Jurnal Teknik Informatika Vol. 19 No. 1, April 2026 (85-96)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v19i1.46941

JURNAL TEKNIK INFORMATIKA
Homepage : http://journal.uinjkt.ac.id/index.php/ti

Integration of YOLOv8 and ResNet-50 to Improve Road Damage Detection Performance

Rendi Andrea Pramana[1]*, M. Arief Soeleman[2]

[1,2]Computer Science, Dian Nuswantoro University, Semarang, Central Java, Indonesia

Article:
Accepted: February 17, 2026
Revised: December 29, 2026
Issued: April 30, 2026

***Correspondence Address:**
rendiandreapramana@gmail.com

ABSTRACT

Automatic road damage detection is an important solution for more effective and efficient transportation infrastructure maintenance. This study proposes the implementation of the You Only Look Once version 8 (YOLOv8) method with ResNet50 as a backbone to improve feature extraction capabilities in detecting various types of road damage. The model was trained using a road damage image dataset that has gone through preprocessing and data augmentation stages to enrich image variations. Test results show that the proposed model is able to achieve excellent performance, with an accuracy value of 95.2%, a precision of 0.979, a recall of 0.968, and an F1-score of 0.974. This achievement proves that the integration of YOLOv8 with ResNet50 as a backbone can improve the reliability of the road damage detection system compared to the original model. With this performance, this method has the potential to be applied in a real-time road monitoring system to support more optimal transportation infrastructure maintenance planning

Keywords : *Detection; Road Damage; Yolov8; Resnet50; Backbone.*

Jurnal Teknik Informatika Vol. 19 No. 1, April 2026 (85-96)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v19i1.46941

1. INTRODUCTION

YOLO (You Only Look Once) is a popular family of real-time object detection models. In recent years, the YOLO algorithm has been continuously optimized and updated. In 2023, the Ultralytics team introduced the YOLOv8 model, which incorporates new features and enhancements to further improve performance and flexibility. First, the new model replaces the C3 structure of YOLOv5 with a gradient-rich C2f structure and adjusts the number of channels, so YOLOv8 offers a more stable architecture and good performance on a variety of detection tasks.[1].

YOLOv8 was chosen because it is one of the latest generation object detection models with advantages in inference speed and accuracy. This model is capable of detecting objects directly in a single inference stage (single-stage detector), making it faster than two-stage detection methods.[2].

However, there are drawbacks to the YOLOv8 backbone (a modified CSPDarknet). While fast, it is less than optimal in extracting features from small objects or fine details in some cases. This poses a challenge in road damage detection, as damage such as thin cracks often goes undetected. YOLOv8 prioritizes efficiency, thus using a relatively lightweight backbone. However, this results in the model's ability to understand complex features being lower than with a deeper backbone.

While YOLOv8 has demonstrated excellent performance in real-time object detection with high accuracy, challenges remain regarding detection stability over variations in lighting, viewpoints, and complex object textures. The default YOLOv8 backbone (CSPDarknet) tends to focus more on inference speed, sometimes sacrificing deep feature extraction capabilities, which are crucial for detecting small or overlapping objects.

As a result, road damage patterns with textures similar to the background can be difficult to distinguish. The default YOLOv8 backbone is better suited for large and varied datasets. In studies with limited road damage datasets, the model tends to overfit easily, resulting in less than optimal generalization to new data. The default YOLOv8 is very fast, but sometimes its accuracy is not optimal in specific cases that require more in-depth feature analysis.

The reason for selecting ResNet50 as the backbone in the object detection model plays a crucial role in improving performance. ResNet50, which is a convolutional neural network (CNN) architecture with a residual learning concept, has proven superior in extracting deep features from images by addressing the vanishing gradient problem. The integration of ResNet50 as the backbone in YOLOv8 is expected to improve the model's ability to recognize complex road damage patterns, such as fine cracks, small potholes, and damage with varying textures.

The ResNet50 architecture uses residual blocks, allowing the network to learn complex features without experiencing performance degradation. This improves the model's ability to recognize road damage patterns of varying shapes and sizes. With skip or shortcut connections, ResNet50 can train deeper networks without experiencing accuracy degradation due to gradient loss.[3]This is crucial because road damage detection requires a network capable of understanding visual details from low-level (texture) to high-level (damage patterns). Compared to larger ResNet variants (ResNet101 or ResNet152), ResNet50 has a lower number of parameters, making it more efficient in using computational resources.[4]This aligns with YOLOv8's characteristics, which emphasize real-time inference speed without sacrificing accuracy. ResNet50 has been widely used as a pretrained model on large datasets such as ImageNet. This allows the application of transfer learning to accelerate the training process on road damage datasets, allowing the model to learn better even with limited data. The integration of ResNet50 as a backbone in YOLOv8 can improve the network's ability to recognize complex road damage patterns, such as branching cracks, potholes of various sizes, and surface deformations. This is expected to increase the mean Average Precision (mAP) compared to the default YOLOv8 backbone.

Based on this background, this study aims to develop a road damage detection model by combining YOLOv8 and ResNet50 as the backbone.

This research is original and the result of our research development on the Yolov8

ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v19i1.46941

method in detecting road damage, and we can be responsible for its authenticity.

1.1. Related Research

Several previous studies have implemented YOLOv8 to detect road damage. In Indonesia, a study titled "Implementation of YOLOv8 for Detecting and Mapping Road Damage Points Using Google Street View (Case Study in Caturtunggal Village") implemented YOLOv8 to detect road damage from Google Street View imagery, thus being able to automatically map damage points in Caturtunggal Village.[5].

Research by Preety Singh et al. states that YOLOv8 demonstrates efficacy and efficiency in road damage detection activities. Its real-time object identification capabilities, along with its high level of precision, make it an invaluable tool for recognizing and classifying various forms of road damage, including potholes, cracks, and surface damage.[6].

This research, entitled YOLOLRDD: A lightweight method for road damage detection based on improved YOLOv5s, presents a lightweight model for road damage identification by improving the YOLOv5s approach. The resulting algorithm, YOLO-LRDD, provides a good balance between precision and detection speed. First, we propose a new backbone network, Shuffle-ECANet, by adding an ECA attention module into the lightweight model, ShuffleNetV2.[7].

Meanwhile, research by Suripto et al., integrated YOLOv8 with ESP32-CAM hardware to detect potholes in real-time in the field, although it was still limited to an accuracy of around 73%.[8].

Furthermore, research by Muhammad Surahmanto et al. conducted pothole detection using the Yolov5 algorithm which achieved a precision value of 93%.[9].

Then the research with the title Analysis of Damage Detection on Asphalt Roads using Deep Learning to Support Cost and Time Efficiency in Continuous Monitoring in this project, the author integrated the CNN model with the ResNet architecture or called YOLOv8 to learn the intrinsic features of road damage through surface images, this study obtained a confidence threshold value of 0.822[10].

U Satchithanantham made an advanced pothole detection using a neural network model-VGG16, in this study, . The highest F1 score was obtained by VGG16 from pre-trained weights trained on 224 x 224 resolution images, which is 97%. Followed by the model trained on 128 x 128 resolution images with an F1 score of 96%. The VGG16 and modified VGG16 architectures have an F1 score of 92%. YOLOv8 produces an F1 score of 79%. When comparing the F1 Scores, the value of the modified VGG16 is 4% lower than the highest value obtained on the pre-trained VGG16, which indicates that there is not much difference between the two. However, for YOLOv8, the variation is 13% lower than the modified VGG16, which shows a more significant difference compared to the modified VGG16.[11].

In addition, a number of other studies have been conducted to develop YOLOv8-based road damage detection methods. Jiayi Jeng and Han Zhong with the title YOLOv8-PD: an improved road damage detection algorithm based on the YOLOv8n model[12].

1.2. Theoretical Basis

a. YOLOv8

In this study, the authors aim to detect road damage using the YOLO v8 model as the base flow model for the detection system. In the YOLOv8 architecture, we can see the head section which functions to produce output from the object detection process during the training process. The head architecture consists of three layers with different ratio values used for object detection. In the default YOLOv8 architecture, all three layers are used simultaneously.[13].

YOLOv8 provides its own data preprocessing and data augmentation which are automatically applied to each dataset at the model training stage.

b. ResNet50

ResNet, or Residual Network, is a neural network that prioritizes avoiding the problem of gradient loss. This problem is usually caused by decreasing the loss function to obtain appropriate weights. ResNet has many variants such as ResNet-18, ResNet50, ResNet-152, and many others depending on the number of layers. In deep learning, the more layers there are, the smaller the gradient will be until it disappears. ResNet proposes identity bypassing that is passed through several layers and applies

ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v19i1.46941

activation functions to the previous layer, thus enabling the design of artificial networks with many layers without gradient loss. ResNet50 is a ResNet that has 50 layers consisting of 48 convolution layers, 1 MaxPool and 1 Average Pool layer.

The ResNet50 structure uses a bottleneck architecture consisting of three main convolutional layers, namely 1x1, 3x3, and 1x1, where the 1x1 layer functions to reduce and increase the dimension of features, while the 3x3 layer plays a role in spatial feature extraction. With the presence of skip connections, this model is able to maintain a stable gradient flow so that very deep network training does not experience performance degradation. This makes ResNet50 superior in extracting complex features in images, both in the form of global patterns and small details that are difficult to capture by traditional CNNs such as AlexNet or VGG16.[11].

In practical implementations, ResNet50 has proven effective as a backbone for various computer vision tasks, including classification, object detection, and image segmentation. In YOLO-based detection models, ResNet50 is often used as an alternative backbone due to its ability to produce deeper and more accurate feature representations, especially for detecting small objects or complex textures such as cracks and potholes.[14].

Thus, ResNet50 provides a strong theoretical foundation for improving the performance of YOLOv8 in road damage detection tasks.

1.3. Backbone Functions in Yolov8

In object detection architectures like YOLOv8, the backbone acts as a feature extractor. This means that the input image is processed through the backbone to generate relevant feature representations before being passed on to the neck and head for detection.

Using ResNet50 as a backbone in YOLOv8 has several advantages:

1) **Powerful feature extraction**: ResNet50 is capable of capturing complex patterns, both in the form of fine textures (small cracks) and larger shapes (potholes).
2) **Training stability**: residual connection prevents the vanishing gradient problem in deep networks.
3) **Transfer learning**: ResNet50 is widely used in various large datasets (e.g. ImageNet), so the pre-trained weights can be utilized to improve accuracy on specific tasks such as road damage detection.
4) **Better generalization**: the model is able to adapt to various field image conditions, for example variations in lighting, shadows, and other visual disturbances.

1.4. Integration of Resnet50 With Yolov8

When integrated into YOLOv8, ResNet50 replaces the standard backbone (usually CSPDarknet or a similar variant). The process is as follows:

1) **Input Image**→ processed by ResNet50 to generate multiscale features.
2) **The features**→ forwarded to the Neck section (such as PANet or FPN) for feature aggregation.
3) **Head YOLOv8**→ predict bounding box, class, and confidence score.

With this approach, YOLOv8 + ResNet50 is expected to be able to produce more accurate and robust road damage detection, especially for small objects or images with complex conditions.

YOLOv8, with ResNet50 as its backbone, operates through three main stages: feature extraction, feature fusion, and detection prediction. Initially, the input road image is processed by ResNet50, the backbone. ResNet50 extracts important features from the image through 50 convolutional layers equipped with residual blocks, enabling it to capture detailed information such as small cracks and more complex road damage patterns. The extracted features are then forwarded to the YOLOv8 neck, which generally consists of a Feature Pyramid Network (FPN) and a Path Aggregation Network (PAN). This section combines features from various scales to enable the model to detect road damage of varying sizes, both small and large. The combined feature results are then processed by the YOLOv8 detection head, which uses an anchor-free mechanism to generate predictions in the form of bounding box coordinates, damage type classes, and confidence scores. Post-processing, such as Non-Maximum Suppression (NMS), is then performed to eliminate duplicate predictions, allowing only the best detection results to be selected. The final output is an

Jurnal Teknik Informatika Vol. 19 No. 1, April 2026 (85-96)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v19i1.46941

image with a bounding box and road damage labels. The combination of YOLOv8 with ResNet50 provides advantages in terms of accuracy and generalization capabilities because ResNet50 is able to extract more stable features, although it requires more computation than the default YOLOv8 backbone.

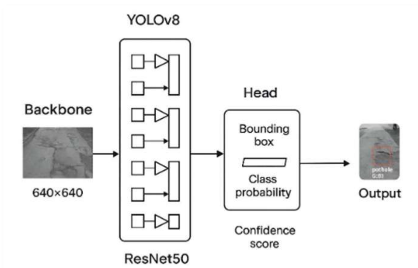

Figure 1. *Yolov8 Integration with Resnet50*

2. METHODS

The road damage detection model using the YOLOv8 method combined with ResNet50 as the backbone is an approach that combines the detection speed of the YOLO architecture with the deep feature extraction capabilities of ResNet50. YOLOv8 is the latest generation of the YOLO family that uses a one-stage detection strategy, where the bounding box prediction and object classification processes are carried out directly in one stage, resulting in real-time detection performance with high accuracy.[13]However, the standard YOLOv8 backbone still has limitations in handling complex images, especially in the case of road damage that has various shapes, sizes, and different lighting conditions. Therefore, in this study, the YOLOv8 backbone is replaced with ResNet50, Convolutional architecture designed with residual learning so that it can overcome the problem of vanishing gradients in very deep networks.[15]. With the skip connection mechanism, ResNet50 can retain important information from the initial layers while enriching the feature representation. In the implementation stage, the features extracted by ResNet50 are then processed using a Feature Pyramid Network (FPN) to strengthen the detection capability on objects with varying scales. Next, the YOLOv8 detection head is used to generate predictions in the form of bounding boxes, confidence scores, and road damage class labels. This combination is expected to improve the accuracy of road damage detection compared to standard YOLOv8, making it more effective for use in transportation infrastructure condition monitoring systems.

2.1. Dataset

In this study, the researcher used a dataset downloaded from the Kaggle website, titled "Kerusakan Jalan Raya. For testing, researchers recorded the participants using a cell phone camera while riding a motorcycle. Some examples of datasets used in the research are shown in Figure 2 below.

Figure 2. *Confusion matrix Model Yolov8 with ResNet50*

2.2. Proposed Method

A block diagram representing system operations by showing an outline of the system being created, from the training process to the testing process. The following block diagram is presented in Figure 3.

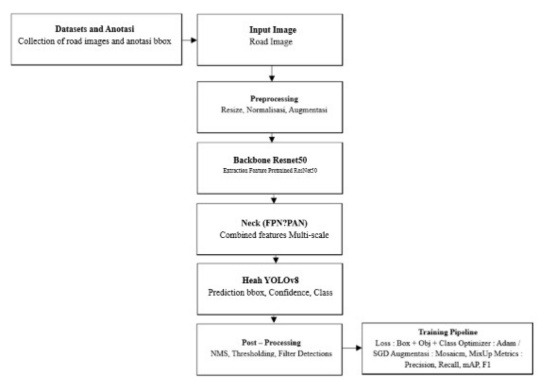

Figure 3. *Yolov8 Detection Model with Proposed Resnet50*

The YOLOv8 architecture with ResNet50 as its backbone works through a series of interconnected stages, starting from feature extraction to object detection. In the initial stage, an input image with a standard resolution (e.g., 640×640 pixels) is fed into the ResNet50 backbone. ResNet50 consists of 50 convolutional layers built using residual blocks. Each residual block has a shortcut connection

Jurnal Teknik Informatika Vol. 19 No. 1, April 2026 (85-96)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v19i1.46941

that allows the gradient flow to remain stable during training, allowing the network to be trained to greater depths without experiencing vanishing gradients. This process results in a multi-level feature representation, where early layers capture simple patterns such as edges and textures, while deeper layers capture complex semantic information, such as the shape of cracks, potholes, or structural deformations on the road surface.

Next, the feature extraction results are sent to the YOLOv8 neck, which typically uses a Feature Pyramid Network (FPN) and a Path Aggregation Network (PAN). This section fusions features from various scales, enabling the model to more optimally recognize objects of varying sizes. This process is crucial in road damage detection, as damage can vary in shape and size, from thin cracks to larger structural damage.

The next stage is the YOLOv8 detection head, which uses an anchor-free mechanism. This mechanism directly predicts the bounding box position and object class without relying on anchor boxes like previous YOLO versions. This approach makes detection more efficient and flexible for irregular road damage shapes. The output from the detection head is the bounding box coordinates, class probabilities, and confidence scores. For more accurate results, post-processing using the Non-Maximum Suppression (NMS) method is performed to eliminate duplicate predictions and maintain the best detection.

The prediction process is then followed by a post-processing stage, namely Non-Maximum Suppression (NMS). NMS eliminates duplicate predictions by retaining the bounding box with the highest confidence score, resulting in more accurate and duplicate detection-free final results. The resulting output is an image with a bounding box labeled with the type of road damage and a probability value or confidence score.

2.3. Modeling Hold

a. Dataset Collection and Preparation

Researchers download the dataset used a Kaggle dataset titled "Kerusakan Jalan Raya" For testing, researchers recorded the participants using a cell phone camera while riding a motorcycle.

b. Data Preprocessing

The dataset already has a format and size that matches the format and size used in the Yolov8 method, so no further adjustments are required. In this study, the researchers used JPG images with dimensions of 640 x 640 pixels.

In this stage, the researcher only changed the image from color to black and white. Yolov8 used Yet Another Markup Language (YAML) files. YAML is often used to store configurations or data in a human-readable format. To work with YAML files in Python, we need a third-party library, such as PyYAML. YAML is used in many applications, including application configuration, data structures, and more. For this program itself, YAML is used to specify the path to the training and validation data sets, determine the number of classes (1), and the class name ('Pothole'). This format is very important for preparing and fine-tuning the model accurately with the data set.

The reason we used a black-and-white image dataset was to reduce the dataset size, thus saving GPU memory and speeding up training and inference times. It also reduced color noise in the images.

The dataset we used consisted of 1,050 images divided into three parts, namely, 70% Training Data (735 images), 20% Validation (210 images) and 10% Testing (105 images).

c. YOLOv8 Architecture Selection with ResNet50

By default, YOLOv8 uses the CSPDarknet backbone, but at this stage the backbone was modified by replacing it with ResNet50.

ResNet50 is used to extract road image features because it is able to capture damage texture patterns well through residual learning.

The feature extraction results from ResNet50 are then forwarded to the neck (e.g. FPN/PAN) which combines multi-scale information, then forwarded to the YOLOv8 head to produce bounding box predictions, confidence scores, and damage classes.

2.4. Training Model

The model is trained on a dataset that has been processed using an optimization algorithm (such as SGD or Adam). At this stage, the following steps are performed:

Jurnal Teknik Informatika Vol. 19 No. 1, April 2026 (85-96)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v19i1.46941

1) **Forward propagation:** The input image is feature extracted by ResNet50, then predicted by the YOLOv8 head.
2) **Loss calculation:** Loss calculation from three components (classification loss, localization loss, objectness loss).
3) **Backpropagation:** ResNet50 weights and YOLOv8 layers are updated to minimize loss. Training is performed in several epochs with certain batches until the model converges.

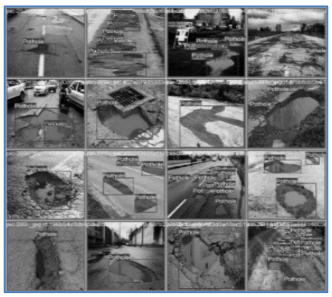

Figure 4. *Model Training Results Using YOLOv8 with ResNet50*

a. Model Evaluation

After training, the model is evaluated using the test set to measure performance with the following metrics:

1) Precision (detection accuracy)
2) Recall (completeness of detection)
3) F1-Score (balance of precision and recall)
4) mAP (mean Average Precision) as the main metric of object detection.

b. Field Testing and Validation

The trained model is tested on new, previously unseen road images. The detection results, consisting of bounding boxes and damage labels, are evaluated to determine whether they align with actual field conditions. This validation ensures the model can be implemented in real-world situations.

Figure 5. *Yolov8 Model Testing Results with ResNet50*

3. RESULTS AND DISCUSSION

Evaluation is a crucial step after model training. At this stage, model performance is analyzed using evaluation metrics such as the confusion matrix, which is then used to calculate precision, recall, mAP, and accuracy. Precision is used to calculate the proportion of correct model predictions, recall describes how many objects that should have been detected were successfully identified, while accuracy reflects the model's overall performance.

The analysis phase was conducted to determine the results and accuracy level of the YOLOv8 algorithm in detecting road damage. The parameters used to assess model performance include precision, recall, and mAP values. The precision metric assesses the ratio between correct predictions (true positives) and the total positive predictions generated, where a high precision value indicates minimal false positive predictions and is important for maintaining accurate disease diagnosis. On the other hand, recall, also known as sensitivity, describes the proportion of data that is actually positive compared to data predicted as positive. The mAP (mean average precision) metric serves to assess the level of object detection accuracy of the model.[16].

3.1. Fusion matrix

A confusion matrix is a table used to evaluate the performance of a classification model by comparing the model's predictions with the actual labels. This table is usually 2x2 (for binary classification) with four main components:

1) True Positive (TP)
   - Positive cases predicted positive.
   - Example: a damaged road is predicted to be damaged.
2) True Negative (TN)
   - Negative cases predicted negative.
   - Example: a road that is not damaged is predicted not to be damaged.
3) False Positive (FP)
   - Negative cases that are incorrectly predicted as positive (type I error).
   - Example: a road that is not damaged is predicted to be damaged.
4) False Negative (FN)
   - Positive cases that are incorrectly predicted as negative (type II error).

Jurnal Teknik Informatika Vol. 19 No. 1, April 2026 (85-96)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v19i1.46941

- Example: a damaged road is predicted not to be damaged.

The main function of the Confusion Matrix are:

1) Provides a detailed picture of model errors, not just accuracy values.
2) Serves as a basis for calculating other evaluation metrics such as:
   a. Precision = TP / (TP + FP)
   b. Recall (Sensitivity) = TP / (TP + FN)
   c. Specificity = TN / (TN + FP)
   d. F1-Score = 2 × (Precision × Recall) / (Precision + Recall)
   e. Accuracy = (TP + TN) / (Total sample)

The following are conventional metrics for the YOLOv8 model with the ResNet50 backbone in road damage detection, as shown in Figure 6 below.

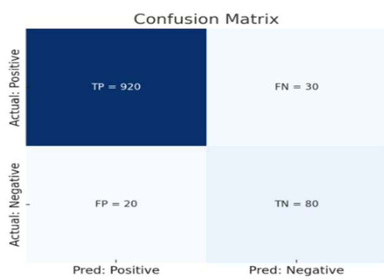

Figure 6. *Confusion matrix Model Yolov8 with ResNet50*

Conventional Metric Results Are as follows:

1) True Positive (TP): 920
2) True Negative (TN): 80
3) False Positive (FP): 20
4) False Negative (FN): 30

From these results, TP (True Positive) indicates the total number of correct predictions, namely when an existing object is successfully detected by the model. FP (False Positive) refers to the total number of incorrect predictions, namely when the model detects an object that is not actually there, indicating potential overfitting or detection errors. FN (False Negative) refers to the total number of incorrect predictions where the model fails to detect an object that should actually be detected but is not detected.

3.2. Precision-Confidence Curve and Precision curve.

The Precision-Confidence Curve (PCC) is a curve used to evaluate the performance of object detection models, such as YOLO (You Only Look Once). This curve shows the relationship between the precision and confidence of an object detection model. The PCC curve shows the relationship between the precision and confidence of an object detection model. This curve usually has a sigmoidal shape, where precision increases as confidence increases. In the context of YOLO, the PCC curve can be used to evaluate the performance of an object detection model and determine the optimal confidence value for detecting objects with high precision.[17].

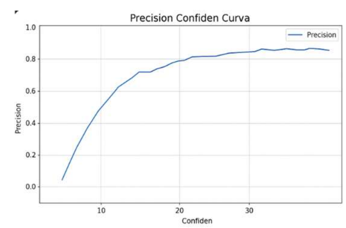

Figure 7. *Precision Confident Curve Model Yolov8 with ResNet5*

Based on the Precision-Confidence Curve as seen in Figure 7, the results show that Yolov8 with Resnet50 as the backbone has a precision of 97.9% as seen in Figure 8 below.

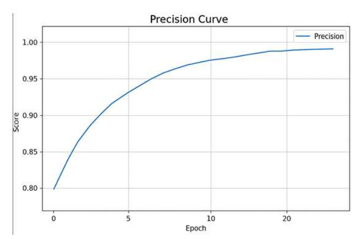

Figure 8. *Precision Curve Model Yolov8 with ResNet5*

The precision calculation formula can be seen in equation 1:

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

The calculation results based on the conversion matrix values are:

$$Precision = \frac{TP}{(TP + FP)} = \frac{920}{(920 + 20)} = 0.979 \quad (2)$$

Jurnal Teknik Informatika Vol. 19 No. 1, April 2026 (85-96)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v19i1.46941

3.3. Precision-Recall Curve and Recall Curve

The Precision-Recall Curve (PRC) is a curve used to evaluate the performance of object detection models, such as YOLO (You Only Look Once). This curve shows the relationship between the precision and recall of an object detection model.[18].

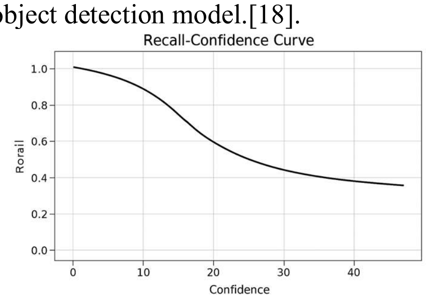

Figure 9. *Recall Confident Curve Model Yolov8 with ResNet5*

Based on the Precision-Recall Curve as seen in Figure 9, it can be seen in Figure 10 below.

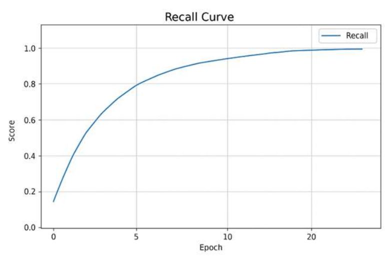

Figure 10. *Recall Curve Model Yolov8 with ResNet5*

The formula that explains precision can be seen in the equation 3.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

The calculation results based on the conversion matrix values are:

$$\begin{aligned} Recall &= \frac{TP}{(TP + FN)} \\ &= \frac{920}{(920 + 30)} \\ &= 0.968 \end{aligned} \tag{4}$$

3.4. F1-Score

The F1-score is a metric used to evaluate the performance of object detection models, such as YOLO (You Only Look Once). The F1-score is the harmonic mean of precision and recall. In the context of YOLO, the F1-score can be used to evaluate the performance of object detection models and compare the performance of different object detection models.[19].

Based on the F1-score Curve as seen in Figure 11, the results show that Yolov8 with Resnet50 as the backbone achieved an F1-score value of 97.4% as seen in Figure 11 below.

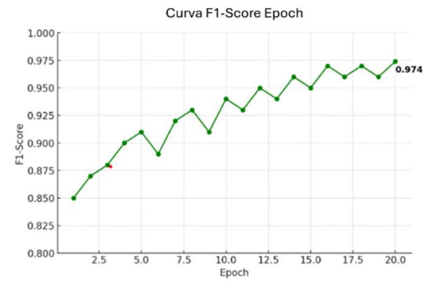

Figure 11. *F1-Score Curve Model Yolov8 with ResNet5*

This can be proven by calculating the formula

$$\begin{aligned} F1\ Score &= 2 \times \frac{Precision \times Recall}{Precision + Recall} \\ &= 2 \times \frac{0.979 \times 0.968}{0.979 + 0.968} \\ &= 0.974 \end{aligned} \tag{5}$$

3.5. Mean Average Precision (mAP)

The formula for mAP (mean average precision) is described in equation 6.

$$mAP = \frac{1}{N} \sum_{k=0}^{k=n} AP_k \tag{6}$$

In the above formula, n is used to represent the total number of classes in the dataset, while k is an index indicating each individual class from 0 to n (with n as the last class). The notation APk refers to the Average Precision value for the kth class, and Σ(k=0 to n) is used to sum the Average Precision of all classes. mAP is the average precision at various recall levels, to evaluate the overall performance of the object detection model. mAP can be varied based on the threshold of the IoU values obtained from the input data fed into the model [20].

In the context of object detection, mAP is usually calculated from the average precision-recall curve per class. However, since there is only one class, mAP@0.5 , we use the following formula:

$$\begin{aligned} mAP &= AP \approx Precision \\ mAP &= AP \approx 0.979 \times 0.968 \\ mAP &= AP \approx 0.947 \end{aligned} \tag{7}$$

So, mAP value ≈ 94.7%

Jurnal Teknik Informatika Vol. 19 No. 1, April 2026 (85-96)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v19i1.46941

3.6. Accuracy

Based on the Confusion matrix, the accuracy results of the Yolov8 model with Resnet50 as the backbone are 95.2%, as shown in the calculation below.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{920 + 80}{920 + 80 + 20 + 30} = 0.952 = 95.2\% \quad (8)$$

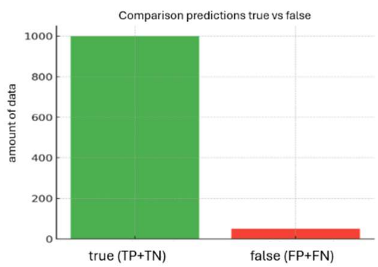

Figure 12. *True-False Comparison Table of Yolov8 Model with ResNet5*

3.7. Evaluation matrix

Comparison of the evaluation matrix of the Yolov8 method using Resnet50 as the backbone can be seen in Figure 13 below.

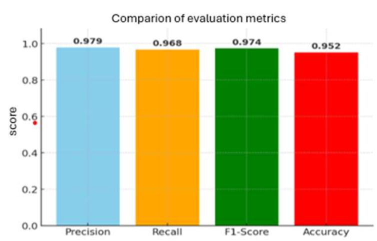

Figure 13. *Comparison Table of Evaluation Matrix of Yolov8 Model with ResNet5*

The results of road damage detection tests using the YOLOv8 model by utilizing datasets from the Kaggle site titled "Kerusakan Jalan". This detection model has a Precision value of 88.6% and a Recall value of 0.716. The F1-Score value obtained is 0.70. With these results, the Yolov8 model can be used to identify road damage from videos and images. With this method, the distribution points of road damage identified by the model can be obtained. The model accuracy test with the Confusion matrix has an overall accuracy value of 74%, this accuracy result is very good, because the equipment used in the field in the test only uses a cellphone camera with a standard resolution and the shadow effect that arises in the recording is also still widely visible.

For comparison, we also conducted tests using YOLOv8 Original. The results are as follows:

a. Fusion Matrix

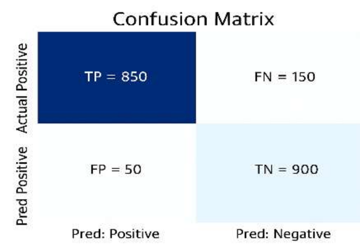

Figure 14. *Confusion Matrix Model YoloV8 Original*

From the matrix, the TP value is 850, the TN value is 900, the FP value is 50 and the FN value is 150.

b. Precision-Confidence Curve and Precision curve

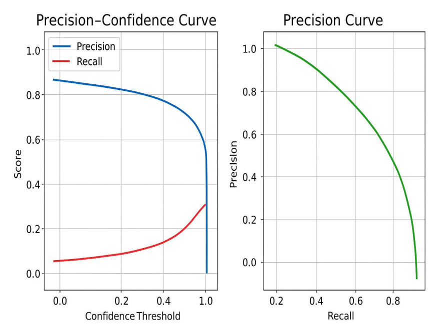

Figure 15. *Precision-Confidence Curve and Precision curve Model YOLOv8 Original*

$$Precision = \frac{TP}{TP + FP} = \frac{850}{850 + 50} = 0.94 \quad (9)$$

c. Precision-Recall Curve and Recall Curve

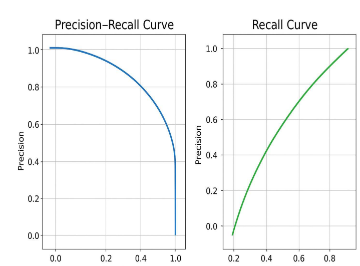

Figure 16. *Precision-Recall Curve and Recall Curve Original YOLOv8 Model*

Jurnal Teknik Informatika Vol. 19 No. 1, April 2026 (85-96)
ISSN: p-ISSN 1979-9160 (Print)| e-ISSN 2549-7901 (Online)
DOI: https://doi.org/10.15408/jti.v19i1.46941

$$Recall = \frac{TP}{TP + FN} = \frac{850}{850 + 150} = 0.85 \quad (10)$$

d. F1-Score

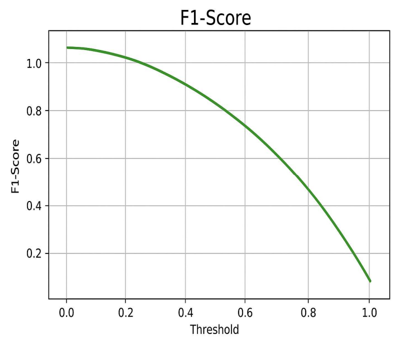

Figure 17. *F1-Score Model YOLOv8 Original*

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precsion + Recall} = 2 \times \frac{0.94 \times 0.85}{0.94 + 0.85} = 0.89 \quad (11)$$

e. Mean Average Precision (mAP@0.5)

$$mAP = AP \approx Precision$$
$$mAP = AP \approx 0.94 \times 0.85$$
$$mAP = AP \approx 0.799 \quad (12)$$

So, mAP value ≈ 79.9%

f. Accuracy

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{850 + 900}{850 + 900 + 50 + 150} = 0.875 = 87.5 \quad (13)$$

g. Comparison chart of YOLOv8 + ResNet50 with YOLOv8 Original

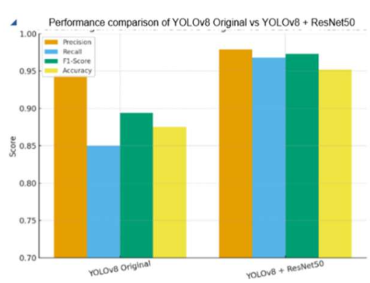

Figure 18. *Comparison Chart of YOLOv8 + ResNet50 with Original YOLOv8*

Based on the performance comparison graph between YOLOv8 Original and YOLOv8 + ResNet50 above, it can be seen that:

- YOLOv8 + ResNet50 shows improvement in all metrics (Precision, Recall, F1-Score, Accuracy).
- The most significant improvement was seen in Recall, indicating the model missed fewer objects.

CONCLUSION

In this study, the authors have successfully built a road damage object detection and classification model using the Yolv8 method with Resnet50 as the backbone. Based on testing conducted using the same dataset, the Yolv8 method with Resnet50 produced the highest accuracy value of 95.2%, an F1-Score value of 0.974, a Precision value of 0.979 and a Recall value of 0.968.

From the results above, it can be concluded that integrating YOLOv8 with ResNet50 as a backbone can improve detection accuracy and generalization capabilities, especially on datasets with high complexity. ResNet50, which has proven reliable in visual feature extraction, allows the model to be more sensitive to variations in road damage forms, although the consequence is increased computational requirements compared to YOLOv8's built-in backbone, such as CSPDarknet.

REFERENCES

[1] M. Hussain, “Yolov5, yolov8 and yolov10: The go-to detectors for real-time vision,”*arXiv Prepr. arXiv2407.02988*, 2024.

[2] RS Wijaya, S. Santonius, A. Wibisana, ER Jamzuri, and MAB Nugroho, “Comparative Study of YOLOv5, YOLOv7and YOLOv8 for Robust Outdoor Detection," J. Appl. Electr. Eng., vol. 8, no. 1, pp. 37–43, 2024.

[3] MA Ghofur and AB Ulum, “Comparative Analysis of the Performance of ResNet50 and EfficientNet-B0 Models in Road Damage Classification,”*J. Media Inform.*, vol. 6, no. 5, pp. 2504–2511, 2025.

[4] S. Sharma, S. Dhakal, and M. Bhavsar, “Transfer Learning for Wildlife Classification: Evaluating YOLOv8 against DenseNet, ResNet, and VGGNet on a Custom Dataset,”*arXiv Prepr.*

arXiv2408.00002, 2024.

[5] V. Alfiansyah, “Implementation of YOLOv8 for Detecting and Mapping Road Damage Points Using Google Street View (Case Study in Caturtunggal Village).” Gadjah Mada University, 2024.

[6] P. Singh, B. Likhitha, DG Reddy, D. Keerthana, BT Students, and AR Inspection, “Intelligent Road Monitoring : Advanced Damage And Pothole Detection With Yolov8,” no. 1, pp. 674–685, 2025.

[7] F. Wan, C. Sun, H. He, G. Lei, L. Xu, and T. Xiao, “YOLO-LRDD: a lightweight method for road damage detection based on improved YOLOv5s,”*EURASIP J. Adv. Signal Process.*, vol. 2022, no. 1, 2022, doi: 10.1186/s13634-022-00931-x.

[8] ERMBA Sucipto, RRM Putri, and BD Setiawan, “Development of a Pothole Detection System on Roads Using the Yolo Algorithm Based on Esp32-Cam,”*J. Development of Information Technology and Computer Science.*, vol. 9, no. 4, 2025.

[9] M. Surahmanto, S. Aras, M. Rifki Idhan Adhim, and P. Ussalama, “Pothole Detection Using the Yolov5 Algorithm,”*J. Digit. Bus. Inf. Technol.*, vol. 1, no. 1, pp. 1–8, 2024, doi: 10.23971/jobit.v1i1.198.

[10] LG Denaro and R. Lim, “Analysis of Asphalt Road Damage Detection using Deep Learning to Support Cost and Time Efficiency in Continuous Monitoring,”*J. Dimens. Ins. Prof.*, vol. 3, no. 1, pp. 16–25, 2025, doi: 10.9744/jdip.3.1.16-25.

[11] U. Satchithanantham, “Advanced pothole detection using neural network model-VGG16,”*Int. J. Commun. Inf. Technol.*, vol. 5, no. 2, pp. 11–16, 2024, doi: 10.33545/2707661x.2024.v5.i2a.86.

[12] J. Zeng and H. Zhong, “YOLOv8-PD: an improved road damage detection algorithm based on YOLOv8n model,”*Sci. Rep.*, vol. 14, no. 1, pp. 1–14, 2024, doi: 10.1038/s41598-024-62933-z.

[13] J. Terven, D.-M. Córdova-Esparza, and J.-A. Romero-González, “A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas,”*Mach. Learn. Knowl. Extr.*, vol. 5, no. 4, pp. 1680–1716, 2023.

[14] Y. Lin, T. Yu, and Z. Lin, “FTN-ResNet50: flexible transformer network model with ResNet50 for road crack detection,”*Evol. Syst.*, vol. 16, March. 2025, doi: 10.1007/s12530-025-09667-z.

[15] DBD Hanggoro, “Comparative Analysis of Deep Learning Architectures for Computer Vision Applications: A Literature Review Study,”*J. Comput. Technol. Inf. Sis. Inf.*, vol. 4, no. 2, pp. 1001–1008, 2025.

[16] A. Setiyadi, E. Utami, and D. Ariatmanto, “Analysis of the capability of the YOLOv8 algorithm in human object detection using the architectural modification method,”*J-SAKTI (Journal of Computer Science and Information).*, vol. 7, no. 2, pp. 891–901, 2023.

[17] MH Rais, A. Musnansyah, and H. Fakhrurroja, “Application of Yolo V8 Algorithm for Recognizing Motorcycle Riders Without Helmets in Traffic Violation Monitoring System,”*eProceedings Eng.*, vol. 12, no. 1, 2025.

[18] RG Wijanarko, AI Pradana, and D. Hartanti, “Implementation of Drone Detection Using YOLO (You Only Look Once),”*J. Faculty of Computer Science*, vol. 14, no. 2, pp. 437–442, 2024.

[19] I. Andi, M. Muchtar, and JY Sari, "Mask Detection Using the YOLO (You Only Look Once) Method,"*J. Media Inf. Technol.*, vol. 1, no. 1, pp. 1–12, 2024.

[20] ZS Hidayat, YA Wijaya, and R. Kurniawan, "Optimizing YOLOv8 for Autonomous Driving: Batch Size for Best Mean Average Precision (mAP),"*J. Tech. Inform.*, vol. 5, no. 4, pp. 1147–1153, 2024.