Indonesian Journal of Electrical Engineering and Informatics (IJEEI)
Vol. 11, No. 1, March 2023, pp. 119~132
ISSN: 2089-3272, DOI: 10.52549/ijeei.v11i1.4362

Efficient Pavement Crack Detection and Classification Using Custom YOLOv7 Model

Arselan Ashraf[1], Ali Sophian[2], Amir Akramin Shafie[2], Teddy Surya Gunawan[1], Norfarah Nadia Ismail[3], Ali Aryo Bawono[4]
[1]Department of Electrical and Computer Engineering, International Islamic University Malaysia, Malaysia
[2]Department of Mechatronics Engineering, International Islamic University Malaysia, Malaysia
[3]School of Civil Engineering, College of Engineering, Universiti Teknologi MARA, Malaysia
[4]Faculty of Rail, Transport, and Logistics, Technical University of Munich Asia, Singapore

Article Info

Article history:

Received Dec 1, 2022
Revised Dec 19, 2022
Accepted Jan 15, 2023

Keywords:

Deep Learning
Image Preprocessing
Pavement Crack Detection
YOLOv7

ABSTRACT

It is crucial to detect and classify pavement cracks as part of maintaining road safety. The inspection process for identifying and classifying cracks manually is tedious, time-consuming, and potentially dangerous for inspectors. As a result, an efficient automated approach for detecting road cracks is essential for this development. Numerous issues, such as variations in intensity, uneven data availability, the inefficacy of traditional approaches, and others, make it challenging to accomplish. This research has been carried out to contribute towards developing an efficient pavement crack detection and classification system. This study uses state of the art deep learning algorithm, customized YOLOv7 model. Data from two sources, RDD2022, a publicly available online dataset, and the second set of data gathered from the roads of Malaysia have been used in this investigation. In order to have balanced data for training, many image preprocessing techniques have been applied to the data, such as augmentations, scaling, blurring, etc. Experimental results demonstrate that the detection accuracy of the YOLOv7 model is significant, 92% on the RDD2022 dataset and 88% on our custom dataset. This study reports the outcomes of experiments conducted on both datasets. RDD2022 achieved a precision of 0.9523 and a recall of 0.9545. On the custom dataset, the resulting values for precision and recall were 0.93 and 0.9158, respectively. The results of this study were compared to those of other recent studies in the same field in order to establish a benchmark. Results from the proposed system were more encouraging and surpassed the benchmarking ones.

Corresponding Author:

Ali Sophian,
Department of Mechatronics Engineering,
International Islamic University Malaysia, Malaysia
Email: ali_sophian@iium.edu.my

1. INTRODUCTION

Millions of dollars are spent each year on the acquisition of various tools and technologies for the purpose of damage detection from vital infrastructure such as roads, bridges, buildings, etc.[1] . Natural calamities such as prolonged exposure to sunlight, rainfall, earthquakes, natural weathering, and regular use can greatly strain civil structures such as roads, bridges, and pavements. All of these factors have variable effects on pavement performance [2]. These occurrences can either result in the entire collapse of the structure or in physical damage, which is frequently manifested as cracks. Typically, pavement surface cracks appear at a microscopic scale [3]. These cracks make the pavement fragile, limit its load-bearing capacity, and cause surface discontinuities [4]. If these cracks are detected early, subsequent damages can be mitigated. Cracks that go undetected can grow through the surface and limit the pavement's lifespan, resulting in casualties, injuries, and economic loss.

The majority of early pavement crack detection and classification systems rely on manual operations. Manual techniques for crack identification include experts visually inspecting the pavement and using specific equipment to identify any faults in the pavement [5]. These approaches, however, are time-consuming and labor-intensive and have limited detection accuracy and certain related dangers [6]. Image-based crack detection algorithms have received a lot of attention in recent decades. Methods in early research were mainly based on the combining or refinement of traditional digital image processing techniques such as mathematical morphology[7], thresholding [8], and edge detection [9]. These methods are often based on photometric and geometric theories about crack image attributes [10]. Crack pixels are the darker pixels in a picture, which is the most notable photometric feature. Based on this, a global or local threshold value is determined to segment cracks and backgrounds [11]. These techniques, however, are quite susceptible to noise because they are implemented upon individual pixels. Other solutions consider geometrical information to overcome this problem. For example, crack continuity is thought to limit false detection [7]. Based on the local orientation, the local binary pattern operator is utilized to identify whether a pixel belongs to cracks [12]. The multiscale analysis uses the wavelet transform to differentiate crack regions from crack-free regions. These approaches are effective at detecting cracks, but they are not precise enough to discover all of the cracks in a picture.

With the growth of Artificial Intelligence and computer vision technologies, various attempts have been made to employ AI and computer vision technologies for automatic crack detection [13]. AI is now frequently employed to solve various real-world problems [14]. From solving complex technical challenges to applications in banking and healthcare, AI and machine vision have become ubiquitous [15]. Deep learning models can be used for the automatic identification and categorization of pavement cracks [16] due to the widespread application of machine learning, particularly deep learning, in industry and research. Several approaches relying on feature extraction and pattern recognition have been developed for crack identification [17] since the advent of machine learning. The performance of these approaches is excellent but highly reliant on the extracted features. Due to the complexity of pavement conditions, it is difficult to establish practical elements for all pavements. Deep Learning proves to have a significant impact on overcoming performance issues in the sphere of road maintenance. Inspired by deep learning, the research interest of various researchers has boosted significantly towards this sphere, which can be seen in Figure 1. Crack detection follows these three key steps: preprocessing, detection, and classification.

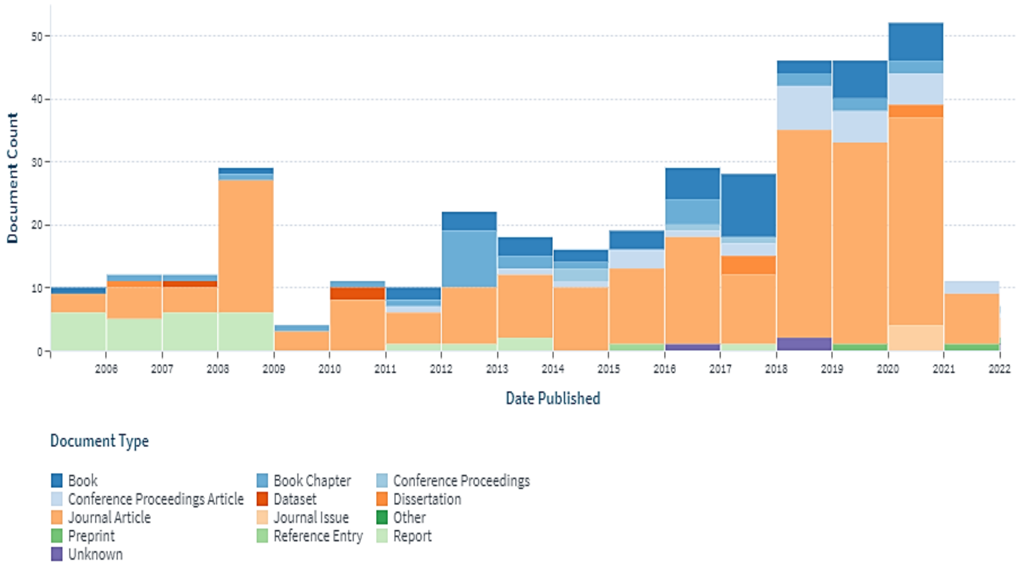

Figure 1. The graph on scholarly works in the field of road crack detection and classification using deep learning from 2006 to 2022

Most commonly, detection and classification in CNN methods are carried out simultaneously. By building new network models and learning from the acquired data, CNN techniques attain good results, but they need more processing resources than analytical or logical approaches. According to [18], they suggested a convolution neural network-based pavement automated detecting system termed CrackNet. The method is designed to extract fractures at the pixel level and has automated the identification of 3-dimensional concrete crack pavement. CrackNets lacked a pooling layer to lower the output of the layer below it, unlike conventional CNNs. CrackNet implemented the continuous image width and height methodology throughout all network layers to guarantee the accuracy of crack extraction. The detection accuracy of this method is much higher than that of the standard machine learning-based crack detection methodology. Impressed by CrackNet, [19] introduced CrackNet-V, an effective deep network built on top of the CrackNet used to detect cracks in 3D asphalt pavement images at the pixel level. With its more complex structure and fewer parameters, CrackNet-V improves upon the original CrackNet in terms of both the precision and speed with which it computes. CrackNet-V included the same space size for all layers to support supervised learning at the pixel level. CrackNet-V's success at autonomously identifying pavement fractures at the pixel level is another proof of the advantages of deep learning technology.

DeepCrack is an end-to-end trainable deep convolution neural network suggested by Ref. [20] for automated crack identification. Different convolution layers' multiscale deep convolution characteristics are merged together to generate a linear structure. This method produced entire image features in small-scale feature maps and detailed features in large-scale feature maps. DeepCrack is able to obtain an F-measure greater than 0.87 on the test dataset. An approach based on a Deep Convolutional Neural Network (DCNN) fusion model was presented by [21]. This model combines the benefits of the multitarget single-shot multibox detector (SSD) CNN model with the U-Net model. Using a deeper neural network, Ref. [22] successfully categorize patches as either cracks or non-cracks, proving the network's superiority. Ref. [23] suggests a deep active learning system to address the issue of insufficient label data. Critical insight into various literature related to this field can be examined, along with their strengths and limitations, in Table 1.

Table 1. Recent Related Works

Research Title	Method Used	Strength	Limitations	Reference
Road Damage Detection Using Deep Neural Networks with Images Captured Through a Smartphone	Deep Neural Networks	A self-generated dataset was used for training and testing the suggested system. A total of 163,664 images were collected after researchers and local governments in Japan worked together on the project.	The extent of the cracks was not taken into consideration. In addition, the recall and accuracy of the dataset were subpar, at 75%.	[24]
Pavement Crack Detection and Segmentation Method Based on Improved Deep Learning Fusion Model	Deep Convolutional Neural Network Fusion, U-Net model.	In this model, hyperparameters for crack classification were optimized. They constructed their own dataset and assessed it against the proposed model. Using pixel labelling and scanning, the length of a crack was determined by this approach.	The model had an average accuracy of about 78.7%, which was not very encouraging.	[21]
Detection of Road Cracks Using Convolutional Neural Networks and Threshold Segmentation	Convolutional Neural Network, Thresholding	Images have been used as input data, preprocessed, and subjected to threshold segmentation. The processed output is fed into a Convolutional Neural Network for further feature extraction and classification. The accuracy in training was determined to be 96.20 %, in validation to be 96.5 %, and in testing to be 94.5 %.	The detected cracks are not classified based on the type.	[6]
Feature Pyramid and Hierarchical Boosting Network for Pavement Crack Detection	Deep Supervision Learning	The feature pyramid and hierarchical boosting techniques were utilized in this method, which resulted in an improvement of the low-level characteristics. The time it takes for the model to recognize a crack is really short.	The outcomes of the experiment are evaluated extensively on time. The outcomes lack accuracy, recall, and precision.	[25]
Road Crack Detection Using Deep Convolutional Neural Network	Supervised Deep Convolutional Neural Network	This research employed CNN's which performed better than other classic approaches such as Boosting, SVM, etc. The recall performance metrics achieved a level of 92.51 %.	In terms of cost and real-world application, this research's viability is not encouraging. The length of the crack has not been addressed.	[13]
Automatic Detection of Cracks in Asphalt Pavement Using Deep Learning to Overcome Weaknesses in Images and GIS Visualization	Convolutional Neural Network	The suggested approach incorporates ResNet and a geographic information system. In addition, a mobile mapping system has been incorporated into this effort. The accuracy of the system was 94.3%.	Some false positives in the model exhibit cracks in photos without cracks. Crack picture pixel refinement was not particularly promising.	[26]

Deep learning approaches, such as Convolutional neural networks, have been shown to outperform conventional machine learning techniques [27]. All of these techniques can detect pavement cracks to a certain extent, but there are still issues to be resolved:

i. Feature extraction, which is still used by the majority of algorithms to extract crack information, is rather complex and programmatically challenging to implement [28].
ii. While deep learning-based pavement crack detection algorithms do exist, they are not yet specific enough to meet the "one generalized algorithm" criteria. Using a single dataset produces unreliable outcomes when the source image is captured using different techniques, equipment or on diverse road segments. Since they are not very adaptable [29].
iii. Complex environmental conditions have an impact on the robustness and precision of crack recognition algorithms [30].
iv. Although pavement crack detection methods employing CNNs exist, most models only perform a specific function like detection, which cannot be utilized directly for assessing road conditions [31].

In order to address the problems mentioned above and contribute to pavement maintenance, this paper proposes a method based on YOLOv7 for detecting and classifying pavement cracks. The aim is to train the deep learning model on the image data, which will be collected from the developed data acquisition setup. Apart from that online available data is also used to induce generalizability. The same model will perform the detection and classification of the pavement cracks simultaneously, thereby significantly improving model efficiency. The rest of the paper is organized as section 2 presents the proposed methodology of this work. Section 3 displays the experimental data, outcomes, and comparisons. The paper concludes with Section 4.

2. RESEARCH METHOD

In this paper, a pavement crack detection and classification model is proposed. Two types of data are used for this research; 1) Purposely collected road data through inspection vehicle-mounted camera setup, 2) Online available road crack image data. This research data is subjected to different preprocessing steps: image frames extraction, labelling, data augmentation, and resizing. The data is split into train, test, and validation samples which are then fed to the YoloV7 algorithm for training purposes. Once the training is complete, the trained model is subjected to testing for performance evaluation. The proposed research method is shown in Figure 2.

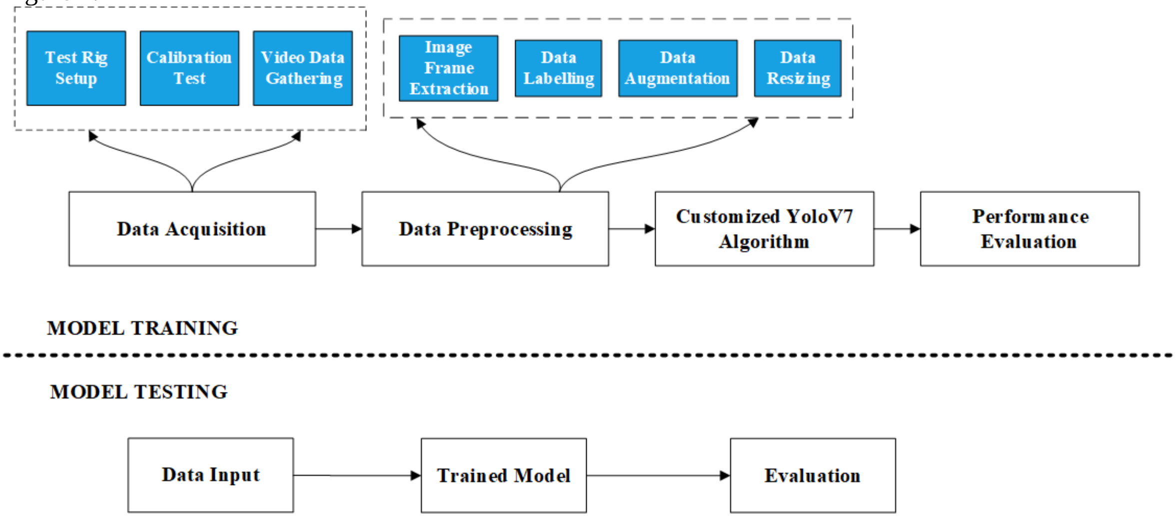

Figure 2. Proposed Research Methodology

2.1. Data Acquisition

The data for this research consists of an online dataset, RDD2022 [32], which incorporates 47,420 photographs of roads in six nations (Japan, India, the Czech Republic, Norway, the United States, and China). More than 55,000 occurrences of road damage have been marked in the images. Transverse cracks, longitudinal cracks, alligator cracks, and potholes are the four forms of road damage that are included in the data set. From RDD2022, data collected using smartphones mounted on motorbikes was used for this research. Apart from that, purposely collected road crack data from the roads around International Islamic University Malaysia, Gombak campus has also been used. The purposely collected data has been collected with a GoPro Hero 8 camera mounted on an inspection vehicle setup. Figure 3 shows the mounted camera setup used for data gathering.

Figure 3. Camera setup mounted on the inspection vehicle

The road data was collected with the aim of keeping it closer to the real-world scenarios. Therefore, the height and angle of the camera with respect to the ground were calibrated in such a way that it covers 3.1m road width, which is the average single road width. The camera specifications and calibration results are shown in Tables 2 and 3, respectively. Calibration Setup 2 was selected for data gathering, where the camera height was 1.6m.

Table 2. Camera Setup Specifications

Camera	GoPro Hero 8
Mount	GoPro Rod Mount
Mode	Custom Video
Resolution	1080 pixels
FPS	24, 60
Lens	Linear Angle
Bit Rate	Standard (45 Mbps)
ISO min	100
ISO max	1600

Table 3. Calibration Results

Setup	Width	Camera angle with respect to ground	Distance from the marked 3.1m length to the mounted camera setup	Height
1	3.1 m	90° ± 35°	1.1 m	1.30 m
2	3.1 m	90°	Directly above midpoint	1.6 m

2.2. Data Preprocessing

Before feeding the data to the deep learning algorithm for training, various preprocessing strategies are applied to the research data. The preprocessing steps utilized for this study are image frame extraction, Data labelling, Data Augmentation, and Data Resizing.

The purposely collected data was recorded in the form of videos. Therefore, image frames were extracted from the video clips. Images obtained were manually labelled using the Roboflow data annotation tool. The online available dataset RDD2022 contained labelled road crack images. The labels used for data annotation for both data sets are given in Table 4.

Table 4. Crack and Pothole Labels

Dataset	Longitudinal Crack	Transverse Crack	Alligator Crack	Potholes
RDD 2022	D00	D10	D20	D40
Custom Data	crack_long (1)	crack_trans (2)	crack_alligator (0)	Pothole (3)

The accuracy of the predictions made by Supervised Deep Learning models is highly dependent on the quantity and variety of training data. However, a lack of data is one of the most typical obstacles when creating deep learning models. Data augmentation is the process of enriching the quantity of data by producing additional data points from current data. This includes making minimal data changes or using machine learning models to produce additional data points to enhance the dataset. The data augmentation techniques used in this study were:

i. **Blurring:** A blur is a visual effect that causes the borders of text or pictures to look hazy or out of focus. Blur effect on images from dataset RDD2022 was set to $2px$ and for the custom data the effect was set to $4.5px$.
ii. **Brightness:** Brightness relates to the image's overall luminance or blackness. After a picture is captured by a digital camera or processed by an analog-to-digital converter, its brightness is a measurement of its intensity. For both the datasets, the data augmentation using image brightness was achieved by setting brightness values between $+20\%$ and -20%.

After performing data augmentation, the image samples were subjected to resizing. In computer vision, picture resizing is an essential preprocessing step. Typically, resizing an image comprises altering the image's visible dimensions. For example, the 1920×1080 pixels image can be converted to 480×270 pixels. Deep learning algorithms train more quickly with smaller-sized images, increasing efficiency. The images from both data sets were resized to 640×640 pixels.

The total number of data samples used for this research before and after employing preprocessing methods are presented in Table 5. Both the data sets were split into training, validation, and testing sets. The train-test split ratio used is 80:20, in which the test split has 10% test data and 10% validation data.

Table 5. Total image samples before and after data preprocessing

Dataset	Before Preprocessing	After Preprocessing
RDD2022	2477	3893
Custom data	470	851

2.3. Custom YOLOv7 Deep Learning Model

Deep Learning is an area of machine learning that focuses on algorithms based on the architecture and functioning of the brain, known as artificial neural networks. Deep learning is a machine learning approach that encourages computers to mimic what people do naturally: learn by example. Recently, there has been a great deal of focus on deep learning. It consists of obtaining outcomes that were previously impossible. In deep learning, a computational model learns directly from pictures, text, or voice to solve detection and categorization tasks. Models based on deep learning are capable of achieving state-of-the-art precision, occasionally surpassing human performance. Using a vast quantity of labeled data and multi-layered neural network architectures, deep learning models are trained.

The deep learning algorithm for this study is YOLOv7 [33]. A prominent family of real-time object identification techniques, YOLO is an acronym for "You Only Look Once." Single-stage object detectors are YOLO models. A Yolo architecture includes the head, neck, and spine/backbone. In a YOLO model, picture frames are characterized by a spine. These characteristics are merged and blended in the network's neck and then forwarded to the network's head, where YOLO predicts the positions and classifications of objects around which bounding boxes must be created. YOLO does post-processing using non-maximum suppression (NMS) in order to provide its final output. The basic architecture of yolo is shown in Figure 4.

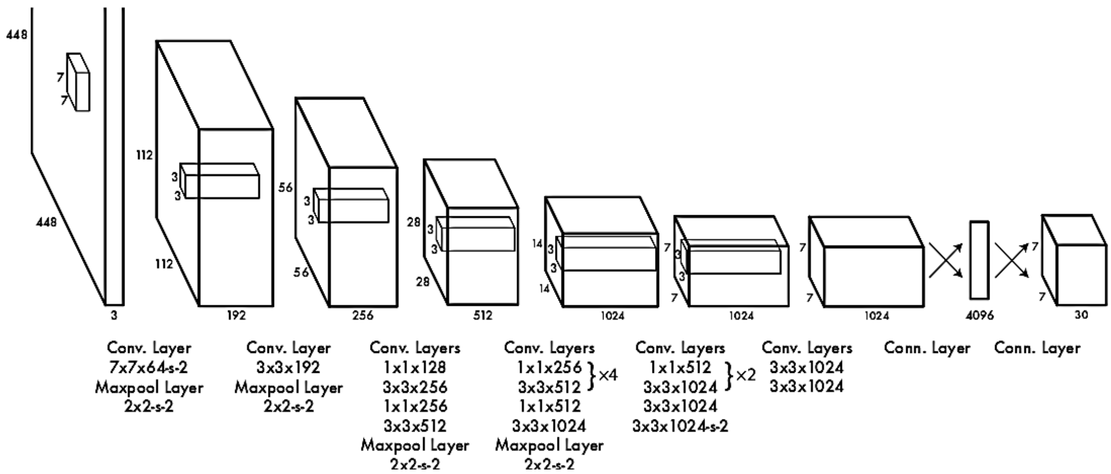

Figure 4. YOLO Architecture [34]

In this study, the input to the YOLOv7 model was the preprocessed image data of size 640×640 pixels. In order to customize the algorithm execution, the values assigned to the hyperparameters were set are shown in Table 6.

Table 6. Customized hyperparameter values

Hyperparameter	Value
Batch Size	20
Epochs	15 (for RDD2022) 50 (for Custom Dataset)
Initial Learning Rate	0.01
Final Learning Rate	0.1
Weight Decay	0.0005
Box loss gain	0.05
Cross Entropy Loss	0.3
Momentum	0.9

The reason for selecting YOLOv7 for this research is mainly due to Extended Efficient Layer Aggregation, Model Scaling Techniques, Re-parameterization Planning, and Auxiliary Head Coarse-to-Fine. The YOLO network's convolutional layers in the backbone must be highly efficient for fast inference. The developers of YOLOv7 expand upon previous work in this area, taking into account the distance a gradient must travel in order to back-propagate through the layers and the amount of memory required to store the layers. The faster the network can learn, the smaller the gradient has to be. Finally, E-ELAN, an enhanced variant of the ELAN computational block, was selected as the layer aggregation of choice, as shown in Figure 5.

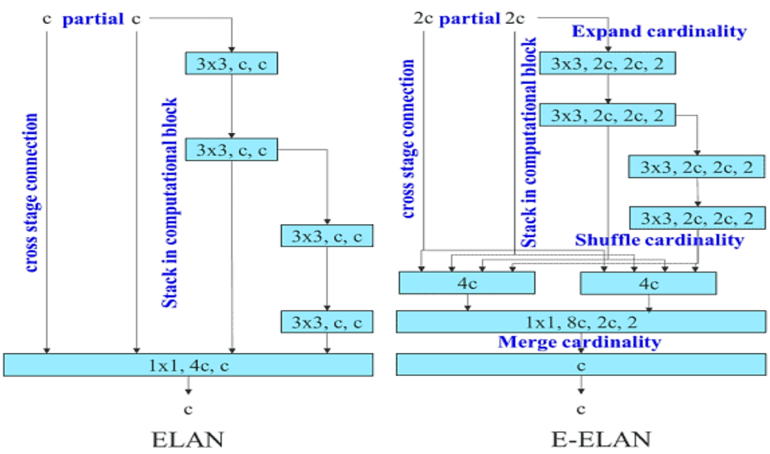

Figure 5. Evolution of layer aggregation strategies in YOLOv7 [33]

Network depth, breadth, and resolution used during training are typical metrics taken into account by object detection algorithms. The developers of YOLOv7 scale the network's depth and breadth simultaneously when concatenating layers shown in Figure 6. Research shows that this method maintains the best possible model design while scaling up or down.

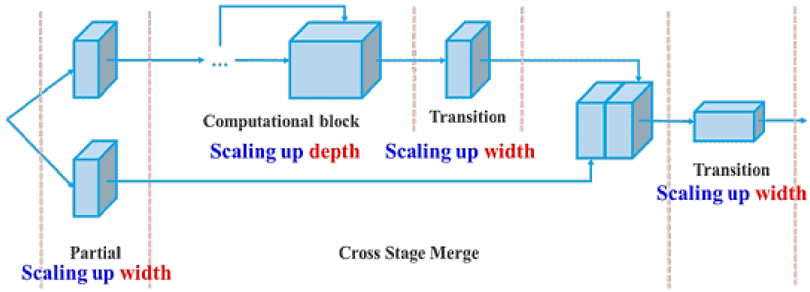

Figure 6. Scaling in YOLOv7 [33]

Methods of re-parameterization often include averaging a collection of model weights to produce a model with improved robustness to generic patterns. Recent studies have centered on re-parameterization at the module level, where individual nodes in the network employ their unique methods. When determining which modules of a network need to employ re-parameterization procedures, the YOLOv7 looks at the pathways of gradient flow. Although the YOLO network's predictions are ultimately made by the network's "head," which is located quite far down the chain of nodes, an auxiliary head located closer to the network's "center" can be useful. During training, monitoring both the prediction head and the detection head can be done. Since there is less network between the auxiliary head and the prediction, the developers of YOLOv7 try out various degrees of supervision for this head before settling on a coarse-to-fine definition in which supervision is passed back from the lead head at varying degrees of granularity in order to improve training efficiency as shown in figure 7.

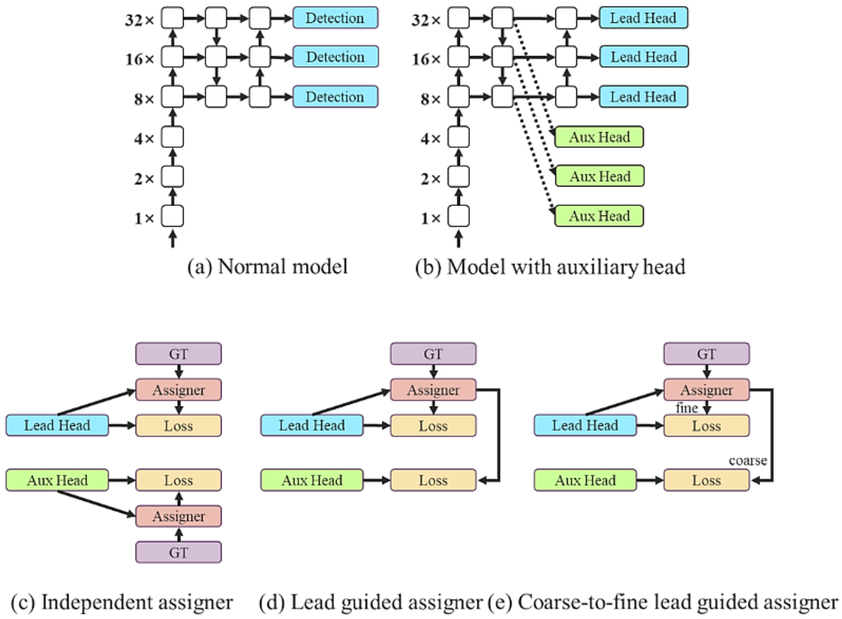

Figure 7. Coarse-to-fine auxiliary head supervision in the YOLOv7 [33]

3. RESULTS AND DISCUSSION

This research on developing an autonomous system for crack detection and identification using deep learning was carried out on a laptop with specifications presented in Table 7.

Table 7. Hardware Specifications

System	HP PAVILION 15-BC408TX
Central Processing Unit	Intel Core i7-8750H (8th Gen)
Graphics Processing Unit	NVIDIA GeForce GTX 1050
Random Access Memory	8 GB DDR4 RAM
HDD	1TB
SSD	512GB
Graphics Memory	4GB

3.1. Evaluation Matrices

In order to determine the performance of the developed model, various performance metrics are used in the machine and deep learning sphere. Performance evaluations are done using accuracy, precision, recall, confusion matrix, F1-score, and more.

The performance of the classification model may be measured with the help of the confusion matrix, an essential element in statistical analysis. Confusion matrix is a two-dimensional table including both estimated and actual values shown in Figure 8.

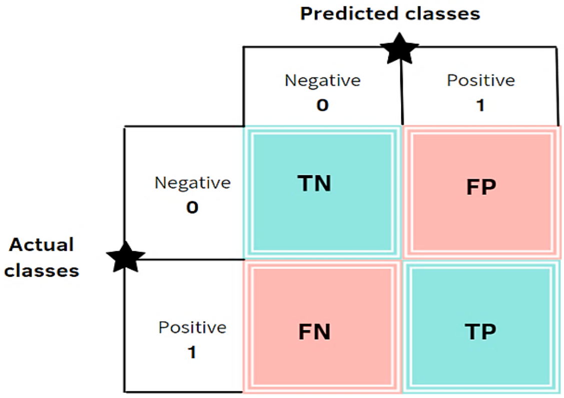

Figure 8. Confusion Matrix

True Positive (TP) – a class that is anticipated to be true and is really true. True Negative (TN) – a class projected to be false and really false. False Positive (FP) – a class projected to be positive but really false. False Negative (FN) – a class that is projected to be false but is really true.

Accuracy: Accuracy is the percentage of right predictions relative to the total number of predictions. It is the ratio of successful predictions to total estimates shown in Eq (1).

Precision: The level of precision indicates the fraction of successes among all positive predictions. It is determined by dividing the classifier's true positive (TP) rate by the sum of all true positives (TP + FP) shown in Eq (2).

Recall: The recall metric displays the fraction of all possible positive predictions that were really accurate. All True Positives are divided by the sum of True Positives and False Negatives shown in Eq (3).

F1-Score: The F1 Score uses a harmonic mean calculation to find a balance between accuracy and recall. It's a metric for assessing how well a test works, with 1 being the best possible result shown in Eq (4).

$$Accuracy = True\ Positive + True\ Negative\ /\ Total\ sample \quad (1)$$

$$Precision = True\ Positive\ /\ True\ Positive + False\ Positive \quad (2)$$

$$Recall = True\ Positive\ /\ True\ Positive + False\ Negative \quad (3)$$

$$F1 - Score = 2 \times Precision \times Recall\ /\ Precision + Recall \quad (4)$$

3.2. Experimental Results from RDD2022 Dataset

On the RDD2020 dataset, the customized YOLOv7 model was trained. The number of epochs utilized was 15, and the batch size was set at 20. Figure 9 depicts the outcomes of the experiment. The precision was measured at 0.9523, and its recall at 0.9545. The accuracy of the model was about 92%. The confusion matrix of the developed model is shown in Figure 10. Compared to previous models in the same domain, the proposed system's performance was significantly better. The individual accuracies of the crack classes along with the final accuracy, are presented in Table 8.

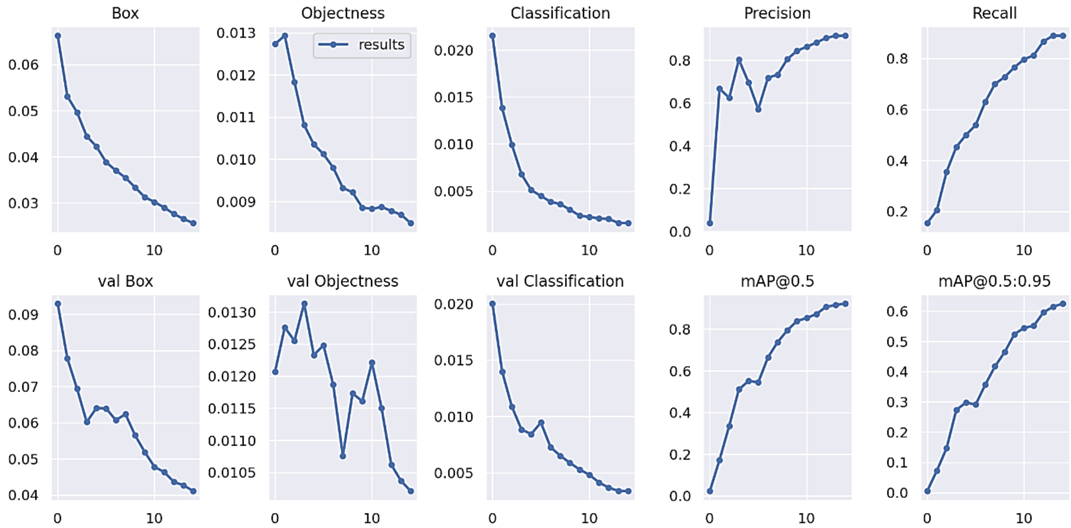

Figure 9. Performance metrics.

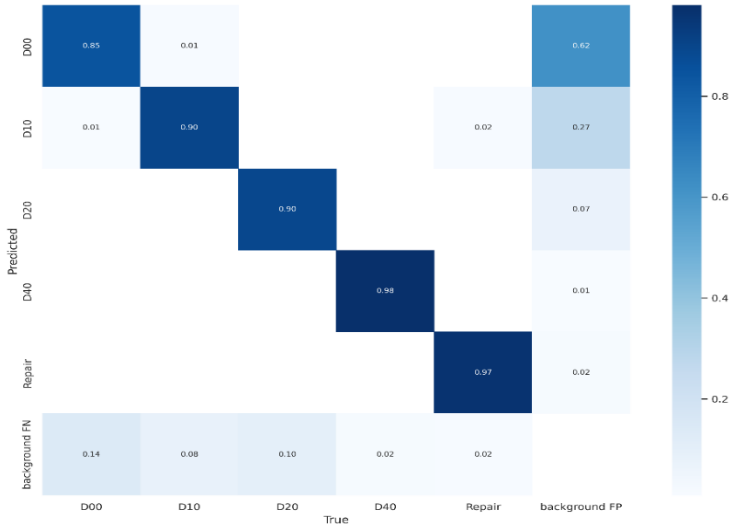

Figure 10. Confusion matrix.

Table 8. Accuracy of the model

Defect Type	Accuracy (%)
Alligator	90
Longitudinal	85
Transverse	90
Pothole	98
Repair	97
Total Accuracy = 92%	

This data set included five well represented data classes: longitudinal (D00), Alligator (D20), Transverse (D30), Potholes (D40), and Repair. The number of data samples provided for each class is the primary factor determining how accurate the individual classes are, representing the different types of cracks in this study. The developed model provided some promising outputs using optimal deep learning algorithms and image-processing techniques. Testing predictions from the developed system are presented in Figure 11.

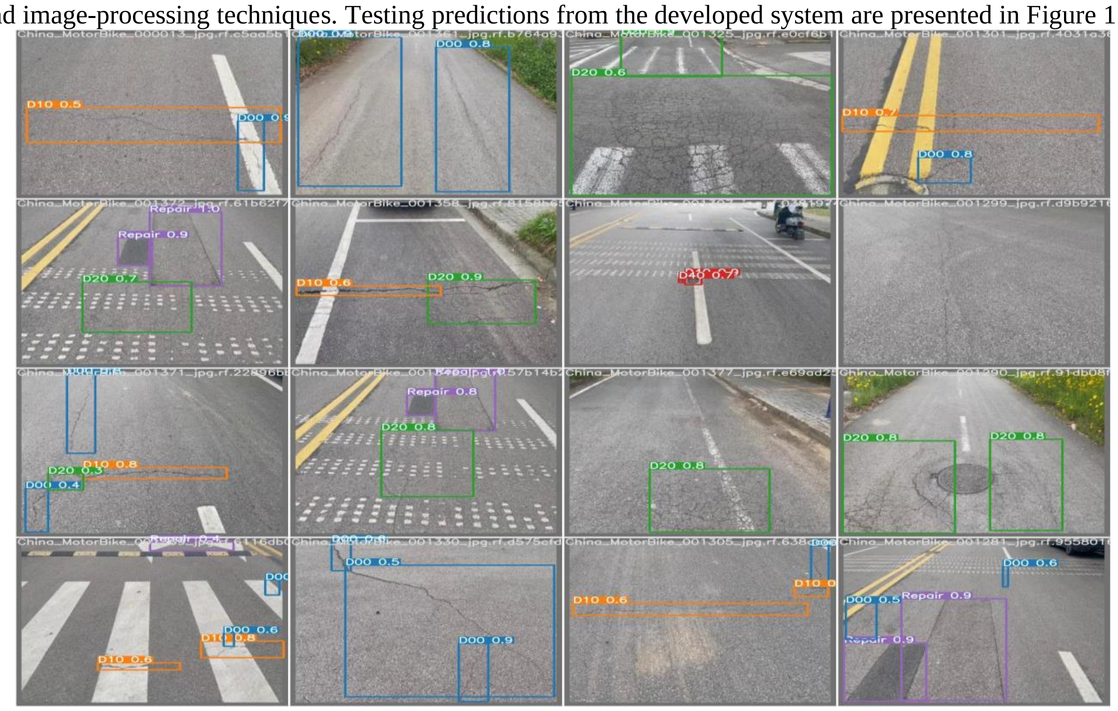

Figure 11. Predictions on Testing Data

3.3. Experimental Results from the Custom Dataset

Our custom YOLOv7 model was trained on the custom dataset. The batch size was set to 20, and the epochs used were 50. The results are shown in Figure 12. The model attained the highest precision of 0.93 on epochs 35 and 41, respectively, and the highest recall of 0.9158 on epoch 41. The results obtained were highly encouraging owing to the fact that a single model was able to perform both detection and classification on real-world data. Table 9 shows the accuracy of the individual crack types, i.e., alligator crack, longitudinal crack, transverse crack, and pothole, as well as the average for the developed model. The accuracy of the developed model on the custom dataset came out to be 88 %. The accuracy is promising for the custom data, which surpasses many existing models. Testing predictions from the developed system are presented in Figure 13.

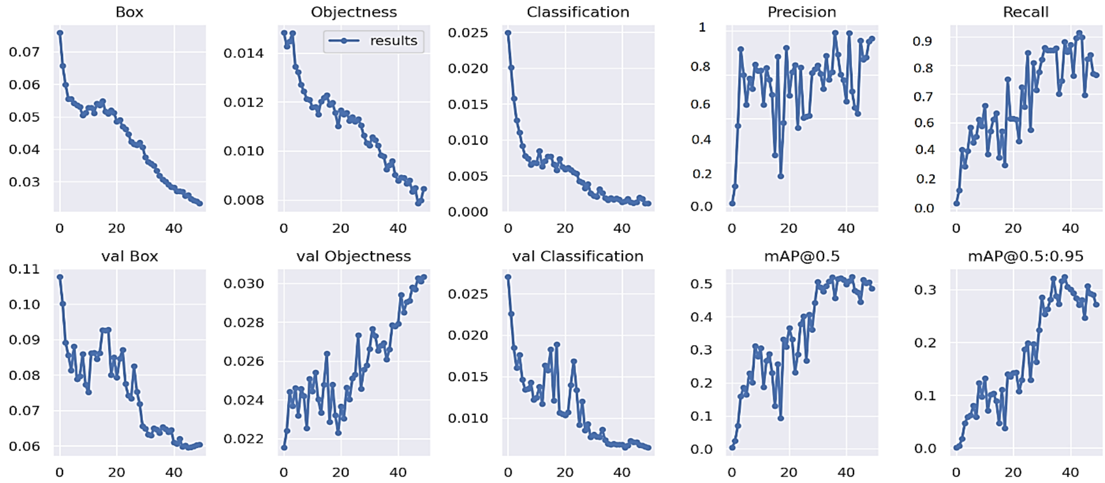

Figure 12. Performance metrics

Table 9. Accuracy of the model

Crack Type	Accuracy (%)
Alligator	91
Longitudinal	87
Transverse	84
Pothole	90
Total Accuracy = 88%	

Figure 13. Predictions on Testing Data

The accuracy of the individual classes, which are the crack types in this research, mostly depends on the number of data samples available for that class. Data samples for alligator cracks, longitudinal cracks, and potholes were adequate; therefore, the accuracy for these classes was also pretty good compared to the transverse cracks, in which data was comparatively less. The health of the customized data set can be seen in Figure 14; the health report has been generated from Roboflow. Roboflow is a platform for Computer Vision programmers that facilitates improved data collecting, preprocessing, and model training.

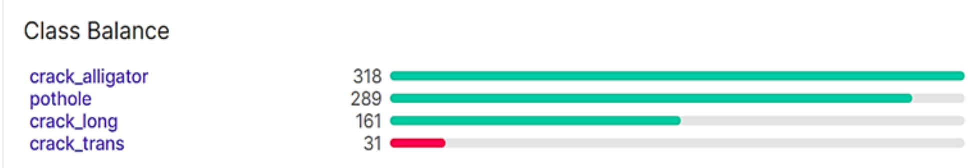

Figure 14. Dataset health representation

3.4. Benchmarking

This study was compared directly with others in the field. The outcomes of this study are more encouraging than those of comparable research shown in Table 10. The proposed model's superior performance may be attributed to its thorough execution, which included the selection of the most appropriate algorithms and configurations, implementing of more effective image processing techniques, and creating a conducive training environment.

Table 10. Benchmarking

S.No.	Technique Used	Result		Citation
01.	ConvNets	Recall = 0.9251 Precision = 0.8696		[13]
02.	Deep Neural Networks	Recall and Precision = 0.75		[24]
03.	Structured Prediction with the Convolutional Neural Network	Precision = 0.9178 Recall = 0.8812		[35]
04.	YOLOv5s	Precision = 0.891 Recall = 0.512		[36]
05.	**Proposed Model (Customized YOLOv7)**	**For RDD 2022**	**Recall = 0.9545 Precision = 0.9523**	
		For Custom Data	**Recall = 0.9158 Precision = 0.93**	

4. CONCLUSION

Given the importance of crack identification for maintaining roads and traffic safety, researching how to detect them has been a popular area of study for some time now. There have been several proposed solutions to this issue. In conjunction with one of the most advanced object identification models, YOLOv7, we address the prospect of applying this model to the detection and classification of cracks in the pavement. Two data types have been used in this study; RDD2022, an online accessible dataset, and purposely collected data from the roads around Gombak campus of International Islamic University Malaysia. Various preprocessing steps have been applied to the data, like augmentations, resizing, blurring, etc., to have balanced and clean data for training. The experimental findings indicate that the YOLOv7 model's detection accuracy exceeds 90%. The study presents the experimental results on both the datasets. The precision and recall from RDD2022 came out to be 0.9523 and 0.9545, respectively. Similarly, the precision and the recall on the custom dataset came out to be 0.93 and 0.9158, respectively. The results from this research were benchmarked with those of recent related ones. The proposed system surpassed them with some promising outcomes. This research was successful in addressing various open issues that were highlighted in the introductory section.

Future initiatives for inclusion can take into account a wider variety of modalities. Edge and distributed computing can be used to assess the proposed architecture. After cracks are identified and categorized, they may be given a GPS label. To improve the efficiency of the crack detection and characterization models, pixel-level segmentation might be used. Future research may incorporate a check for crack severity.

ACKNOWLEDGMENTS

The authors thank the Malaysian Ministry of Higher Education (MOHE) for financing the research project via FRGS/1/2021/TK02/UIAM/02/4. The author would also like to thank Kulliyyah of Engineering, International Islamic University Malaysia, for offering the KOE Postgraduate Tuition Fee Waiver Scheme and a research platform.

REFERENCES

[1] F. Ni, J. Zhang, and Z. Chen, “Pixel-level crack delineation in images with convolutional feature fusion,” *Struct Control Health Monit*, vol. 26, no. 1, p. e2286, Jan. 2019, doi: 10.1002/stc.2286.

[2] 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,” *J Civ Struct Health Monit*, vol. 11, no. 1, pp. 205–222, Feb. 2021, doi: 10.1007/s13349-020-00447-8.

[3] S.-T. Liong, Y. S. Gan, Y.-C. Huang, C.-A. Yuan, and H.-C. Chang, “Automatic Defect Segmentation on Leather with Deep Learning,” Mar. 2019.

[4] C. V. Dung and L. D. Anh, “Autonomous concrete crack detection using deep fully convolutional neural network,” *Autom Constr*, vol. 99, pp. 52–58, Mar. 2019, doi: 10.1016/j.autcon.2018.11.028.

[5] H. Oliveira and P. L. Correia, “Automatic Road Crack Detection and Characterization,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 155–168, Mar. 2013, doi: 10.1109/TITS.2012.2208630.

[6] A. Ashraf, A. Sophian, A. A. Shafie, T. S. Gunawan, N. N. Ismail, and A. A. Bawono, “Detection of Road Cracks Using Convolutional Neural Networks and Threshold Segmentation,” *Journal of Integrated and Advanced Engineering (JIAE)*, vol. 2, no. 2, pp. 123–134, Sep. 2022, doi: 10.51662/jiae.v2i2.82.

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

[8] H. Oliveira and P. L. Correia, “Automatic road crack segmentation using entropy and image dynamic thresholding,” in *2009 17th European Signal Processing Conference*, 2009, pp. 622–626.

[9] H. Zhao, G. Qin, and X. Wang, “Improvement of canny algorithm based on pavement edge detection,” in *2010 3rd International Congress on Image and Signal Processing*, Oct. 2010, pp. 964–967. doi: 10.1109/CISP.2010.5646923.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] A. Ghimire, S. Thapa, A. K. Jha, S. Adhikari, and A. Kumar, “Accelerating Business Growth with Big Data and Artificial Intelligence,” in *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, Oct. 2020, pp. 441–448. doi: 10.1109/I-SMAC49090.2020.9243318.

[16] 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.

[17] Y. Shi, L. Cui, Z. Qi, F. Meng, and Z. Chen, “Automatic Road Crack Detection Using Random Structured Forests,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 12, pp. 3434–3445, Dec. 2016, doi: 10.1109/TITS.2016.2552248.

[18] A. Zhang *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, Oct. 2017, doi: 10.1111/mice.12297.

[19] Y. Fei *et al.*, “Pixel-Level Cracking Detection on 3D Asphalt Pavement Images Through Deep-Learning- Based CrackNet-V,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 1, pp. 273–284, Jan. 2020, doi: 10.1109/TITS.2019.2891167.

[20] Q. Zou, Z. Zhang, Q. Li, X. Qi, Q. Wang, and S. Wang, “DeepCrack: Learning Hierarchical Convolutional Features for Crack Detection,” *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1498–1512, Mar. 2019, doi: 10.1109/TIP.2018.2878966.

[21] X. Feng *et al.*, “Pavement Crack Detection and Segmentation Method Based on Improved Deep Learning Fusion Model,” *Math Probl Eng*, vol. 2020, pp. 1–22, Dec. 2020, doi: 10.1155/2020/8515213.

[22] L. Pauly, H. Peel, S. Luo, D. Hogg, and R. Fuentes, “Deeper Networks for Pavement Crack Detection,” Jul. 2017. doi: 10.22260/ISARC2017/0066.

[23] C. Feng, M.-Y. Liu, C.-C. Kao, and T.-Y. Lee, “Deep Active Learning for Civil Infrastructure Defect Detection and Classification,” in *Computing in Civil Engineering 2017*, pp. 298–306. doi: 10.1061/9780784480823.036.

[24] H. Maeda, Y. Sekimoto, T. Seto, T. Kashiyama, and H. Omata, “Road Damage Detection Using Deep Neural Networks with Images Captured Through a Smartphone,” Jan. 2018, doi: 10.1111/mice.12387.

[25] F. Yang, L. Zhang, S. Yu, D. Prokhorov, X. Mei, and H. Ling, “Feature Pyramid and Hierarchical Boosting Network for Pavement Crack Detection,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 4, pp. 1525–1535, Apr. 2020, doi: 10.1109/TITS.2019.2910595.

[26] P. Chun, T. Yamane, and Y. Tsuzuki, “Automatic Detection of Cracks in Asphalt Pavement Using Deep Learning to Overcome Weaknesses in Images and GIS Visualization,” *Applied Sciences*, vol. 11, no. 3, p. 892, Jan. 2021, doi: 10.3390/app11030892.

[27] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[28] H. Oliveira and P. L. Correia, “CrackIT & #x2014; An image processing toolbox for crack detection and characterization,” in *2014 IEEE International Conference on Image Processing (ICIP)*, Oct. 2014, pp. 798–802. doi: 10.1109/ICIP.2014.7025160.

[29] S. Zhao and H. Ning, “Detection of pavement cracks based on convolutional neural network,” *Sensors and Microsystems*, vol. 36, no. 11, pp. 135–138, 2017.

[30] B. Qian, Z. Tang, X. Shen, and J. Guo, “Pavement crack detection based on multi feature manifold learning and matrix decomposition,” *Journal of Instrumentation*, vol. 37, no. 7, pp. 1639–1646, 2016.

[31] Z. Fan *et al.*, “Automatic Crack Detection on Road Pavements Using Encoder-Decoder Architecture,” *Materials*, vol. 13, no. 13, p. 2960, Jul. 2020, doi: 10.3390/ma13132960.

[32] D. Arya, H. Maeda, S. K. Ghosh, D. Toshniwal, and Y. Sekimoto, “RDD2022: A multi-national image dataset for automatic Road Damage Detection,” Sep. 2022.

[33] 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.

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

[35] Z. Fan, Y. Wu, J. Lu, and W. Li, “Automatic Pavement Crack Detection Based on Structured Prediction with the Convolutional Neural Network,” Feb. 2018.

[36] G. X. Hu, B. L. Hu, Z. Yang, L. Huang, and P. Li, “Pavement Crack Detection Method Based on Deep Learning Models,” *Wirel Commun Mob Comput*, vol. 2021, pp. 1–13, May 2021, doi: 10.1155/2021/5573590.