buildings

MDPI

Article

Automatic Detection of Urban Pavement Distress and Dropped Objects with a Comprehensive Dataset Collected via Smartphone

Lin Xu [1], Kaimin Fu [2], Tao Ma [3], Fanlong Tang [4] and Jianwei Fan [3,*]

[1] Jiangxi Transportation Engineering Group Company Ltd. Haitong Branch, Nanchang 330000, China; xulinjg@163.com
[2] Jiangxi Provincial Transportation Investment Group Co., Ltd., Nanchang 330025, China; twilight0001@163.com
[3] School of Transportation, Southeast University, Nanjing 211189, China; matao@seu.edu.cn
[4] College of Network and Communication Engineering, Jinling Institute of Technology, Nanjing 211169, China; tangfanlong@jit.edu.cn
* Correspondence: jianwei_fan@seu.edu.cn

Abstract: Pavement distress seriously affects the quality of pavement and reduces driving comfort and safety. The dropped objects from vehicles have increased the risks of traffic accidents. Therefore, automatic detection of urban pavement distress and dropped objects is an effective method to timely evaluate pavement condition. Firstly, this paper utilized a portable platform to collect pavement distress and dropped objects to establish a high-quality dataset. Six types of pavement distresses: transverse crack, longitudinal crack, alligator crack, oblique crack, potholes, and repair, and three types of dropped objects: plastic bottle, metal bottle, and tetra pak were included in this comprehensive dataset. Secondly, the real-time YOLO series detection models were used to classify and localize the pavement distresses and dropped objects. In addition, segmentation models W-segnet, U-Net, and SegNet were utilized to achieve pixel-level detection of pavement distress and dropped objects. The results show that YOLOv8 outperformed YOLOv5 and YOLOv7 with a MAP of 0.889. W-segnet showed an overall MIoU of 70.65% and 68.33% on the training set and test set, respectively, being superior to the comparison model and being able to achieve high-precision pixel-level segmentation. Finally, the trained models were performed on the holdout dataset for the generalization test. The proposed methods integrated the detection of urban pavement distress and dropped objects, which could significantly contribute to driving safety.

Keywords: pavement distress; dropped objects; object detection; semantic segmentation

check for updates

Citation: Xu, L.; Fu, K.; Ma, T.; Tang, F.; Fan, J. Automatic Detection of Urban Pavement Distress and Dropped Objects with a Comprehensive Dataset Collected via Smartphone. *Buildings* **2024**, *14*, 1546. https://doi.org/10.3390/buildings14061546

Academic Editor: Francesco Ascione

Received: 28 April 2024
Revised: 13 May 2024
Accepted: 19 May 2024
Published: 27 May 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Different types of urban pavement distress are continuously showing up on the surface and are requiring more and more effort in terms of identification and maintenance. In addition to impairing pavement performance, pavement deterioration also contributes to traffic accidents [1,2]. Consequently, the lifespan of pavements continually shortens, demanding more frequent maintenance to address escalating distress. Thus, it is imperative to promptly recognize and rectify these issues. Timely detection and repair are paramount, with real-time identification of pavement distress proving particularly worthwhile [3,4]. Once information on urban pavement distress is gathered, specific maintenance tasks can be promptly executed, ensuring pavements fulfill their designated functions within their intended service life.

Traditionally, pavement distress detection primarily has relied on manual inspections complemented by multi-functional pavement detection vehicles to assess distress types and corresponding damage levels [5]. However, manual detection methods are notably inefficient and subject to subjective evaluation of pavement distress. Furthermore, manual inspections necessitate traffic control measures, disrupting normal road usage and posing

safety hazards to detection personnel. The introduction of pavement detection vehicles has significantly boosted detection efficiency, enabling the rapid collection of pavement surface condition data [6]. Despite these advancements, limitations persist, as detection tasks cannot be consistently performed at fixed speeds or lanes due to traffic flow constraints, resulting in low detection frequencies and hindering real-time, repeated inspections at specific points [7–9].

Dropped objects from vehicles present another risk to driving safety. Dropped objects, such as rocks, bottles, and boxes, could distract drivers and interrupt traffic flow. Manual inspection of dropped objects is still a prevalent method to detect these risks. However, it is dangerous to detect and get rid of those dropped objects with manual work. Therefore, detecting dropped objects on the pavement and promptly addressing them is crucial for ensuring safety. The integrated detection of pavement distress and dropped objects could provide a more accurate and reliable solution. Computer-vision-based techniques have been introduced into pavement engineering to achieve automatic detection. Images acquired by different devices were used in deep-learning-based models to automatically detect pavement distress and dropped objects.

Deep learning algorithms encompass two primary categories: supervised and unsupervised learning [10]. Supervised learning necessitates a substantial dataset, while unsupervised learning achieves recognition through clustering algorithms. Traditionally, supervised learning has been favored for its capacity to deliver higher accuracy. Within deep learning algorithms, three fundamental tasks prevail: image classification, object detection, and semantic segmentation. Recent advancements in deep learning technology have seen researchers employ neural network models capable of end-to-end target recognition directly from input images, without manual intervention. Consequently, this approach has found extensive applications in pavement crack detection [11,12]. The methodology for identifying pavement cracks using deep learning typically progresses through three stages: Initially, CNN sliding window technology is primarily employed for crack classification [13]. Subsequently, anchors are devised to pinpoint pavement cracks in images [14]. Finally, pixel-level semantic segmentation is employed to precisely extract pavement crack morphology [15,16]. Input data can vary in format, including grayscale, color, depth, point cloud, and infrared images. Similarly, outputs can range from recognition and detection results at different levels, such as image-level, grid-cell level, region level, and pixel level. Ultimately, deep-learning-oriented models facilitate the identification, localization, segmentation, and measurement of pavement distress and dropped objects [17–19].

Although the aforementioned method can effectively classify pavement distress, it lacks the capability to analyze the characteristics of pavement distress based on classification using the trained model. To address this limitation, a fusion model has been developed, integrating Faster R-CNN and morphology to classify, localize, and measure pavement cracks. Faster R-CNN is employed for the classification of various types of pavement distress [20], simultaneously providing distress coordinates for localization [21,22]. The methodology involves using CNN sliding windows to extract the pavement crack skeleton, followed by digital morphology operations to extract crack geometric features, thereby facilitating the evaluation of pavement damage degree. Building upon the two-stage pavement crack detection and segmentation algorithm, the deep learning model YOLO is adopted for crack classification in the first stage, while crack extraction in the second stage is based on the enhanced U-Net [23,24]. In contrast to current single-step classification or crack segmentation methods, the two-stage pavement crack detection model demonstrates superior accuracy, enabling swift classification of pavement cracks and laying a foundation for integrated pavement distress detection [25,26]. However, existing automatic recognition algorithms exhibit limited generality and fail to deliver consistent performance across diverse pavement conditions, while the models' parameters are excessively large, rendering them unsuitable for offline deployment [27,28].

The current automatic identification and evaluation technology of urban pavement distress makes it difficult to meet the requirements of real-time processing and analysis.

The generalization performance of the recognition algorithm under different pavement structures and conditions is poor. The current detection equipment is expensive and heavy, and it cannot be applied to large-scale detection. There are several issues, as follows:

1. Lack of a lightweight, multi-dimensional, and high-frequency pavement distress detection platform. Pavement distress detection is carried out by manual inspection combined with a multi-functional pavement detection vehicle. However, the integrated equipment for highway detection and detection is expensive. As for the multi-functional road automated detection vehicle, it is installed with three-dimensional scanning, laser sensors, and other expensive components. Moreover, the full road surface distress data need to be collected several times by lane, and it cannot realize portable, lightweight, and high-frequency automatic road surface detection.
2. The recognition algorithm has poor generalization ability. Current algorithms cannot be used for distress recognition of various pavement conditions. The types of pavement distress are single, the model generalization ability is poor, and the data quantity acquisition of data-driven deep learning models is difficult.
3. Lack of automatic pavement condition evaluation methods. The evaluation of pavement conditions should include seven steps: identification, localization, segmentation, extraction, measurement, statistics, and evaluation of pavement conditions. However, the current research only contains part of the content.

Therefore, the objective of this research is to develop a lightweight platform to collect pavement distress and dropped objects to establish a comprehensive dataset. YOLO-based algorithms were used to classify and localize urban pavement distress and dropped objects. The W-segnet model was applied to segment pavement distress and dropped objects to provide information for evaluation.

2. Establishment of the Dataset

2.1. Data Collection Method

To establish a cost-effective and lightweight platform to collect pavement distress and dropped object images, a smartphone was used to collect images. As shown in Figure 1, the hand-held gimbal stabilizer mounted with a smartphone was utilized to record videos along the pavement to continually collect images. To establish a dataset with various backgrounds, images were captured under different pavement conditions, as well as on rainy and sunny days, to cover commonly seen backgrounds in real life. Six types of pavement distress and three types of dropped objects were collected for model training and testing. A total of 2000 images of pavement distress were collected, and a total of 500 images of dropped objects were captured for model training. As shown in Figure 1, videos were recorded at the speed of 30 km/h along the road with this lightweight platform. Videos were converted into image sequences to establish the dataset.



(**a**) (**b**)

Figure 1. A lightweight platform for data collection. (**a**) A smartphone with a gimbal. (**b**) Collecting pavement images.

2.2. Image Annotation

For the supervised learning models, the ground truth of targets is the foundation of high accuracy. Therefore, collected images were annotated to provide information for deep-learning-based models. There are a total of six pavement distresses classified in this research. As shown in Figure 2, the images were transverse crack (TC), longitudinal crack (LC), oblique crack (OC), alligator crack (AC), potholes, and repairs (sealed crack and patch), which frequently occur on the pavement.

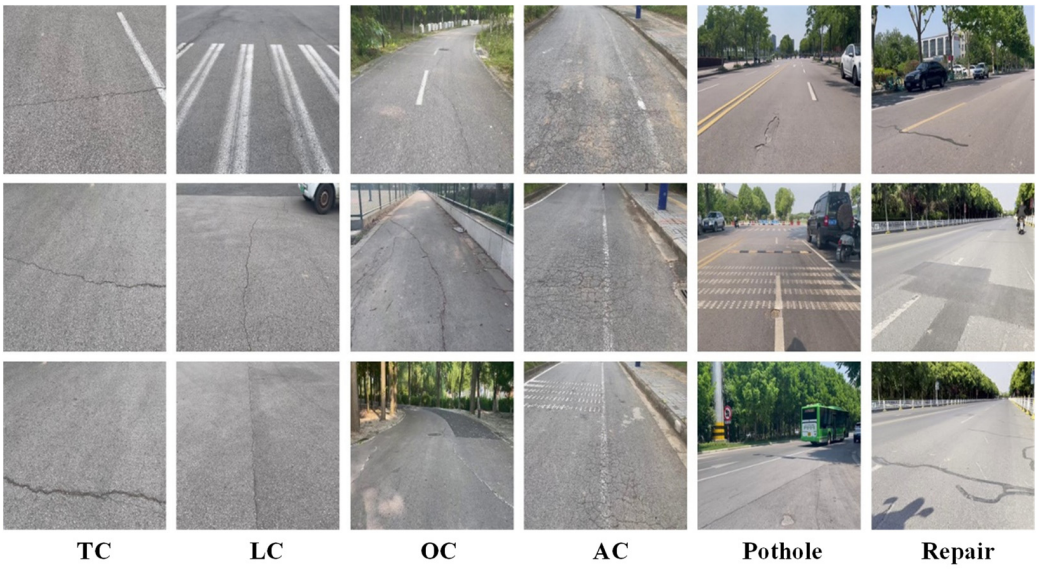


Figure 2. Pavement distress images.

Figure 3 presents three types of typical dropped objects on pavement, containing plastic bottles, tetra pak, and metal bottles. Labelme and LabelImg were used to label pavement distress and dropped objects at the pixel and region levels to automatically detect pavement distress and dropped objects.



Figure 3. Dropped objects on pavement.

Segmentation is more difficult than object detection since it classifies each pixel into one category. Therefore, the segmentation was based on object detection, and it can improve

accuracy. Pavement transverse crack, longitudinal crack, oblique crack, and alligator crack were classified as cracks in segmentation to simplify the pixel-level detection as we obtained the detailed classification from region-level detection. Moreover, dropped objects were classified as one class as well to improve the precision of pixel-level detection. One-hot was utilized to encode the targets where [1, 0, 0, 0], [0, 1, 0, 0], [0, 0, 1, 0], and [0, 0, 0, 1] represent pavement crack, pothole, repair, dropped objects, respectively.

3. Methodology

3.1. Models for Region-Level Detection

The YOLO series models are state-of-the-art algorithms nowadays for object detection. The YOLO series models are one-stage models that balance the accuracy and training computation. Therefore, YOLOs have been popular for object detection and achieve end-to-end detection, thereby improving the detection efficiency. YOLOv5, YOLOv7, and YOLOv8 were used in this research to achieve region-level detection of pavement distress and dropped objects. YOLOv8 is stable compared with the other two models and it includes five scales named n, s, m, l, and x for different datasets. YOLOv8s is suitable for small datasets, and it made a tradeoff between accuracy and inference. Figure 4 presents the specific structure of YOLOv8 used for region-level detection. YOLOv8 is composed of three parts: backbone, neck, and head. The backbone is responsible for feature extraction via convolutional neural network to obtain abstracted feature maps for the model. The prediction head can detect pavement distress and dropped objects with three scales: small scale (256), medium scale (512), and large scale (1024), which is suitable for detecting objects of different sizes, especially for pavement distress and dropped objects.

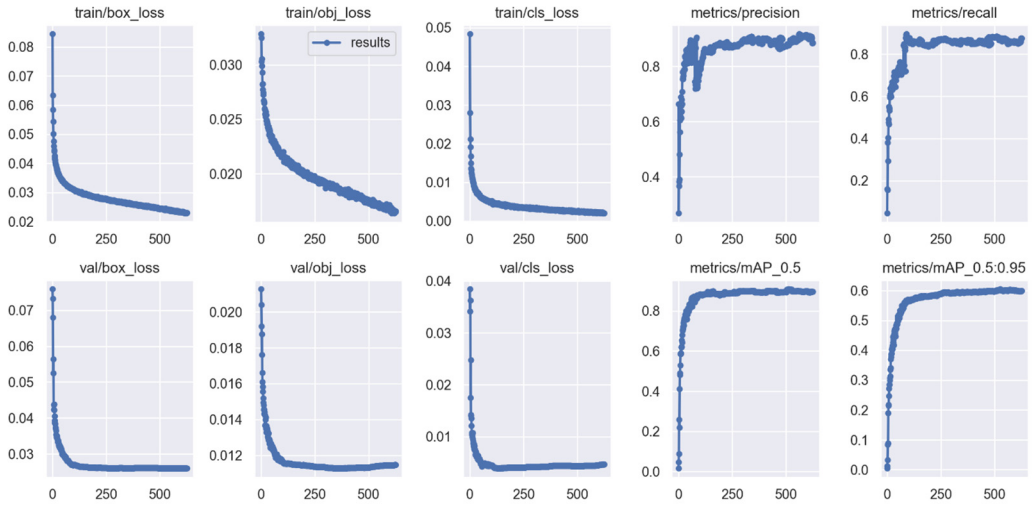


Figure 4. Training stage for YOLOv8.

3.2. Models for Pixel-Level Detection

To obtain the geometric information of detected pavement distress and dropped objects, segmentation models were adopted to achieve pixel-level detection. Segmentation models assign each pixel with a specific value to extract targets and output an image with a background in black. The mainstream encoder–decoder-based segmentation models were used and compared to select the most suitable model for both pavement distress and dropped objects.

U-Net, SegNet, and W-segnet [2] were compared for the segmentation to extract the pixel-level information from the targets. U-Net and SegNet are the original structures based on the symmetric architecture to extract the features and then restore them into segmentation masks. The W-segnet is inspired by feature fusion, and it utilizes two symmetric encoder–decoder structures to better fuse the features of pavement distresses, thereby improving the segmentation performance.

3.3. Metrics

3.3.1. Loss Function for Region-Level Detection

The target detection model contains three types of losses, which can be calculated according to Equation (1):

$$Loss = loss_{box} + loss_{confidence} + loss_{cls} \tag{1}$$

where L_{box} is the difference between the real anchor and the predicted value; $L_{confidence}$ is the value of confidence in the predicted result of the actual existing box compared with 1, and the value of maximum IOU of the actual non-existent box compared with 0; and L_{cls} are the boxes that exist, where the kind of predicted results are compared with the actual results.

The regression loss consists of two items: one is the loss of center point coordinates, and the other is the loss of width and height, which can be calculated according to Equation (2):

$$L_{box} = \lambda_{box} \sum_{i=0}^{N_1 \times N_1} \sum_{j=0}^{3} 1_{ij}^{obj} \left[(t_x - t_x^*)^2 + \left(t_y - t_y^* \right)^2 \right] + \lambda_{box} \sum_{i=0}^{N_1 \times N_1} \sum_{j=0}^{3} 1_{ij}^{obj} \left[(t_w - t_w^*)^2 + (t_h - t_h^*)^2 \right] \tag{2}$$

I_{ij}^{obj} indicates whether the j-th anchor in the i-th grid contains an object. If it does, the value is 1; otherwise, it is 0. I_{ij}^{noobj} indicates that the j-th anchor in the i-th grid does not contain an object. If it does not, the value is 1; otherwise, it is 0.

The loss of confidence can be calculated by using binary cross-entropy, according to Equation (3):

$$\begin{aligned} L_{confidence} = & -\sum_{i=0}^{N \times N} \sum_{j=0}^{3} I_{ij}^{obj} \left[\overset{*}{C}_i log(C_i) + \left(1 - \overset{*}{C}_i \right) log(1 - C_i) \right] \\ & -\lambda_{noobj} \sum_{i=0}^{N \times N} \sum_{j=0}^{3} I_{ij}^{noobj} \left[\overset{*}{C}_i log(C_i) + \left(1 - \overset{*}{C}_i \right) log(1 - C_i) \right] \end{aligned} \tag{3}$$

The classification loss can be calculated by using binary cross-entropy, according to Equation (4):

$$L_{cls} = -\sum_{i=0}^{N \times N} I_{ij}^{obj} \sum_{c \in \text{ classes}} \left(\left[\overset{*}{P}_i log(P_i) + \left(1 - \overset{*}{P}_i \right) log(1 - P_i) \right] \right. \tag{4}$$

where P_i can be calculated by using the logistic function.

3.3.2. Loss Function for Pixel-Level Detection

The dice loss function was used for training, calculated by Equation (5):

$$DiceLoss = 1 - \frac{2 \sum_1^N y_i^* y_i + \varepsilon}{\sum_1^N y_i^* + \sum_1^N y_i + \varepsilon} \tag{5}$$

where N is the number of pixels in images, y_i^* is the value of ground truth, and y_i is the predicted values by the model. ε is a smooth constant to avoid division by zero.

3.3.3. Performance Evaluation

Performance evaluation was illustrated with the results of the confusion matrix, as shown in Table 1.

Table 1. Confusion matrix.

True Label	Prediction Results	
	Positive	**Negative**
Positive	TP (true positive)	FN (false negative)
Negative	FP (false positive)	TN (true negative)

For the detection of pavement distress and dropped objects, the images with targets (distress or dropped objects) are positive and the images without targets are negative. When the performance of the model is poor, false detections or missed samples can occur, detected as false positives or false negatives. The precision and recall of the model can be calculated based on *TP*, *FP*, and *FN*, according to

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

$$Recall = \frac{TP}{TP + \mathrm{FN}} \tag{7}$$

$$AP = \int p(r)dr \tag{8}$$

$$MAP = \frac{1}{n}\sum_{i=1}^{n} AP \tag{9}$$

The precision and recall of the model can be calculated based on *TP*, *FP*, and *FN*, according to

$$MIoU = \frac{1}{k+1}\sum_{i=0}^{k} \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}} \tag{10}$$

where p_{ii} is the number of *TP*; p_{ij} is the number of *FP*; and p_{ji} is the number of *FN*. k is the classification category.

3.3.4. Experimental Settings

The dataset was divided into a train and a test set for performance evaluation. The proportion of train and test was 8:2, where 2000 images of pavement distress and dropped objects were used for training and 500 images were used for testing. Cross-validation was used in training to adjust hyperparameters automatically to improve the performance of models. Transferring learning, as a commonly used trick, was used to initialize the parameters to help converge. The pre-trained model was trained on the COCO dataset, which includes many types of objects in real life, and they are helpful in improving the detection accuracy of spoiled loads. The initial learning rate was 1×10^{-4}, and the epochs were 700 for all models. All images were resized to 512 by 512 pixels to reduce computation.

4. Results and Discussion

4.1. Performance of Region-Level Detection

The detection performance was evaluated on the test set. From Table 2, compared with YOLOv5 and YOLOv7, it can be seen that YOLOv8 had the highest recognition accuracy. The MAP for six types of pavement distress and three types of dropped objects was 0.889, outperforming YOLOv5 and YOLOv7. YOLOv5 presented higher accuracy for oblique crack and alligator crack. The backbone for YOLOv5, YOLOv7, and YOLOv8 was almost the same, and they presented similar performance to pavement distress and dropped objects. Many tricks were used for YOLOv7 and YOLOv8, and the detection performance was improved based on this. Two of the latest pavement distress detection methods were selected for comparison [29,30]. These methods differed in pavement distress classification and did not include dropped object detection.

Table 2. Region-level detection of pavement distress and dropped objects.

Model	AP									MAP
	Pavement Distress						Dropped Objects			
	TC	LC	OC	AC	Pothole	Repair	Plastic Bottle	Metal Bottle	Tetra Pak	
YOLOv5	0.852	0.812	0.764	0.811	0.964	0.833	0.975	0.946	0.952	0.879
YOLOv7	0.824	0.802	0.702	0.765	0.987	0.865	0.913	0.963	0.927	0.861
YOLOv8	0.877	0.842	0.724	0.799	0.981	0.868	0.968	0.984	0.962	0.889
CrackYOLO	0.682	0.677	0.749	-	-	-	-	-	-	0.703
Paper [29]	0.6	0.5	0.7	0.8	0.8	0.8	-	-	-	0.7

Therefore, to intuitively describe the training stages, parameter changes in training are plotted in Figure 4. The loss of train and validation was not significantly different, and it means that there was no overfitting during the training process. The precision and recall began to stabilize at around 100 epochs, and it showed a slowly increasing trend in the following epochs. It means that the transferring contributed a lot to the initialization of the model since YOLOv8 presented a higher precision at the early training stages.

Additionally, YOLOv8 used the Mosaic data augmentation, as shown in Figure 5. Four pavement distress and dropped object images were combined in the training to improve the diversity of the training batch and reduce the difficulties of learning features from different classes, especially for pavement dropped objects with different colors and textures.

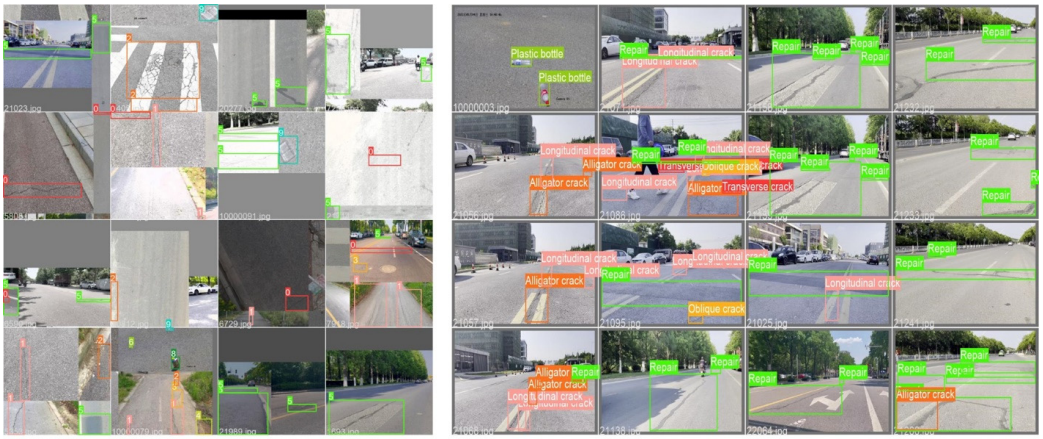


Figure 5. Data augmentation used in the training of YOLOv8.

Figure 6 depicts the detection results of pavement distress with YOLOv5, YOLOv7, and YOLOv8. Different crack shapes and orientations are shown in these images. In the first row of Figure 6, all models were able to detect the transverse crack accurately while YOLOv5 had the highest confidence for transverse cracks with a probability of 0.94. However, YOLOv7 and YOLOv8 were successful in localizing the pavement transverse crack with a more proper bounding box to the real case. In the second row, YOLOv5 still presented the highest confidence, while YOLOv7 and YOLOv8 showed the more accurate location of cracks. In the third row, the pavement cracks were more complicated than those simple scenarios where only linear cracks existed. It is difficult to detect the long cracks since models will detect the pavement cracks in different segments. There are two longitudinal cracks and one oblique crack shown in the third row of Figure 6. However, models failed to detect all the longitudinal cracks. One longitudinal crack was divided into two parts, while one segment was classified as a longitudinal crack, and the other was recognized as an oblique crack. Therefore, more diverse pavement cracks should be included for model training to improve the detection performance.

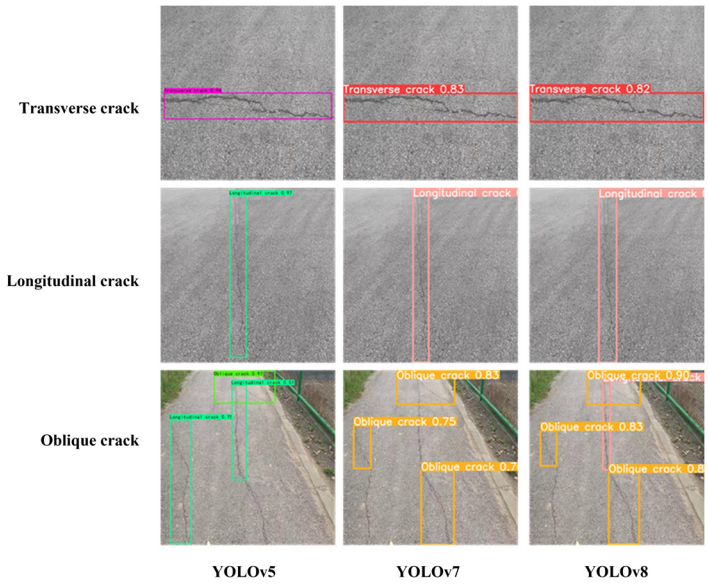


Figure 6. Performance comparison with different models.

Since YOLOv8 had the highest overall detection performance over pavement distress and dropped objects, Figure 7 presents the region-level detection results. Well-trained YOLOv8 was robust in different scenarios with various pavement conditions. The longitudinal crack that occurred on the pavement marking was detected accurately. The most challenging part was that there were some overlapped pavement distresses, as shown in Figure 7. Alligator cracks occurred around the repaired surface area, and it made the object not that distinguishable. YOLOv8 presents an impressive ability to detect urban pavement distress under these scenarios.

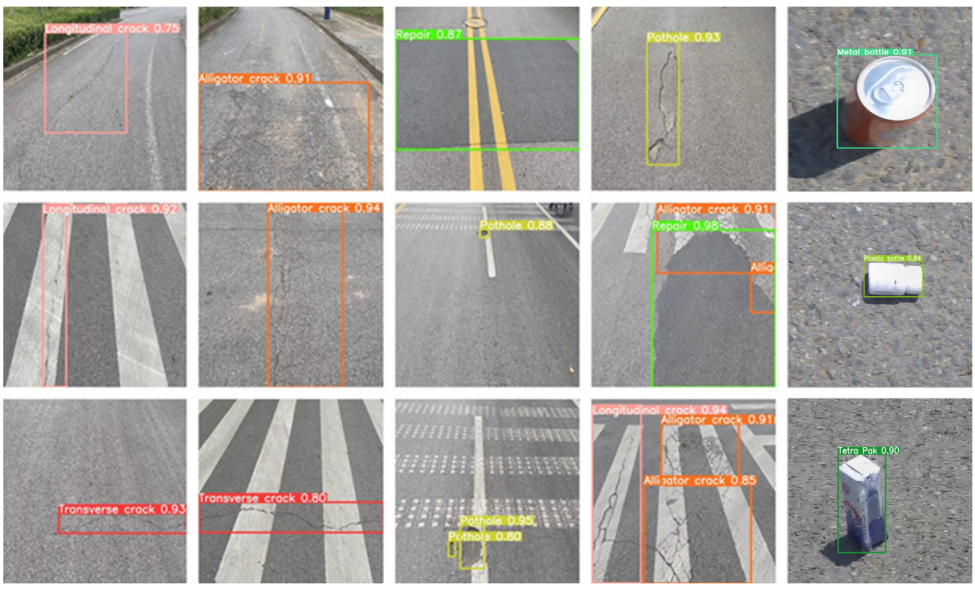


Figure 7. Region-level detection of urban pavement distress and dropped objects.

4.2. Performance of Region-Level Detection

Table 3 presents the pixel-level detection of pavement distress and dropped objects based on segmentation models. It is noted that the metrics were calculated based on all types of pavement distress and dropped objects. W-segnet outperformed U-Net and SegNet in terms of precision, recall, F1, and MIoU. In addition, W-segnet used the VGG16,

which has fewer parameters compared to ResNet50, thereby balancing the accuracy and the training time.

Figure 8 depicts some pixel-level detection samples for pavement cracks and dropped objects. W-segnet can segment fine pavement cracks well compared to U-Net and SegNet. However, these models still cannot produce perfect segmentation results compared to the GT. The first row and second row of Figure 9 present some complicated pavement alligator cracks existing at many intersections among cracks, thereby reducing the detection accuracy. For the third row, three models were able to obtain the main crack skeleton for segmentation, while some detailed information was missing. W-segnet still provided slightly better performance for fine cracks, as shown in Figure 8.

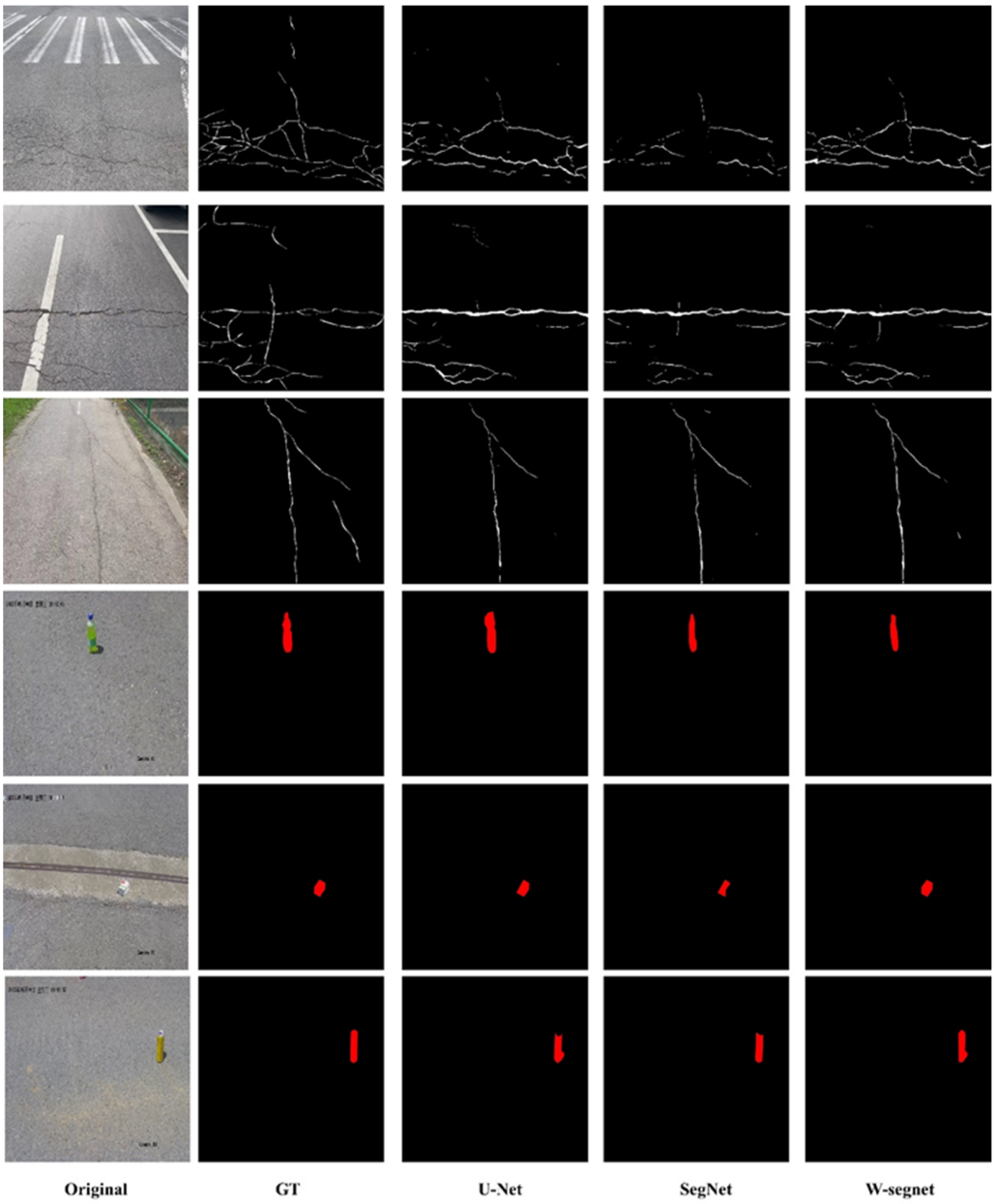


Figure 8. Pixel-level detection of pavement distress and dropped objects.



Figure 9. Generalization test for region-level detection of urban pavement distress.

Table 3. Pixel-level detection results of pavement distress and dropped objects.

Model	Backbone	Precision (%)		Recall (%)		F1 (%)		MIoU (%)	
		Train	Test	Train	Test	Train	Test	Train	Test
W-segnet	VGG16	76.85	75.43	81.02	83.55	78.88	79.28	70.65	68.33
U-Net	VGG16	71.04	72.64	72.56	70.68	71.79	71.65	68.72	65.73
SegNet	ResNet50	63.75	65.62	72.89	75.23	68.01	70.10	60.52	61.73

The last three rows in Figure 8 present the segmentation of dropped objects. The red indicates the dropped objects, and the background is black. It was not easy to obtain the exact shape for the segmented masks compared to the ground truth, especially for plastic and metal bottles. W-segnet was able to segment a more accurate edge for tetra pak with the straight edge, while it was difficult for the smooth edge for plastic and metal bottles. Therefore, more diverse pavement dropped objects included in the dataset could contribute to performance improvement.

4.3. Generalization Performance Test

To evaluate the generalization performance of the YOLOv8 and W-segnet, holdout images from a video were used to achieve region-level and pixel-level pavement distress detection, as shown in Figures 9 and 10. It is noted that the region-level detection was more accurate compared with the pixel-level detection. The scenario was different from

the collected dataset used for training, and it included a complicated background. The well-trained YOLOv8 still presented impressive region-level results for pavement distresses. The holdout images include the common scene in our daily life, and it demonstrates that the well-trained YOLOv8 based on our dataset provides a good generalization when applied to other datasets. As shown in Figure 10, even with the influence of shadows and uneven illumination, W-segnet was able to produce comparably high segmentation accuracy. It is worth noting that both region-level detection and pixel-level detection met the real-time detection of urban pavement distress and dropped objects, with region-level detection achieving 78 frames per second and pixel-level detection achieving 53 frames per second.



Figure 10. Generalization test for pixel-level detection of urban pavement distress.

Due to the susceptibility of hardware equipment to moisture, the detection process is typically conducted under clear weather conditions. However, factors like trees along urban roadsides or overcast weather can introduce shadows and water stains on the urban pavement, as depicted in Figure 9. Despite such challenges, the detection and segmentation algorithm proposed in this paper effectively identified images without generating false positives. Yet, there exists a concern regarding overlapping areas among different disease detection boxes within the image. This overlap can potentially inflate results during the calculation of the urban pavement condition index. To address this issue, a general non-maximum suppression technique must be applied to all detection boxes within the same image. This involves consolidating all detection boxes to obtain a comprehensive view for accurate index calculation. One notable limitation of the algorithm is its inability to precisely delineate the detection range of the urban pavement in sections where road markings are absent, which may impact the scoring accuracy of the urban pavement condition index.

5. Conclusions

Pavement distress and dropped objects significantly impact pavement quality, reducing driving comfort and compromising safety. Automatic detection of pavement distress and dropped objects is an effective method to reduce risks and save money. This research obtained a high-quality pavement distress and dropped objects dataset, establishing a cost-effective collection platform. The well-established dataset laid the solid foundation for the detection of pavement distress and dropped objects based on deep learning. The YOLO series object detection models were used to realize the region-level classification and localization. Moreover, W-segnet was adopted to realize pixel-level recognition of pavement distress and dropped objects to obtain the geometric information for evaluation. The main findings of this study are listed as follows:

1. A multi-scene and multi-category pavement distress and dropped objects dataset was established with a cost-effective method. The hand-held gimbal stabilizer mounted with a smartphone was developed as a lightweight platform for data collection. A total of 2000 pavement distress images and 500 dropped objects images were collected for training and testing.
2. Three YOLO series models were compared to select the most suitable one-stage detection model for region-level detection. YOLOv8 outperformed YOLOv5 and YOLOv7 in terms of all evaluation metrics with an overall MAP of 0.889. YOLOv8

presented higher precision for longitudinal and transverse cracks compared to oblique and alligator cracks. The overall precision for dropped objects was over 0.95, and it succeeded in detecting dropped objects with different sizes. Therefore, YOLOv8 is suitable for both pavement distress and dropped object detection.

3. Encoder–decoder-based segmentation models were compared for segmenting pavement distress and dropped objects. A multi-scale feature fusion model: W-segnet showed an overall MIoU of 70.65% and 68.33% on the training set and test set, respectively. W-segnet had a better segmentation performance for the detection of tetra pak with the straight edge, while it showed inferior performance on plastic and metal bottles. W-segnet is more suitable for fine cracks compared with U-Net and SegNet due to the feature fusion.
4. Well-trained YOLOv8 and W-segnet were performed on a holdout dataset to evaluate the generalization of the models. With a more complicated background, YOLOv8 can still have better region-level detection results, while W-segnet indicated a slightly inferior segmentation performance. Further, the trained models demonstrated the generalization ability to other datasets.
5. In the presence of water stains and shadows in the real environment of urban roadways, the algorithm used in this paper can still effectively identify urban pavement distress and dropped objects accurately. In the process of calculating the urban road condition index, it is necessary to consider the issue of duplicate calculation of detection boxes and establish regulations for urban pavement boundaries.

Author Contributions: Conceptualization, T.M.; methodology, L.X.; validation, K.F.; investigation, L.X. and K.F.; resources, T.M.; data curation, L.X.; writing—original draft preparation, L.X.; writing—review and editing, F.T.; visualization, J.F.; supervision, J.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China (grant number 52378445) and the Postdoctoral Fellowship Program of CPSF (grant number GZC20230432).

Data Availability Statement: Please contact the corresponding author to request access to the data mentioned in the article, but note that it cannot be used for commercial activities.

Conflicts of Interest: Author Lin Xu and Kaimin Fu were employed by the company Jiangxi Transportation Engineering Group Co. Ltd., and Jiangxi Provincial Transportation Investment Group Co., Ltd., respectively. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Cha, Y.; Choi, W.; Buyukozturk, O. Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks. *Comput. Aided Civ. Infrastruct. Eng.* **2017**, *32*, 361–378. [CrossRef]
2. Zhong, J.; Zhu, J.; Huyan, J.; Ma, T.; Zhang, W. Multi-scale feature fusion network for pixel-level pavement distress detection. *Autom. Constr.* **2022**, *141*, 104436. [CrossRef]
3. Zhong, J.; Huyan, J.; Zhang, W.; Cheng, H.; Zhang, J.; Tong, Z.; Jiang, X.; Huang, B. A deeper generative adversarial network for grooved cement concrete pavement crack detection. *Eng. Appl. Artif. Intell.* **2023**, *119*, 105808. [CrossRef]
4. Mei, Q.; Gül, M.; Azim, M. Densely connected deep neural network considering connectivity of pixels for automatic crack detection. *Autom. Constr.* **2019**, *110*, 103018. [CrossRef]
5. Yang, F.; Zhang, L.; Yu, S.; Prokhorov, D.; Mei, X.; Ling, H. Feature Pyramid and Hierarchical Boosting Network for Pavement Crack Detection. *arXiv* **2019**, arXiv:1901.06340. [CrossRef]
6. Shi, Y.; Cui, L.; Qi, Z.; Meng, F.; Chen, Z. Automatic Road Crack Detection Using Random Structured Forests. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 3434–3445. [CrossRef]
7. Zhong, J.; Zhang, M.; Ma, Y.; Xiao, R.; Cheng, G.; Huang, B. A Multitask Fusion Network for Region-Level and Pixel-Level Pavement distress detection. *J. Transp. Eng. Part B Pavements* **2024**, *150*, 04024002. [CrossRef]
8. Oliveira, H.; Correia, P. CrackIT—An image processing toolbox for crack detection and characterization. In Proceedings of the IEEE International Conference on Image Processing, Québec City, QC, Canada, 27–30 September 2015.
9. Maeda, H.; Sekimoto, Y.; Seto, T.; Kashiyama, T.; Omata, H. Road Damage Detection and Classification Using Deep Neural Networks with Smartphone Images: Road damage detection and classification. *Comput. Aided Civ. Infrastruct. Eng.* **2018**, *33*, 1127–1141. [CrossRef]

10. Yang, X.; Li, H.; Yu, Y.; Luo, X.; Huang, T.; Yang, X. Automatic Pixel-level Crack Detection and Measurement Using Fully Convolutional Network. *Comput. Aided Civ. Infrastruct. Eng.* **2018**, *33*, 1090–1109. [CrossRef]
11. Zhu, J.; Zhong, J.; Ma, T.; Huang, X.; Zhang, W.; Zhou, Y. Pavement distress detection using convolutional neural networks with images captured via UAV. *Autom. Constr.* **2022**, *133*, 103391. [CrossRef]
12. Silva, L.A.; Blas, H.S.S.; García, D.P.; Mendes, A.S.; González, G.V. An Architectural Multi-Agent System for a Pavement Monitoring System with Pothole Recognition in UAV Images. *Sensors* **2020**, *20*, 6205. [CrossRef] [PubMed]
13. Li, Y.; Liu, C.; Shen, Y.; Cao, J.; Yu, S.; Du, Y. RoadID: A Dedicated Deep Convolutional Neural Network for Multipavement Distress Detection. *J. Transp. Eng. Part B Pavements* **2021**, *147*, 04021057. [CrossRef]
14. Mei, Q.; Gül, M. A cost-effective solution for pavement crack inspection using cameras and deep neural networks. *Constr. Build. Mater.* **2020**, *256*, 119397. [CrossRef]
15. Yang, H.D.; Huyan, J.; Ma, T.; Tong, Z.; Han, C.J.; Xie, T.Y. Novel Computer Tomography image enhancement deep neural networks for asphalt mixtures. *Constr. Build. Mater.* **2022**, *352*, 129067. [CrossRef]
16. Liu, J.; Yang, X.; Lau, S.; Wang, X.; Luo, S.; Lee, V.C.; Ding, L. Automated Pavement Crack Detection and Segmentation based on Two-step Convolutional Neural Network. *Comput. Aided Civ. Infrastruct. Eng.* **2020**, *35*, 1291–1305. [CrossRef]
17. Peng, Y.; Yang, H.D. Aggregate boundary recognition of asphalt mixture CT images based on convolutional neural networks. *Road Mater. Pavement Des.* **2023**, *25*, 1127–1143. [CrossRef]
18. Sattar, D.; Thomas, R.; Marc, M. Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete. *Constr. Build. Mater.* **2018**, *186*, 1031–1045.
19. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, Nevada, USA, 27–30 June 2016; pp. 779–788.
20. Cha, Y.J.; Choi, W.; Suh, G.; Mahmoudkhani, S.; Büyüköztürk, O. Autonomous Structural Visual Inspection Using Region-Based Deep Learning for Detecting Multiple Damage Types. *Comput. Aided Civ. Infrastruct. Eng.* **2018**, *33*, 731–747. [CrossRef]
21. Song, L.; Wang, X. Faster region convolutional neural network for automated pavement distress detection. *Road Mater. Pavement Des.* **2021**, *22*, 23–41. [CrossRef]
22. Du, Y.; Pan, N.; Xu, Z.; Deng, F.; Shen, Y.; Kang, H. Pavement distress detection and classification based on YOLO network. *Int. J. Pavement Eng.* **2021**, *22*, 1659–1672. [CrossRef]
23. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
24. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
25. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
26. Shi, J.; Dang, J.; Cui, M.; Zuo, R.; Shimizu, K.; Tsunoda, A.; Suzuki, Y. Improvement of Damage Segmentation Based on Pixel-Level Data Balance Using VGG-Unet. *Appl. Sci.* **2021**, *11*, 518. [CrossRef]
27. Liu, Z.; Cao, Y.; Wang, Y.; Wang, W. Computer vision-based concrete crack detection using U-net fully convolutional networks. *Autom. Constr.* **2019**, *104*, 129–139. [CrossRef]
28. Ning, Z.; Wang, H.; Li, S.; Xu, Z. YOLOv7-RDD: A Lightweight Efficient Pavement Distress Detection Model. *IEEE Trans. Intell. Transp. Syst.* 2024. [CrossRef]
29. Li, Y.; Sun, S.; Song, W.; Zhang, J.; Teng, Q. CrackYOLO: Rural Pavement Distress Detection Model with Complex Scenarios. *Electronics* **2024**, *13*, 312. [CrossRef]
30. Guerrieri, M.; Parla, G.; Khanmohamadi, M.; Neduzha, L. Asphalt Pavement Damage Detection through Deep Learning Technique and Cost-Effective Equipment: A Case Study in Urban Roads Crossed by Tramway Lines. *Infrastructures* **2024**, *9*, 34. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.