] N. D. Hoang and Q. L. Nguyen, “A novel method for asphalt pavement crack classification based on image processing and machine learning,” Engineering with Computers, vol. 35, no. 2, pp. 487–498, 2019.

[20] H. Maeda, Y. Sekimoto, T. Seto, T. Kashiyama, and H. Omata, “Road damage detection using deep neural networks with images captured through a smartphone,” 2018, http://arxiv .org/abs/1801.09454.

[21] X. Wang and Z. Hu, “Grid-based pavement crack analysis using deep learning,” in 2017 4th International Conference on Transportation Information and Safety (ICTIS), pp. 917–924, Banff , AB, Canada, 2017.

[22] A. Zhang, K. C. P. Wang, B. Li et al., “Automated pixel-level pavement crack detection on 3D asphalt surfaces using a deep-learning network,” Computer-Aided Civil and Infrastructure Engineering, vol. 32, no. 10, pp. 805–819, 2017.

[23] Y. Fei, K. C. P. Wang, A. Zhang et al., “Pixel-level cracking detection on 3D asphalt pavement images through deep learning-based Crack Net-V,” IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 1, pp. 273–284, 2020.

[24] Q. Zou, Z. Zhang, Q. Li, X. Qi, Q. Wang, and S. Wang, “Deep Crack: learning hierarchical convolutional features for crack detection,” IEEE Transactions on Image Processing, vol. 28, no. 3, pp. 1498–1512, 2019.

[25] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” Jun. 2015.

[26] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, “YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors,” Jul. 2022.