

## Article

# A Pavement Crack Detection and Evaluation Framework for a UAV Inspection System Based on Deep Learning

Xinbao Chen , Chang Liu <sup>\*</sup>, Long Chen, Xiaodong Zhu, Yaohui Zhang and Chenxi Wang

School of Earth Sciences and Spatial Information Engineering, Hunan University of Sciences and Technology, Xiangtan 411201, China; 1901080111@mail.hnust.edu.cn (L.C.); xiaodong@mail.hnust.edu.cn (X.Z.); eminem@mail.hnust.edu.cn (Y.Z.); 2101080215@mail.hnust.edu.cn (C.W.)

<sup>\*</sup> Correspondence: xchen@hnust.edu.cn (X.C.); liuchang@mail.hnust.edu.cn (C.L.); Tel.: +86-18670925719 (X.C.); +86-18845160205 (C.L.)

**Abstract:** Existing studies often lack a systematic solution for an Unmanned Aerial Vehicles (UAV) inspection system, which hinders their widespread application in crack detection. To enhance its substantial practicality, this study proposes a formal and systematic framework for UAV inspection systems, specifically designed for automatic crack detection and pavement distress evaluation. The framework integrates UAV data acquisition, deep-learning-based crack identification, and road damage assessment in a comprehensive and orderly manner. Firstly, a flight control strategy is presented, and road crack data are collected using DJI Mini 2 UAV imagery, establishing high-quality UAV crack image datasets with ground truth information. Secondly, a validation and comparison study is conducted to enhance the automatic crack detection capability and provide an appropriate deployment scheme for UAV inspection systems. This study develops automatic crack detection models based on mainstream deep learning algorithms (namely, Faster-RCNN, YOLOv5s, YOLOv7-tiny, and YOLOv8s) in urban road scenarios. The results demonstrate that the Faster-RCNN algorithm achieves the highest accuracy and is suitable for the online data collection of UAV and offline inspection at work stations. Meanwhile, the YOLO models, while slightly lower in accuracy, are the fastest algorithms and are suitable for the lightweight deployment of UAV with online collection and real-time inspection. Quantitative measurement methods for road cracks are presented to assess road damage, which will enhance the application of UAV inspection systems and provide factual evidence for the maintenance decisions made by road authorities.

**Keywords:** road cracks; UAV; deep learning; target detection; road damage evaluation; framework



**Citation:** Chen, X.; Liu, C.; Chen, L.; Zhu, X.; Zhang, Y.; Wang, C. A Pavement Crack Detection and Evaluation Framework for a UAV Inspection System Based on Deep Learning. *Appl. Sci.* **2024**, *14*, 1157. <https://doi.org/10.3390/app14031157>

Academic Editor: Luis Picado Santos

Received: 13 December 2023

Revised: 20 January 2024

Accepted: 27 January 2024

Published: 30 January 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

Roads are one of the crucial transportation infrastructures that deteriorate over time, due to factors such as heavy vehicles, changing weather conditions, human activity, and the use of inferior materials. This deterioration impacts economic development, travel safety, and social activities [1]. Therefore, it is crucial to periodically assess the condition of roads to ensure their longevity and safety. Additionally, it is imperative to accurately and promptly identify road damage, especially cracks, in order to prevent further deterioration and enable timely repairs.

Currently, pavement condition inspection technologies mainly include traditional manual measurements and automatic distress inspections, such as vehicle-mounted inspection [2]. Manual inspection methods rely primarily on visual discrimination, requiring personnel to travel along roads to identify damage points. However, this approach is slow, laborious, subjective, time-consuming, and has a lower accuracy [3]. Therefore, the development of automatic inspection technologies is crucial for quickly and accurately detecting and identifying cracks on the road. In recent years, intelligent crack inspection systems have gained increasing attention and application, such as vehicle-mounted inspections and

their intelligent systems [4]. Guo et al. [5] utilized core components such as on-mounted high-definition image sensors, laser sensors, and infrared sensors, etc. These components enable the acquisition of high-precision road crack data in real-time. However, the overall configuration of the vehicle-mounted system is expensive and limited in scope, making it challenging to widely apply [2].

Notably, automatic pavement distress inspection has traditionally utilized image-processing techniques such as Gabor filtering [6], edge detection, intensity thresholding [7], and texture analysis. Cracks are identified by analyzing the changes in edge gradients and intensity differences compared to the background, and then extracting them through threshold segmentation [2]. However, these methods are highly influenced by environmental factors, including lighting conditions, which can affect their accuracy. Moreover, these methods are not effective when the camera configurations vary, making their widespread use impractical [1,8]. Given the limitations of these traditional approaches, it is crucial to develop a cost-effective, accurate, fast, and independent method for the accurate detection of road cracks.

In recent years, there have been significant advancements in machine learning and deep learning algorithms, leading to the emergence of automatic deep learning methods as accurate alternatives to traditional object recognition methods. These methods have shown immense potential in visual applications and image analysis, particularly in road distress inspection [1,8]. Krizhevsky et al. [9] proposed a deep convolutional neural network (CNN) architecture for image classification, especially in the detection of distresses in asphalt pavements. Cao et al. [3] presented an attention-based crack network (AC-Net) for automatic pavement crack detection. Extensive experiments on the CRACK500 demonstrated that ACNet achieved a higher detection accuracy compared to eight other methods. Tran et al. [10] utilized a supervised machine learning network called RetinaNet to detect and classify various types of cracks that had developed in asphalt pavements, including lane markers. The validation results showed that the trained network model achieved an overall detection and classification accuracy of 84.9%, considering both the crack type and severity level. Xiao et al. [11] proposed an improved model called C-Mask RCNN, which enhances the quality of crack region proposal generation through cascading multi-threshold detectors. The experimental results indicated that the mean average precision of the C-Mask RCNN model's detection component was 95.4%, surpassing the conventional model by 9.7%. Xu K et al. [12] also proposed a crack detection method based on an improved Faster-RCNN for small cracks in asphalt pavements, even under complex backgrounds. The experiments demonstrated that the improved Faster-RCNN model achieved a detection accuracy of 85.64%. Xu X et al. [13] conducted experiments to evaluate the effectiveness of Faster R-CNN and Mask R-CNN and compared their performances in different scenarios. The results showed that Faster R-CNN exhibited a superior crack detection accuracy compared to Mask R-CNN, while both models demonstrated efficiency in completing the detection task with small training datasets. The study focuses on comparing Faster R-CNN and Mask R-CNN, but does not compare the proposed methods with other existing crack detection methods. In general, these above-mentioned methods not only detect the category of an object, but also determine the object's location in the image [14]. The use of deep learning methods can reduce labor costs and improve work efficiency and intelligence in recognizing road cracks [1].

Meanwhile, unmanned aerial vehicles (UAV) have demonstrated their versatility in a wide range of applications, including urban road inspections. This is attributed to their exceptional maneuverability, extensive coverage, and cost effectiveness [2]. Equipped with high-resolution cameras and various sensors, these vehicles can capture images of the road surface from multiple angles and heights, providing a comprehensive assessment of its condition. Several researchers have utilized UAV imagery to study deep learning methods for road crack object detection, and they have achieved impressive accuracy results. Yokoyama et al. [15] proposed an automatic crack detection technique using artificial neural networks. The study focused on classifying cracks and non-cracks, and the



algorithm achieved a success rate of 79.9%. Zhu et al. [2] utilized images collected by a UAV to conduct experimental comparisons of three deep learning target detection methods (Faster R-CNN, YOLOv3, and YOLOv4) via convolutional neural networks (CNN). The study verified that the YOLOv3 algorithm is optimal, with an accuracy of 56.6% mAP. In another study, Jiang et al. [16] proposed an RDD-YOLOv5 algorithm with Self-Attention for UAV road crack detection, which significantly improved the accuracy with an mAP of 91.48%. Furthermore, Zhang et al. [17] proposed an improved YOLO3 algorithm for road damage detection from UAV imagery, incorporating a multi-layer attention mechanism. This enhancement resulted in an improved detection accuracy with an mAP of 68.75%. Samadzadegan et al. [1] utilized the YOLOv4 deep learning network and evaluated its performance using various metrics such as F1-score, precision, recall, mAP, and IoU. The results showed that the proposed model had an acceptable performance in road crack recognition. Additionally, Zhou et al. [18] introduced a UAV visual inspection method based on deep learning and image segmentation for detecting cracks on crane surfaces. Moreover, Xiang et al. [19] presented a lightweight UAV road crack detection algorithm called GC-YOLOv5s, which achieved an accuracy validation mAP of 74.3%, outperforming the original YOLOv5 by 8.2%. Wang et al. [20] introduced BL-YOLOv8, an improved road defect detection model that enhances the accuracy of detecting road defects compared to the original YOLOv8 model. BL-YOLOv8 surpasses other mainstream object detection models, such as Faster R-CNN, SDD, YOLOv3-tiny, YOLOv5s, YOLOv6s, and YOLOv7-tiny, by achieving detection accuracy improvements of 17.5%, 18%, 14.6%, 5.5%, 5.2%, 2.4%, and 3.3%, respectively. Furthermore, Omobamije et al. [21] proposed an improved CNN method based on UAV imagery, demonstrating a remarkable accuracy of 99.04% on a customized test datasets. Lastly, Zhao et al. [22] proposed a highway crack detection and CrackNet classification method using UAV remote sensing images, achieving 85% and 78% accuracy for transverse and longitudinal crack detection, respectively. These aforementioned studies primarily aim to enhance the deep learning algorithm using UAV images. This enhancement improves the accuracy of road crack detection and also establishes the methodological foundation for the crack target recognition algorithm discussed in this paper.

However, most of the above-mentioned studies primarily focused on UAV detection algorithms and neglected UAV data acquisition and high-quality imagery integrated into detection methods. For instance, the flight settings required for capturing high-quality images have not been thoroughly studied [2]. Flying too high or too fast may result in poor-quality images [22]. Zhu et al. [2] and Jiang et al. [16] both introduced flight setup and experimental tricks for efficient UAV inspection. Liu K.C. et al. [23] proposed a systematic solution for automatic crack detection for UAV inspection. These studies are still incomplete due to a lack of detailed data acquisition and pavement distress assessment. Additionally, there is a lack of quantitative measurement methods for cracks, which hampers accurate data support for road distress evaluation. Furthermore, inconsistency in flight altitude and the absence of ground real-scale information of cracks adversely impact the subsequent quantitative assessment of cracks.

Obviously, existing studies frequently lack a systematic solution or integrated framework for UAV inspection technology, which hinders its widespread application in pavement distress detection. Therefore, this study aims to propose a formal and systematic framework for automatic crack detection and pavement distress evaluation in UAV inspection systems, with the goal of making them widely applicable.

Our proposed framework for a UAV inspection system for automatic road crack detection offers several advantages: (1) It demonstrates a more systematic solution. The framework integrates data acquisition, crack identification, and road damage assessment in orderly and closely linked steps, making it a comprehensive system. (2) It exhibits a greater robustness. By adhering to the flight control strategy and model deployment scheme, the drone ensures high-quality data collection while employing state-of-the-art automatic detection algorithms based on deep learning models that guarantee accurate

crack identification. (3) It presents an enhanced practicality. The system utilizes the cost-effective DJI (DJI-Innovations Company, headquartered in Shenzhen, China) Min2 drone for imagery acquisition and DL-based model deployment, making it an economically viable solution with significant potential for widespread implementation.

The rest of this paper is organized as follows: Section 2 presents the framework for the UAV inspection system designed specifically for pavement distress analysis. In Section 3, we provide a comprehensive overview of four prominent deep-learning-based crack detection algorithms, namely Faster-RCNN, YOLOv5s, YOLOv7-tiny, and YOLOv8s, along with their distinctive characteristics. Section 4 elaborates on the well-defined procedures employed for UAV data acquisition and subsequent data reprocessing. The experimental setup and comparative results are presented in Section 5. In Section 6, we propose quantitative methods to evaluate road cracks and assess pavement distress levels. Finally, in Section 7, we summarize our research while discussing future work.

### 2. Framework of UAV Inspection System

To enhance the practical application of UAV inspection systems in road crack detection, this study presents a comprehensive DL-based method and technical solution framework. As illustrated in Figure 1, the technical framework consists of four main components: (1) Data Acquisition: a flight suitability parameter model is established to ensure high-quality pavement imagery acquisition by the UAV. Prior image data are utilized to create crack datasets for model training, while the subsequent phase is directly employed for pavement crack detection. (2) Model Training and Evaluation: UAV imagery is pre-processed through frame extraction, image dividing, and data enhancement, and then labeled according to five major categories of cracks (longitudinal, transverse, diagonal, mesh, and no cracks) to create the datasets. Based on this, four mainstream DL target detection algorithms (Faster-RCNN, YOLOv5, YOLOv7, and YOLOv8) are individually conducted for the road crack detection model training. Finally, the models are compared and validated using precision ( $P$ ), recall ( $R$ ), F1-score, and mean accuracy precision ( $mAP$ ) as evaluation metrics, and the best model is selected. (3) Model Application and Road Crack Detection: The preferred model is employed to identify road crack targets using UAV imagery. To reduce the computing resources, the full-scale images are divided into smaller images before detection. (4) Road Distress Evaluations: Quantitative assessments (for instance, cracks count, cracks length, and crack area etc.) are conducted to evaluate pavement distress, which provide factual evidence and a solid data foundation for evaluating road damage and planning road repair work for transportation departments.

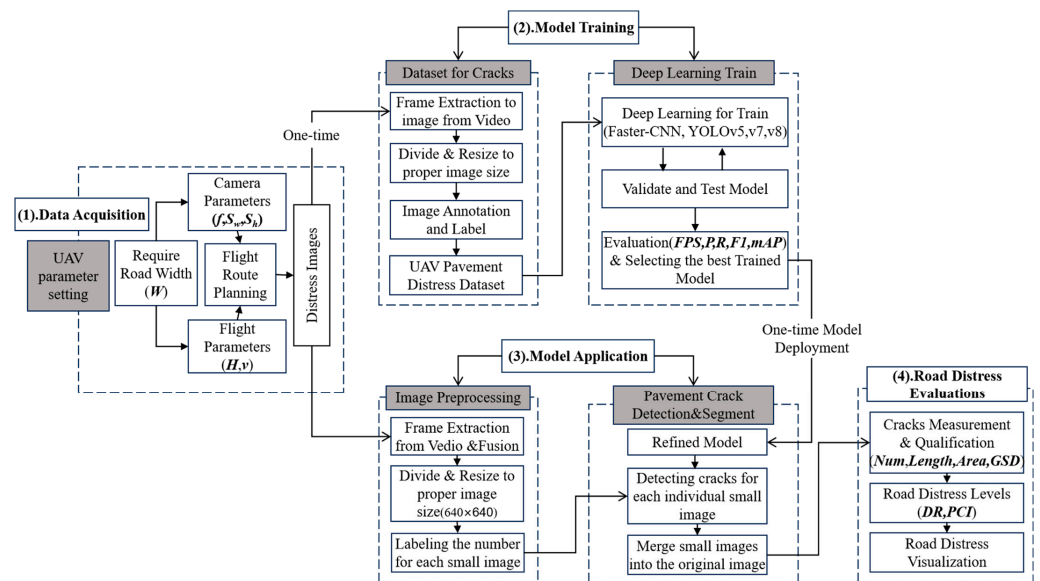
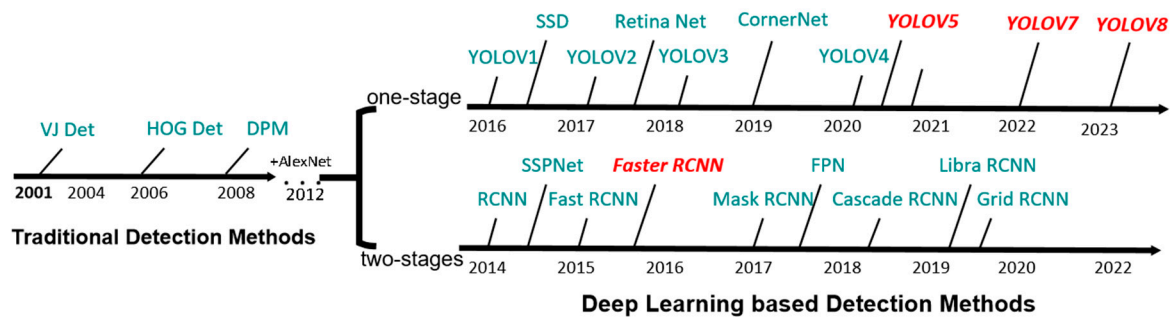


Figure 1. A framework of UAV inspection system based on deep learning for pavement distress.

### 3. Deep Learning Algorithms

In recent years, there has been significant progress in deep learning technology, leading to a paradigm shift in target detection methods from traditional algorithms based on manual features to deep-neural-network-based detection methods [24]. These deep learning algorithms can be categorized into two major approaches (Figure 2): (1) Two-stage methods (two-stage algorithms), which involve labeling multiple target candidate regions in the image and subsequently classifying and regressing the boundary of each candidate region. Representative algorithms belonging to this approach include the RCNN series. (2) Single-stage methods (one-stage algorithms), which directly perform the localization and classification of all detected targets across the entire image without requiring the explicit labeling of candidate regions. Representative algorithms belonging to this approach include the YOLO (You Only Look Once) series. Both approaches have their own advantages, with the single-stage algorithm being faster and the two-stage algorithm being more accurate. Therefore, this study selects the Faster RCNN algorithm [25] and the YOLOv5 algorithm [26] as typical representatives of these two major approaches. Additionally, the latest improved algorithms of YOLOv5, namely YOLOv7 [27] and YOLOv8 [28], are introduced into comparative validation in this study on the use of deep learning algorithms for road crack detection using drones.



**Figure 2.** A road map of object detection models (modified from [23]).

#### 3.1. Faster-RCNN Algorithm

The Faster-RCNN algorithm is a typical representative example of a two-stage algorithm for target detection. The Faster-RCNN model consists of four components: a Backbone, Region Proposal Networks (RPN), ROI (Region of Interest) Pooling, and Classifier. The Backbone extracts a feature map that is used for candidate detection area extraction and classification. The RPN further refines the candidate detection areas based on the initial feature map, which may contain the target features. These refined areas are then used for further classification and localization. The ROI Pooling fine-tunes the candidate detection areas based on their candidate box coordinates. Finally, the Classification component uses the proposals and feature maps to determine the category of the proposal and regress the candidate detection frames to obtain their final precise locations.

The network architecture of Faster-RCNN is illustrated in Figure 3. Firstly, an arbitrary input image ( $P \times Q$ ) is resized to a standard image ( $M \times N$ ) and then fed into the network. The backbone (e.g., VGG and ResNet, etc.) extracts features from the  $M \times N$  image, followed by convolution and pooling operations, resulting in feature maps for this input. These feature maps contain information about different scales and semantics, enabling the detection of objects with various scales and shapes in the image. In the Region Prediction Network (RPN), the RPN network performs a  $3 \times 3$  convolution to generate Positive Anchors and the corresponding Bounding Box Regression offsets. It then calculates Proposals, which are utilized by the ROI pooling layer to extract the Proposals from the Feature Maps. The Proposal Feature is further processed through fully connected and softmax networks for classification.

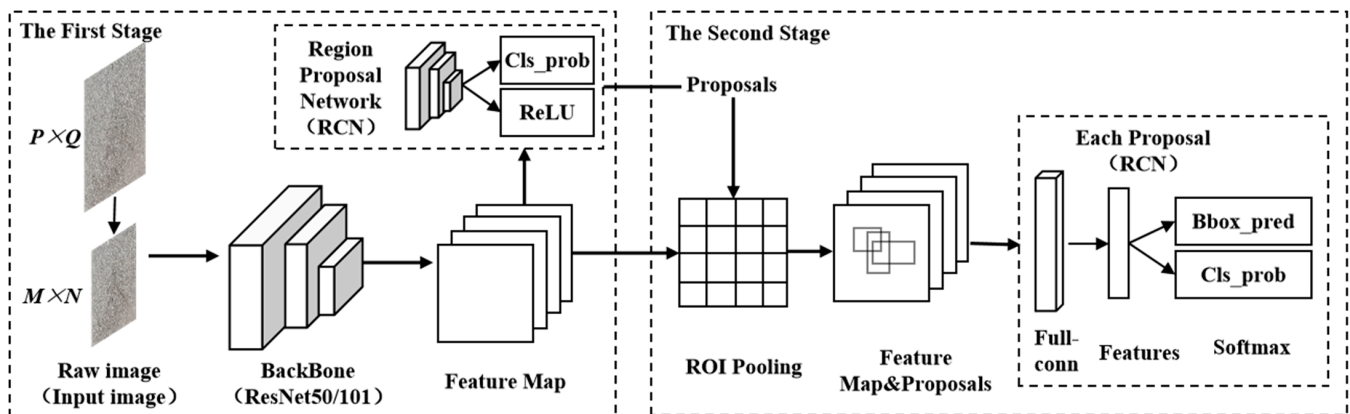


Figure 3. An illustration of Faster-RCNN (modified from [29]).

### 3.2. YOLO Series Algorithms

The YOLO series algorithm is a typical representative example of the one-stage algorithm target detection model. In comparison to the Faster RCNN algorithm, YOLO eliminates the need to extract candidate regions that may contain targets. It completes the detection task using only one network and predicts the category and location of the target object in the detection output through regression. Currently, YOLOv5 is the initial model of the series, which has been proven to be stable and is widely used in lightweight road crack detection methods due to its excellent accuracy [17,21]. YOLOv5 consists of several networks with different depths, namely n, s, m, l, and x. The depth and width of the network increase in the order of n, s, m, l, and x. Among these options, YOLOv5s is suitable for small deep networks or small-scale datasets.

The network architecture of YOLOv5 is depicted in Figure 4. The model comprises three main components: the backbone network (BackBone), the neck network (Neck), and the head detection network (Head). The backbone network (BackBone) primarily performs feature extraction by utilizing a convolutional network to extract object information from the image. This information is then used to create a feature pyramid, which is later employed for target detection. The backbone network consists of various modules, such as the Focus module, Conv module, C3 module, and SPPF module. Notably, the SPPF (Spatial Pyramid Pooling Faster) module is capable of converting feature maps of any size into fixed-size feature vectors. This allows for the fusion of local and global features at the Feature Map level and further expands the receptive field of the feature map. Consequently, objects can be effectively detected even when input at different scales. The neck network (Neck) is responsible for the multi-scale feature fusion of the feature map. It adopts the structure of the Feature Pyramid Network (FPN) and the Path Aggregation Network (PAN), which enhances the model's ability to capture object features at various scales and improves the accuracy and performance of target detection. The head network (Head), also known as the detection module, utilizes techniques like anchor boxes to process the input feature mapping and generate regression predictions. These predictions include information about the type, location, and confidence of the crack detection object.

YOLOv7 [27] is an enhanced target detection framework based on YOLOv5. It incorporates a deeper network structure and robust methods, resulting in an improved accuracy and speed compared to YOLOv5. YOLOv7 introduces several techniques, such as Long-Range Attention Off Network (ELAN) and Bottleneck Attention Module (BAM), to enhance its learning capability. ELAN expands, shuffles, and merges the quantity (Cardinality), thereby improving the equilibrium state of the learning network. To prevent overfitting, YOLOv7 employs a regularization method similar to DropBlock. This regularization method enhances the stability and robustness of the model, enabling it to be trained on larger datasets.

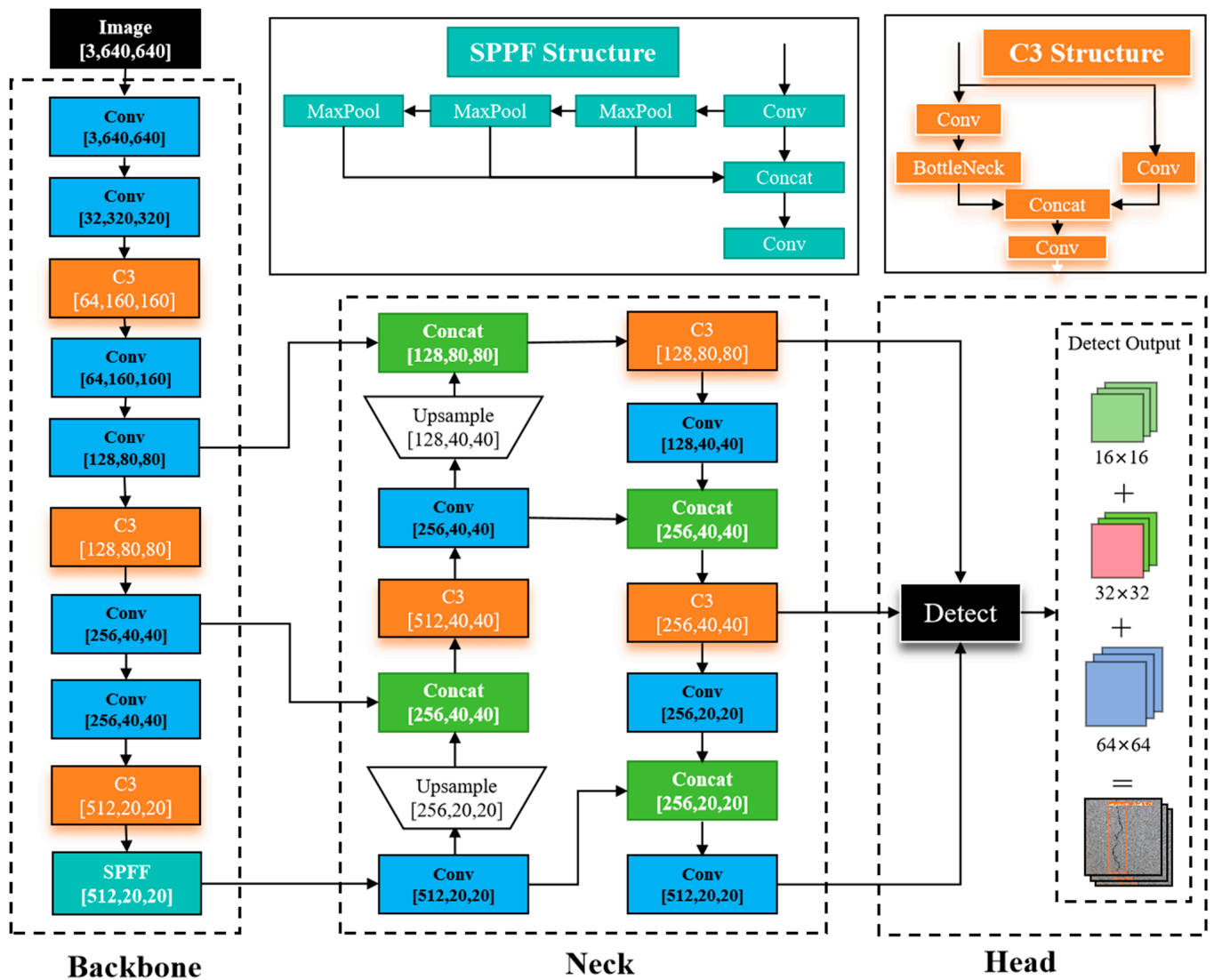


Figure 4. Network architecture of YOLOv5.

YOLOv8 [28] was released in January 2023 by Ultralytics, the company that developed YOLOv5. YOLOv8 further optimizes the model structure and training strategy based on YOLOv7 to enhance both detection speed and accuracy. Notably, YOLOv8 incorporates a more efficient long-range attention network called Extended-ELAN (E-ELAN), which enhances the model’s feature extraction capability. Moreover, YOLOv8 introduces new loss functions, such as VFL Loss and Loss+DFL (Distribution Focal Loss), to improve the model’s localization accuracy and category differentiation ability. Additionally, YOLOv8 employs new data enhancement methods, including Mosaic + MixUp, to enhance the generalization and robustness of the model.

In the current field of deep learning models, Faster-RCNN, YOLOv5, YOLOv7, and YOLOv8 are all target detection methods known for their high accuracy and advanced algorithms. However, there are some variations in terms of model structure, accuracy, speed, training strategy, and robustness. The selection of the appropriate algorithm should be based on specific requirements and application scenarios to effectively address the needs of UAV road crack target detection.



## 4. UAV Data Acquisition and Preprocessing

### 4.1. Flight Control Strategy

During the flight process of a UAV equipped with a high-definition camera, the acquired imagery may suffer from distortion, degradation, or uncovered road due to improper human control or mismatched flight parameters. Therefore, it is crucial to establish a flight control strategy and experimental techniques for the UAV flight parameters to enhance the quality of the imagery captured by the UAV in real-world scenarios.

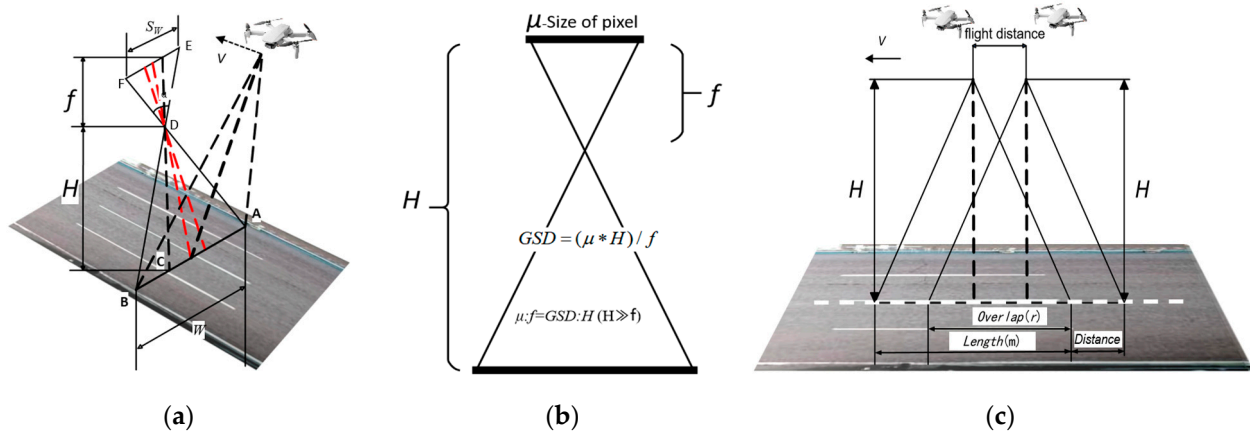
#### 4.1.1. Flight Height

To determine the optimal altitude of the UAV and ensure its efficient flight, the following considerations should be taken into account: (i) the UAV camera view should cover the full width of the road that needs to be inspected; (ii) it is important to avoid any interference from auxiliary facilities such as road trees and street lights during the flight; and (iii) to minimize image distortion, it is crucial to maintain a constant altitude, consistent speed, and capture vertical imagery.

To cover the full width of the pavement, a minimum flight altitude is required. Based on the Pinhole Imaging Principle and the Triangular Similarity Geometric Relationship (Figure 5a), the minimum flight altitude should satisfy Equation (1):

$$H \geq (f * W) / S_w \tag{1}$$

where  $H$  represents flight vertical height.  $f$  represents the focal length of the camera.  $W$  represents the width of the road to be inspected.  $S_w$  represents the camera sensor size ( $S_w \times S_h$ ).



**Figure 5.** Diagram of the main flight parameters (a,c) modified from [16]. (a) flight height ( $H$ ); (b) ground sampling distance ( $GSD$ ); and (c) flight velocity ( $v$ ).

In our experiment, the DJI Mini 2 drone was chosen to perform the flight mission. The camera sensor format was CMOS 1/2.3 inches, with a full-frame sensor size of 17.3 mm  $\times$  13.0 mm. The main focal length ( $f$ ) was 24.0 mm. The experimental pavement consisted of a bi-directional eight-lane road. To ensure high-definition imagery quality, the experiment was conducted only on the left lane, from east to west. The width of the road ( $W$ ) was measured to be 16 m. The minimum flight altitude was calculated at 22.20 m. Taking into account the tolerance for flight stability, the final flight height was chosen as 22.5 m.

#### 4.1.2. Ground Sampling

The Ground Sampling Distance ( $GSD$ ) is a crucial parameter in remote sensing and image processing. It quantifies the distance between the individual pixels in an image and the ground truth, which directly affects the accuracy of geospatial measurements for

cracks. (i) DJI can officially provide GSD values that are applicable to a wide range of focal lengths [16]. The most commonly observed GSD value, typically associated with a 24 mm focal length, is calculated as  $H/55$ . (ii) Alternatively, GSD can be derived directly from the diagram in Figure 5b, using Equation (2):

$$GSD \approx (\mu * H) / f \quad (2)$$

where GSD represents the ground sampling distance of a flight, and its unit is cm/pixel;  $\mu$  is the image pixel calibration size ( $\mu\text{m}$ ), which can be officially provided by DJI. Take DJI Mini 2 as an example, where  $\mu$  is given as 4.4  $\mu\text{m}$ . If the flight height is 22.5 m, thereby GSD can be computed as 0.4125 cm/pixel.

#### 4.1.3. Flight Velocity

The appropriate flight velocity is also essential in UAV imagery acquisition to avoid redundancy and motion blurring. It should be determined based on the degree of overlap between neighboring images and the consistency and quality stability of the aerial images. Typically, a minimum forward overlap of 75% and a minimum side overlap of 60% are recommended. Figure 5c illustrates how the flight velocity can be calculated based on the desired overlap degree and the sampling frequency of the neighboring frames, using Formula (3):

$$v = L * (1 - r) / t \quad (3)$$

where  $v$  is the flight velocity (m/s) and  $t$  represents the shooting interval of two adjacent images (s), typically set to 2. The overlap degree ( $r$ ) is defined as the forward overlap and is commonly taken as 50–75%, since UAVs are often operated at the same speed and uniform linear motion with the forward direction.  $L$  represents the ground truth length of the road in an image (m).  $L$  can be determined based on the GSD and the road width ( $W$ ) covered by the UAV imagery, using the following formula.

$$L = W / GSD = (W * f) / (\mu * H) \quad (4)$$

In our experiment, the road width ( $W$ ) was chosen as 16.0 m and the GSD was computed as 0.4125 m/pixel using Equation (2), thereby, the ground-truth road length ( $L$ ) in an image was calculated as 38.78 m using Equation (4). Given that the sampling frequency  $t$  was 2 s, the forward overlap of the captured images was set to 75%. According to the Equation (3), the minimum speed  $v$  was 4.85 m/s, and finally, 5 m/s was determined as the flight velocity for this experiment.

## 4.2. UAV Imagery Data Preprocessing

### 4.2.1. Frame Extraction and Fusion from UAV Imagery Video

Video frame data play a crucial role in acquiring UAV pavement crack images. In order to obtain and supplement the original pavement crack datasets, it is necessary to extract and crop the frames. During the frame extraction process, it is important to consider the overlap, spacing, and seamlessness of neighboring video frames to ensure the integrity and independence of each frame. Additionally, the setting of the frame extraction interval is of the utmost importance. If the interval is too large, it may result in a lack of seamless fusion and docking. On the other hand, if the interval is too small, there will be significant overlap between frames, leading to an excessive number of frames and an increased computing cost. The formula for calculating the extraction interval number ( $N$ ) is as follows:

$$N \leq \left[ \frac{GSD \times F_l \times (1 - r)}{v} \times fps \right] \quad (5)$$

where  $F_l$  is the frame image size (px), i.e., the flight direction;  $fps$  represents the frames per second in each video; and the other variables are described in the previous section.

For this experiment, UAV imagery in the DJI Mini 2 was set to 4K HD, which corresponds to the DJI official frame image size of  $3840 \text{ px} \times 2160 \text{ px}$ . Namely,  $F_l$  was taken as  $3840 \text{ px}$ . The  $\text{fps}$  was officially  $24 \text{ f}\cdot\text{s}^{-1}$ , and  $GSD$  and  $v$  were calculated to be  $0.4125 \text{ cm/pixel}$  and  $5.0 \text{ m/s}$  as above, respectively. The overlap ( $r$ ) was taken as  $75\%$ . Using Equation (5), the extraction interval number ( $N$ ) was found to be  $[19.01]$ , which was rounded to 19. Finally, to ensure sufficient overlap, this study extracted an image every 19 frames from the video. The extracted frame images were then used to stitch together overlapping parts of neighboring frames using the picture fusion technique.

#### 4.2.2. Pavement Cracks Datasets with GSD Information

Due to the large size of the acquired images or frame images, utilizing them directly as inputs for model training would lead to a sluggish training speed and a significant consumption of processing resources. Therefore, to enhance the parallel batch computing speed, deep learning models have specific requirements for training image datasets. The original images need to be trimmed after frame extraction and fusion, resulting in images with consistent specifications. In this experiment, the extracted frame images from videos were used as the initial images, which were further cropped into  $640 \text{ px} \times 640 \text{ px}$  specification images. The cropping process is shown in Figure 6. Assuming the original image size of a frame was a  $3840 \text{ px} \times 2160 \text{ px}$  road image, 18  $640 \text{ px} \times 640 \text{ px}$  specification images could be cropped. To expand the number of samples for crack categories, data enhancement methods such as augmentation, translation, flipping, and rotation were applied. Additionally, the blurred images were removed to ensure they did not affect the training effect of the model training. The UAV crack original datasets were constructed by manually screening, classifying, and confirming the coverage of the four major crack categories and the no-crack images.

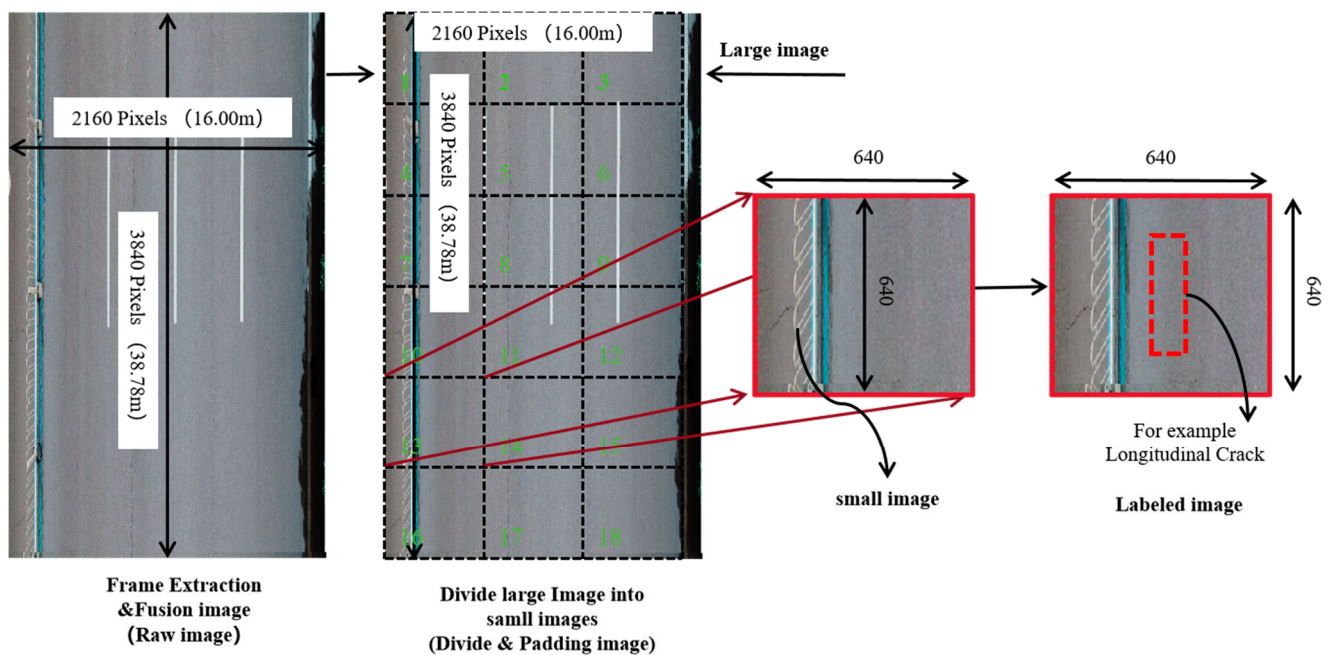
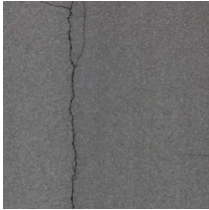
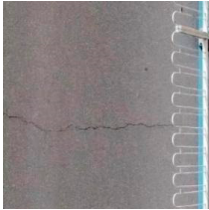





Figure 6. Diagram of trimming process for the large frame image (modified from [10]).

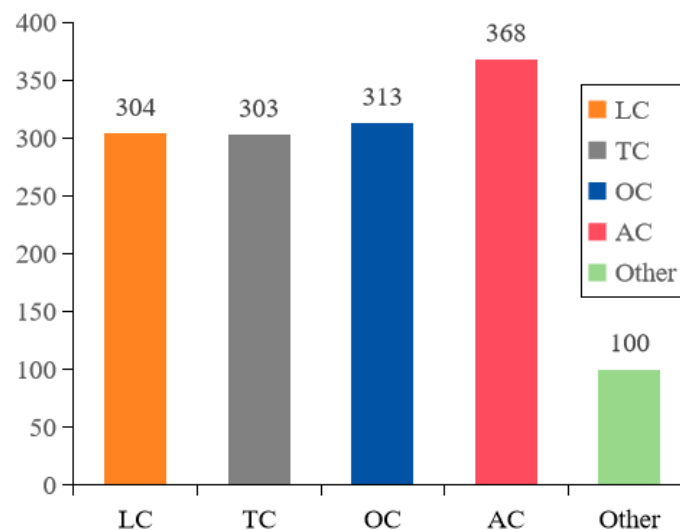
Road crack labeling plays a crucial role in training and testing deep learning models. The accuracy of labeling directly impacts the quality of model learning. In this experiment, we employed various methods, including manual visual labeling and the Labeling tool, to decipher, mark, and categorize different types of cracks based on the original UAV crack datasets. The goal was to create an improved training set for crack recognition. Based on the prominence of cracks and their associated damage hazards, road cracks were categorized

into four types: longitudinal cracks, transverse cracks, diagonal cracks, and mesh cracks. These categories are illustrated in Table 1.

**Table 1.** Classification and description of road cracks.

Longitudinal Crack (LC)	Transverse Crack (TC)	Oblique Crack (OC)	Alligator Crack (AC)	No-Cracks (Other)
				

Generally, the problem of imbalanced sample distribution in datasets can often lead to overfitting of the model [30]. To address this issue, this experiment fully considered the balance of the sample distribution when creating the labeling datasets. Each type of road pavement crack had a more equal number distribution, as shown in Figure 7. A total of 1388 pavement crack images based on a UAV were collected and labeled, with 304 samples identified as being of the longitudinal crack (LC) type, 303 samples identified as being of the transverse crack (TC) type, 313 samples identified as being of the obliquely oriented crack (OC) type, 368 samples identified as being of the alligator crack (AC) type, and 100 samples being identified as of the no-crack type. To ensure the DL-based model's effectiveness, the datasets were divided into training, validation, and test sets in the ratio of 80%, 10%, and 10%, respectively.



**Figure 7.** Samples and distribution of the pavement crack datasets.

Notably, existing crack datasets often do not provide ground-truth information, particularly regarding the spatial resolution of UAV imagery. This lack of information directly affects the accuracy of crack identification and measurement in future studies. In this study, the UAV data collection process included the recording of the real-time flight height, an important parameter for each image. Thereby, the Ground Sample Distance (GSD) can be calculated using Formula (2), and also documented in each treated image, which is crucial for the subsequent automated evaluation of pavement damage.



## 5. Experiments and Results

### 5.1. Experimental Scenario

In this experiment, the flight mission was located on Xuefu Road, Xiangtan City, Hunan Province, China, as shown in Figure 8. The Xuefu Road is an asphalt pavement with eight two-way lanes and a one-way road width of 16 m. The UAV aerial photography covered a distance of 1.5 km. The road was built and opened to traffic in 2010. After more than 13 years, the road surface has suffered significant damage, including transverse cracks, longitudinal cracks, alligator cracks, and no-cracks. The experiment was conducted at 10:00 a.m. on a sunny day with relatively sparse traffic. Based on the previously obtained UAV flight parameters, the flight height ( $H$ ) was set to 22.5 m and the flight velocity ( $v$ ) was set to 5.0 m/s.



**Figure 8.** Experimental road and scenario.

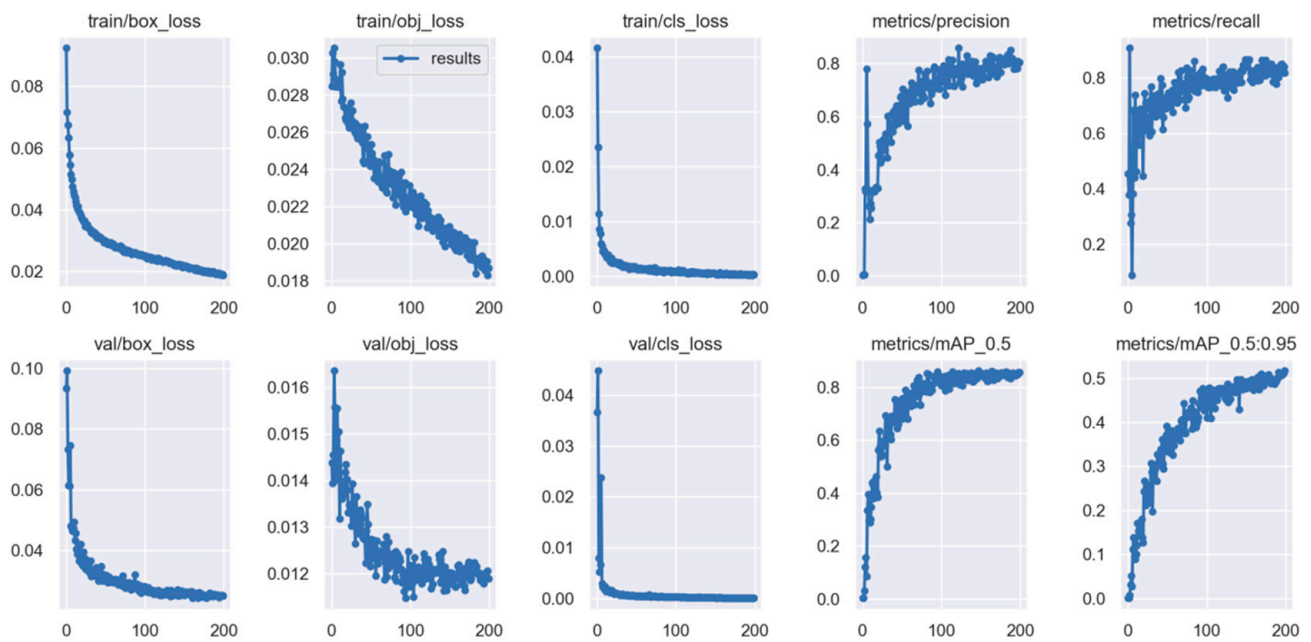
### 5.2. Experimental Configuration

The deep learning algorithms used in this experiment were executed on the same specifications. The specific configuration and experimental environment are detailed in Table 2. The Faster-RCNN model employed the VGG feature extraction network, while YOLOv5, YOLOv7, and YOLOv8 utilized YOLOv5s, YOLOv7-tiny, and YOLOv8s, respectively. The input image size of these models was unified to 640 px × 640 px. The training iterations (Epoch) were set to 200, as depicted in Figure 9. The YOLO algorithm series models were trained with a batch size of eight, whereas the Faster-RCNN used a batch size of four. The experiment's hyperparameters were configured as follows: the initial learning rate was set to 0.01, the learning rate decay employed the Cosine Annealing algorithm, the optimizer used was SGD (Stochastic Gradient Descent), and the Momentum was set to 0.937.

**Table 2.** Configuration of the experimental environment.

Software	Configure	Matrix	Versions
Operating system	Windows10	Python	3.9
CPU	Intel Core i5-9300H	PyTorch	2.0
GPU	NVIDIA GeForce GTX 1660Ti 6G	CUDA	11.8





**Figure 9.** Loss plot of the YOLOv5s model (the optimal iterations number is 200).

### 5.3. Evaluation Metrics of Models

#### 5.3.1. Running Performance

To validate the computational complexity of deep learning models, five evaluation metrics in this experiment were firstly used to assess the algorithm's running performance: the number of parameters, video memory usage, training duration, memory consumption, and frame rate (*FPS*). It is important to note that the *FPS* measures the number of images processed per second and serves as a significant indicator of prediction speed.

#### 5.3.2. Accuracy Effectiveness

Furthermore, in order to demonstrate the algorithm's effectiveness for deep learning models, four evaluation metrics in this experiment were used to access the detection accuracy: Precision (*P*), Recall (*R*), *F1-Score*, Average Precision (*AP*), and Mean Average Precision (*mAP*). *P* represents the probability of correct target detection and is calculated as the ratio of the number of correctly classified samples (*TP*) to the total number of samples. *R* represents the probability of correctly recognizing the target among all positive samples and is calculated as the ratio of the number of correctly classified positive samples to the number of all positive samples. *F1-Score* is a comprehensive evaluation index that takes into account the effects of accuracy and recall. *AP* is obtained by calculating the area under the Precision–Recall curve and reflects the precision of individual crack categories. The *mAP* characterizes the average across the four crack categories and reflects the overall classification precision of the crack prediction.

### 5.4. Experimental Results

To validate the viability of our proposed framework for analyzing UAV imagery crack datasets, this study employed four prominent deep learning algorithms (FasterRCNN, YOLOv5s, YOLOv7-tiny, and YOLOv8s) for conducting pavement crack object detection and a comparative analysis. The experiments were conducted using identical hardware and software environments, with consistent iteration numbers, training datasets, validation datasets, and test datasets. The results were evaluated based on the model performance during execution, the recognition accuracy of the models, and variations in crack category classification.

#### 5.4.1. Comparison Results of Running Performance

The operational performances of the four models are presented in Table 3. Among them, the Faster-RCNN model exhibited, significantly, the lowest performance, with the highest number of parameters ( $136.75 \times 10^6$ ), memory consumption (534.2 MB), and video memory usage (5.6 GB), as well as the longest training duration (7.1 h) and the lowest frame rate ( $12.80 \text{ f}\cdot\text{s}^{-1}$ ). On the other hand, the YOLO series models, which are single-stage algorithms, demonstrated significantly faster running performances. The YOLOv7-tiny model had the fewest parameters and minimal memory requirements, while achieving higher frame rates for YOLOv5s and YOLOv8s, along with a faster execution speed.

**Table 3.** The results of running performance with various models.

Models	Number of Parameters ( $\times 10^6$ )	Training Duration (h)	Memory Consumption (MB)	Video Memory Usage (GB)	FPS ( $\text{f}\cdot\text{s}^{-1}$ )
Faster-RCNN	136.75	7.1	534.2	5.6	12.80
YOLO v5s	7.02	3.7	14.12	3.5	127.42
YOLO v7-tiny	6.01	3.8	12.01	1.9	82.56
YOLO v8s	11.13	3.1	21.98	3.6	125.74

When considering identical datasets, it can be concluded that the Faster-RCNN model required a superior running performance and environment configuration, whereas the YOLO model series required a lower hardware and software environment configuration, while offering faster training speeds. Consequently, the YOLO model series algorithms are highly suitable for the lightweight deployment of real-time detection tasks on UAV platforms.

#### 5.4.2. Comparison Results of Detection Accuracy

##### The Results of Overall Detection Accuracy

The results of comparing the overall detection accuracy are presented in Table 4. Among all models, the Faster-RCNN model demonstrated the highest accuracy, surpassing the YOLO series models in all evaluation indexes of accuracy. It achieved a precision (*P*), recall (*R*), F1 value, and mean average precision (*mAP*) of 75.6%, 76.4%, 75.3%, and 79.3%, respectively. Among the YOLO series models, YOLOv7-tiny exhibited a lower overall precision with values of 66.9% (*P*), 66.5% (*R*), 66.7% (*F1-score*), and 65.5% (*mAP*). On the other hand, both YOLOv5s and YOLOv8s showed similar overall precision, but were slightly inferior to Faster-RCNN by approximately a margin from around 3% to 5%.

**Table 4.** Results of overall accuracy with various models (%).

Models	Precision	Recall	F1-Score	mAP
Faster-RCNN	75.6	76.4	75.3	79.3
YOLO v5s	75.1	71.0	72.6	74.0
YOLO v7-tiny	66.9	66.5	66.7	65.5
YOLO v8s	74.4	75.6	75.0	77.1

##### The Results of Detection Accuracy under Different Crack Types

To further clarify the discrepancies in the model recognition accuracy among the different crack categories, a comparative analysis of the model recognition accuracy was conducted for the four types of cracks: longitudinal cracks (*LC*), transverse cracks (*TC*), oblique cracks (*OC*), and mesh cracks (*AC*). The results are presented in Table 5.

- (i) Regarding the identification of longitudinal cracks (*LC*), the Faster-RCNN model exhibited the highest accuracy, with an average precision (AP) of 85.7% and the highest F1 value of 82.3%. In contrast, the YOLO series demonstrated a relatively inferior

- average precision, with YOLOv7-tiny exhibiting the lowest performance. Therefore, Faster-RCNN outperformed the other models in recognizing longitudinal cracks.
- (ii) For transverse cracks (TC), the YOLOv8s model achieved a superior recognition accuracy with an AP score of 89.5%, followed by YOLOv5s. Although there was a slight decrease in F1 score for YOLOv8s compared to YOLOv5s, their overall recognition accuracies did not significantly differ from each other; however, YOLOv7-tiny displayed a weaker recognition accuracy.
  - (iii) All four algorithm models exhibit low recognition accuracy and F1 values for oblique cracks (OC) compared to the other types of cracks; however, among them, Faster-RCNN still maintained the highest level of recognition accuracy, while all models belonging to the YOLO series demonstrated lower levels of recognition accuracy—this explains why Faster-RCNN performed better overall.
  - (iv) In terms of recognizing mesh cracks (AC), an outstanding performance was observed from the YOLOv8s model, which attained a remarkable recognition accuracy and F1 value at 91.0% and 90.6%, respectively; meanwhile, although slightly less effective than its counterpart, the YOLOv5s model also showcased a commendable performance, whereas a poor performance was exhibited by the YOLOv7-tiny model.

**Table 5.** Results of detection accuracy with various models under four crack types (%).

Models	AP (%)				F1-Score			
	TC	LC	AC	OC	TC	LC	AC	OC
Faster-RCNN	85.7	83.4	60.2	87.8	82.3	78.0	58.1	82.9
YOLO v5s	75.5	87.4	43.8	89.1	72.3	86.5	43.5	88.0
YOLO v7-tiny	70.4	81.2	40.7	80.7	70.0	79.0	44.8	77.1
YOLO v8s	75.4	89.5	45.4	91.0	74.4	85.0	48.5	90.6

#### The Results of Detection Accuracy under Different Crack Datasets

In this study, our self-made pavement crack datasets strictly followed the UAV flight parameter settings and data acquisition process mentioned in Section 4. To validate the reliability and advantages of our self-made crack datasets, we conducted a comparative study using these four model algorithms on existing various open-source UAV pavement crack datasets. Our experiment involved comparing the detection accuracy of our crack datasets with datasets such as UAPD [2], RDD2022 [31], UMSC [19], UAVRoadCrack [21], and CrackForest [32]. We evaluated and compared the accuracy performances of Faster-RCNN, YOLOv5, YOLOv7-tiny, and YOLOv8s after 200 training cycles, as well as Faster-RCNN after 15 rounds.

The results, as presented in Table 6, indicate that our lab's datasets outperformed other datasets used in similar models on most metrics, exhibiting the highest accuracy for crack recognition and algorithmic efficiency. However, the model performance varied across different datasets; while UAVRoadCrack performed relatively well, the UAPD dataset showed the worst performance. These findings strongly highlight the advantages of utilizing our self-collected pavement images via a UAV and emphasize the importance of flight parameter modeling for the quality control of UAV imagery.

#### The Results of Detection Effectiveness

To facilitate a more intuitive comparison of the effects, a specific image with four types of cracks was selected from the test set to evaluate and compare their recognition performances, as presented in Table 7. Based on the results obtained, it is evident that Faster-RCNN outperformed the YOLO series algorithms in terms of overall performance. It is worth noting that, for challenging oblique cracks (OC), all the YOLO series algorithms exhibited unsatisfactory recognition with low confidence levels, often resulting in the omission or separate identification of complete cracks, whereas the Faster-RCNN model demonstrated a superior capability in recognizing oblique cracks (OC) more comprehen-



sively. Additionally, the Faster-RCNN model also exhibited an excellent performance in detecting subtle cracks, as shown in Table 7. For instance, it successfully identified a subtle transverse crack within a longitudinal crack. In other crack types, all four modeling algorithms demonstrated effectiveness in crack detection. A comparative analysis considering both combined effects and confidence levels reveals that Faster-RCNN achieved the best overall performance; among the YOLO series algorithms, YOLOv5s and YOLOv8s showed comparable results, while YOLOv7-tiny performed relatively poorly, with lower confidence levels observed across all detected results.

**Table 6.** Comparison of model valuation with various UAV cracks datasets.

Datasets	Faster-RCNN			YOLO v5s			YOLO v7-Tiny			YOLO v8s		
	FPS (f.s <sup>-1</sup> )	F1 (%)	mAP (%)	FPS (f.s <sup>-1</sup> )	F1 (%)	mAP (%)	FPS (f.s <sup>-1</sup> )	F1 (%)	mAP (%)	FPS (f.s <sup>-1</sup> )	F1 (%)	mAP (%)
UAPD [2]	9.14	47.9	48.8	59.7	52.7	57.7	74.51	56.7	52.8	65.4	57.4	58.6
RDD2022 [31]	11.36	69.5	68.8	63.21	65.2	60.9	65.47	63.1	65.6	53.71	66.5	67.7
UMSC [19]	11.72	73.4	68.8	97.87	68.7	74.3	76.81	63.8	70.1	89.78	72.8	70.4
UAVRoadCrack [21]	10.57	68.9	68.5	108.6	77.8	75.7	75.39	62.5	65.3	69.36	71.0	68.8
CrackForest [32]	/	57.4	59.1	/	57.8	58.8	67.45	61.2	63.5	61.21	60.9	65.2
Our Datasets	12.80	75.3	79.3	127.4	72.6	74.0	82.56	66.7	65.5	125.7	75.0	77.1

**Table 7.** Comparison of detection results with various models.

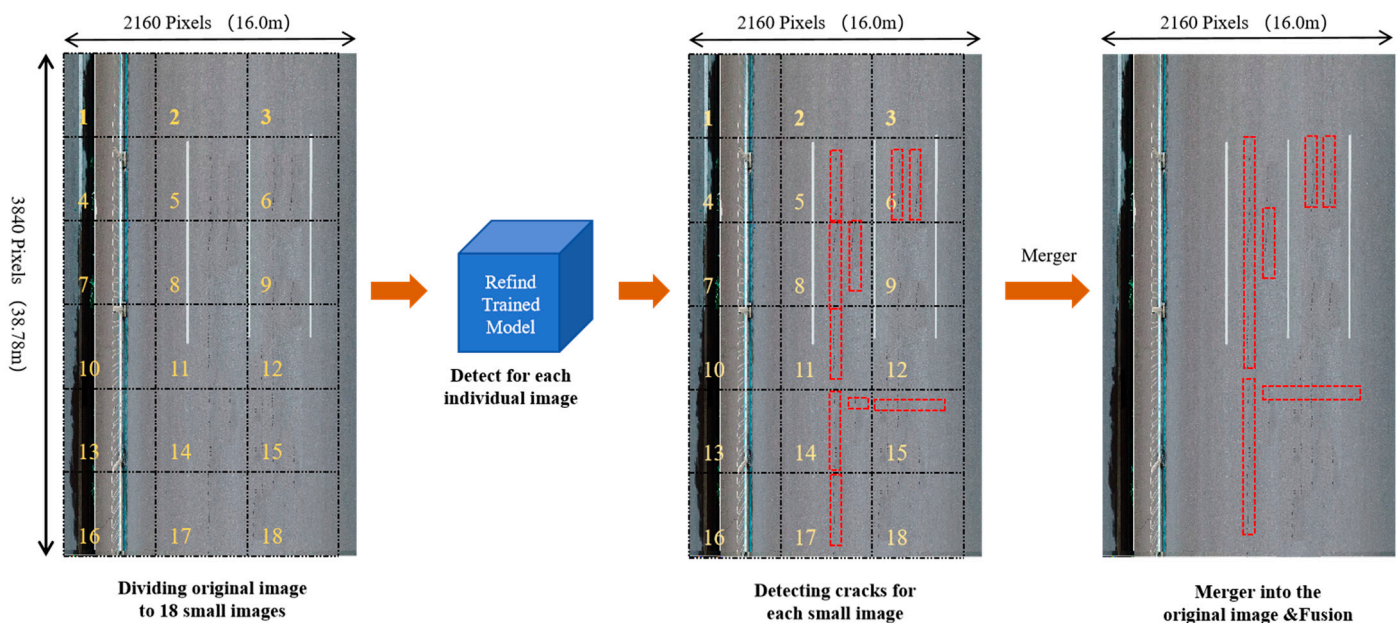
Input Images	Faster-RCNN	YOLOv5s	YOLOv7-Tiny	YOLOv8s

In summary, this experiment compared the accuracy and effectiveness of different models for crack recognition from UAV imagery. The Faster-RCNN model demonstrated the highest accuracy and effectiveness in recognizing fine cracks. On the other hand, the YOLO series model showed significant advantages in terms of training speed and low requirements for video memory. Among the YOLO models, YOLOv5s and YOLOv8s exhibited a comparable recognition accuracy, while YOLOv7-tiny performed the worst. The experiment primarily focused on evaluating the data acquisition quality of UAV imagery, which yielded optimal results in the testing phase.

## 6. Road Crack Measurements and Pavement Distress Evaluations

The primary goal of road crack recognition is to evaluate pavement damage on roads. This will help to enhance the application of these models and provide factual evidence for the maintenance decisions made by road authorities. After conducting a comparative study of various modeling algorithms, it was determined that the model trained by Faster-RCNN outperformed the YOLO serial models and could be identified as the refined model for this experiment.

Due to the large size of the obtained images, it is not efficient to use them directly for road crack recognition. This would result in a slow recognition speed and require a significant amount of processing resources. To address this issue and ensure that the UAV recognition model remained small and fast, the strategy of ‘Divide and Merge’ was employed into the UAV imagery with large-size photos. This strategy utilizes a ‘Divide-Recognition-Merge-Fusion’ method during crack detection, as illustrated in Figure 10. The original frame image (3840 px × 2160 px) was divided into 18 consistently smaller images (640 px × 640 px), each assigned a unique number. Using the optimal model trained by Faster-RCNN in this experiment, cracks were identified within each cropped image. Finally, these identified images were stitched together, with overlapping multi-crack confidence recognition boxes merging with neighboring combinations.



**Figure 10.** Illustration of “Divide and Merge strategy” of UAV imagery for crack detection.

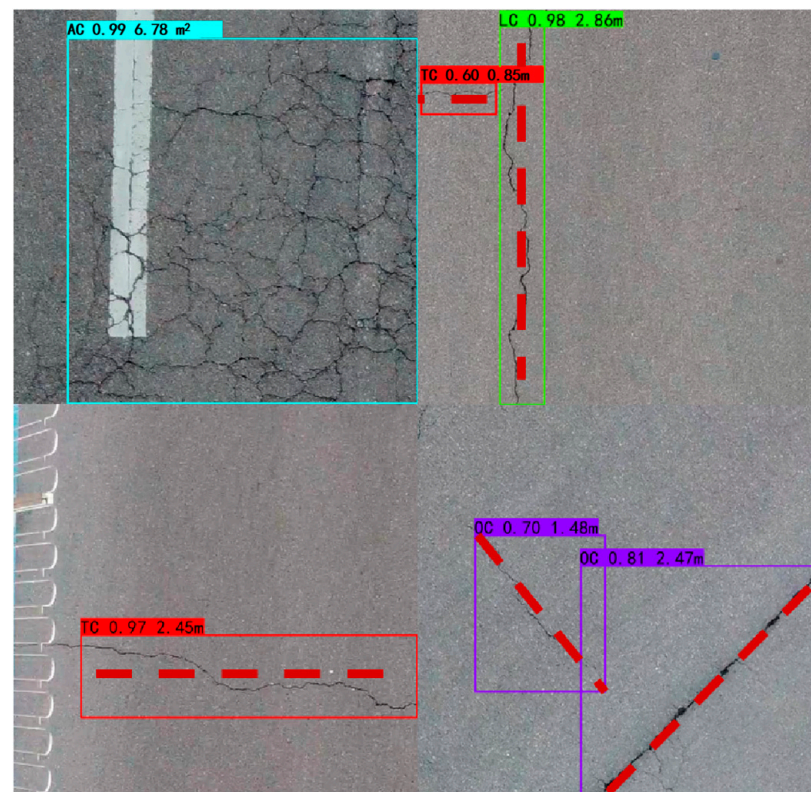
### 6.1. Measurement Methods of Pavement Cracks

The measurement methods for crack analysis play a crucial role in statistically analyzing the quantity of cracks. These methods consider various factors, such as crack location, crack type, crack length, crack width, crack depth, and crack area. In order to improve the



practicality of these methods in road damage maintenance, the quantity of cracks can be roughly estimated, temporarily excluding small cracks.

- (i) Pavement Crack Location: The pixel position of the detected crack in the original UAV imagery can be determined based on the corresponding image number; meanwhile, the actual ground position can be inferred through *GSD* calculation.
- (ii) Pavement Crack Length: This can be determined based on the pixel size of the confidence frame model, as illustrated in Figure 11. Horizontal cracks are measured by their horizontal border pixel lengths; vertical cracks by their vertical border pixel lengths; diagonal cracks by estimated border diagonal distance pixels; and mesh cracks primarily by measuring border pixel areas.
- (iii) Pavement Crack Width: The maximum width of a crack can be determined by identifying the region with the highest concentration of extracted crack pixels.
- (iv) Pavement Crack Area: This mainly aims at alligator cracks (AC), with a measurement of the crack area. It can be calculated by the pixels of AC based on the confidence frame model.



**Figure 11.** Diagram of crack measurements.

Finally, to determine the location, length ( $L$ ), and area ( $A$ ) of road cracks with ground truth, the quantitative results of cracks can be determined by multiplying the ground sampling distance ( $GSD$ , Unit: cm/pixel) by the pixels at which they are located. The actual length or width of the crack in meters can be calculated as the pixel length (m)  $\times GSD/100$ , while the actual area of the block affected by the crack in square meters can be derived from the pixel area (m<sup>2</sup>)  $\times GSD^2/100^2$ .

## 6.2. Evaluation Methods of Pavement Distress

The evaluation of pavement damage can be determined using the internationally recognized pavement damage index ( $PCI$ ), which is also adopted in China. The  $PCI$  provides a crucial indicator for assessing the level of pavement integrity. Additionally, the pavement damage rate ( $DR$ ) represents the most direct manifestation and reflection

of the physical properties related to the pavement condition. In this study, we refer to specifications such as ‘Technical Code of Maintenance for Urban Road (CJJ36-2016)’ [33] and ‘Highway Performance Assessment Standards (DB11/T1614-2019)’ [34] from the Chinese government, incorporating their respective calculation formulas as follows:

$$DR = 100 \times \sum_{i=1}^N w_i A_i / A \tag{6}$$

$$PCI = 100 - a_0 DR^{a_1} \tag{7}$$

where  $A_i$  is the damage area of the pavement of the  $i$ th crack type ( $m^2$ );  $N$  is the total number of damage types, taken here as 4;  $A$  is the pavement area of the investigated road section (the investigated road length multiplied by the effective pavement width,  $m^2$ ); and  $w_i$  is the damage weight of the pavement of the  $i$ th crack type, directly set as 1. According to the ‘‘Highway Performance Assessment Standards (DB11/T 1614-2019)’’ [34],  $a_0$  and  $a_1$  represent the material coefficients of the pavement, in which asphalt pavement is taken as  $a_0 = 10$  and  $a_1 = 0.4$ , while concrete pavement is taken as  $a_0 = 9$  and  $a_1 = 0.42$ . It is evident that a higher  $DR$  leads to a lower  $PCI$  value, indicating poorer pavement integrity.

### 6.3. Visualization Results of Pavement Distress

The original frame image was utilized for crack detection in this experiment, as visualized in Figure 12. On the right side, the statistical results of the four types of crack measurement are presented. This study employed the preferred Faster-RCNN trained model with a remarkable detection accuracy of 87.2% ( $mAP$ ). By conducting crack measurement and statistics on a regional road section, a damage rate ( $DR$ ) of 29.5% and pavement damage index ( $PCI$ ) of 61.28 were calculated, indicating a medium rating for the road section integrity in this region.

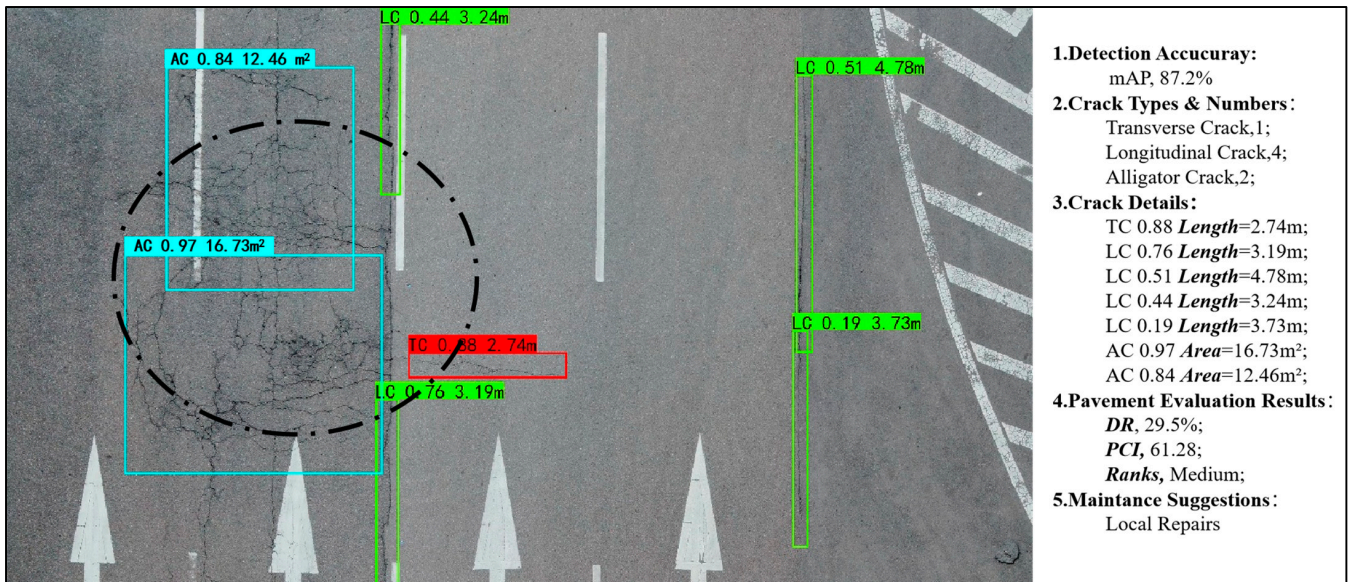


Figure 12. Visualization results of crack detection and pavement distress evaluation.

## 7. Discussion

This study proposes a comprehensive and systematic framework and method for automatic crack detection and pavement distress evaluation in a UAV inspection system. The framework begins by establishing the flight parameter settings and experimental techniques to enhance the high-quality imagery using the DJI Min2 drone in real-world scenarios. Additionally, a benchmark dataset was created and has been made available to the community. The dataset includes important information such as the  $GSD$ , which is

essential for evaluating pavement distress. In this experiment, our self-made crack dataset demonstrated its superiority compared to existing datasets used in similar algorithms, achieving the highest accuracy in crack recognition and algorithmic efficiency. The experimental result (refer to Table 6) revealed the significance of data acquisition quality in the accuracy of crack target recognition, with high-quality image data from the UAV imagery effectively improving the recognition accuracy.

In this experiment, the detection capability for road cracks in a UAV inspection system could be enhanced through a range of strategies. Firstly, adhering to a drone flight control strategy ensured a consistent high and stable speed during data acquisition on urban roads. This guaranteed the collection of clear and high-quality drone images with attached real spatial scale information for distress assessments. Secondly, the sampling ‘divide and conquer’ strategy for model training and target detection involves various key steps, including ‘the frame extracting from video and image cropping for large image’ and ‘model learning and crack detection for small images’, as well as ‘fusion and splicing from small images’. This approach effectively improves the accuracy of identifying cracks in large-scale images while enhancing the operational efficiency of these models. Thirdly, the deployment of drone detection algorithms using both ‘online–offline’ and ‘online–online’ strategies provides flexibility based on different scenarios. The ‘one-stage’ algorithm operates quickly, but has a lower detection accuracy, whereas the ‘two-stage’ algorithm exhibits a slower running efficiency but a higher detection accuracy. These deep learning models can be deployed accordingly, depending on the specific application scenarios. For instance, in sudden situations requiring fast real-time detection, lightweight deployment using a ‘two-stage’ algorithm such as YOLO series models can be employed.

To propose a suitable deployment scheme for the UAV inspection system, this study utilized prominent deep learning algorithms, namely Faster-RCNN, YOLOv5s, YOLOv7-tiny, and YOLOv8s, for pavement crack object detection and a comparative analysis. The results revealed that Faster-RCNN demonstrated the best overall performance, with a precision ( $P$ ) of 75.6%, a recall ( $R$ ) of 76.4%, an  $F1$ -score of 75.3%, and a mean Average Precision ( $mAP$ ) of 79.3%. Moreover, the  $mAP$  of Faster-RCNN surpassed that of YOLOv5s, YOLOv7-tiny, and YOLOv8s by 4.7%, 10%, and 4%, respectively. This indicates that Faster-RCNN outperformed in terms of detection accuracy, but required a higher environment configuration, making it suitable for online data collection using a UAV and offline inspection at work stations. On the other hand, the YOLO serial models, while slightly less accurate, were the fastest algorithms and are suitable for the lightweight deployment of UAVs with online collection and real-time inspection. Many studies have also proposed refined YOLO-based algorithms for crack detection in drones, mainly due to their lightweight deployment in UAV systems. For instance, the BL-YOLOv8 model [20] reduces both the number of parameters and computational complexity compared to the original YOLOv8 model and other YOLO serial models. This offers the potential to directly deploy the YOLO serial models on cost-effective embedded devices or mobile devices.

Finally, road crack measurement methods are presented to assess road damage, which will enhance the application of the UAV inspection system and provide factual evidence for the maintenance decisions made by road authorities. Notably, a crack is a significant indicator for evaluating road distress. In this study, the evaluation results were primarily obtained through a comprehensive assessment of the crack area, degree of damage, and their proportions. However, relying solely on cracks to determine road distress may be deemed limited, and this should only be considered as a reference for the relevant road authorities. Therefore, it is essential to conduct a comprehensive evaluation that takes into account multiple factors, such as rutting and potholes.

## 8. Conclusions

The traditional manual inspection of road cracks is inefficient, time-consuming, and labor-intensive. Additionally, using multifunctional road inspection vehicles can be expensive. However, the use of UAVs equipped with high-resolution vision sensors offers



a solution. These UAVs can remotely capture and display images of the pavement from high altitudes, allowing for the identification of local damages such as cracks. The UAV inspection system, which is based on the commercial DJI Min2 drone, offers several advantages. It is cost-effective, non-contact, highly precise, and enables remote visualization. As a result, it is particularly well-suited for remote pavement detection. In addition, automatic crack detection technology based on deep learning models brings significant additional value to the field of road maintenance and safety. It can be integrated into the commercial UAV system, thereby reducing the workload of maintenance personnel.

In this study, the contributions are summarized as follows: (1) A pavement crack detection and evaluation framework of a UAV inspection system based on deep learning was proposed and can provide technical guidelines for road authorities. (2) To enhance automatic crack detection capability and design a suitable scheme for implementing deep-learning-based models in a UAV inspection system, we conducted a validation and comparative study on prevalent deep learning algorithms for detecting pavement cracks in urban road scenarios. The study demonstrates the robustness of these algorithms in terms of their performance and accuracy, as well as their effectiveness in handling our customized crack image datasets and other popular crack datasets. Furthermore, this research provides recommendations for leveraging UAVs in deploying these algorithms. (3) Quantitative methods for road cracks were proposed and pavement distress evaluations were also carried out in our experiment. Obviously, our final evaluation results were also guaranteed according to *GSD*. (4) A pavement crack image dataset integrated with *GSD* was established and has been made publicly available for the research community, serving as a valuable supplement to existing crack databases.

In summary, the UAV inspection system, under the guidance of our proposed framework, has been proven to be feasible, yielding more satisfactory results. However, drone inspection has the inherent limitation of a limited battery life, making it difficult to perform long-distance continuous road inspection tasks. Drones are better suited for short-distance inspections in complex urban scenarios [16]. With advancements in drone and vision computer technology, drones equipped with lightweight sensors and these lightweight crack detection algorithms are expected to gain popularity for road distress inspection. In the future, this study aims to incorporate improved YOLO algorithms into the UAV inspection system to enhance road crack recognition accuracy. Furthermore, in order to establish a comprehensive UAV inspection system for road distress, we plan to continue researching multi-category defect detection systems in the future, including various road issues such as rutting and potholes, among which are cracks. Additionally, efforts will be made to enhance UAV flight autonomy for stability and high-speed aerial photography, further improving the quality of aerial images and catering to the requirements of various complex road scenarios.

**Author Contributions:** Conceptualization, X.C., C.L. and L.C.; methodology, X.C. and L.C.; software, C.L. and L.C.; validation, C.L., X.Z. and L.C.; formal analysis, X.C., X.Z. and Y.Z.; investigation, L.C., C.L., C.W. and Y.Z.; resources, X.C. and C.L.; data curation, C.L. and C.W.; writing—original draft preparation, X.C., C.L. and Y.Z.; writing—review and editing, X.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by China Postdoctoral Science Foundation (2017M622577), Hunan Provincial Natural Science Foundation (2018JJ2118). Chinese national college students innovation and entrepreneurship training program (S202310534031, S202310534169).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The UAV crack dataset presented in this study are openly available in FigShare at 10.6084/m9.figshare.25103138.

**Acknowledgments:** The authors would like to express many thanks to all the anonymous reviewers.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Samadzadegan, F.; Dadrass Javan, F.; Hasanlou, M.; Gholamshahi, M.; Ashtari Mahini, F. Automatic Road Crack Recognition Based on Deep Learning Networks from UAV Imagery. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2023**, *X-4/W1-2022*, 685–690. [CrossRef]
2. 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*, 103991. [CrossRef]
3. Cao, J.; Yang, G.T.; Yang, X.Y. Pavement Crack Detection with Deep Learning Based on Attention Mechanism. *J. Comput. Aided Des. Comput. Graph.* **2020**, *32*, 1324–1333.
4. Qi, S.; Li, G.; Chen, D.; Chai, M.; Zhou, Y.; Du, Q.; Cao, Y.; Tang, L.; Jia, H. Damage Properties of the Block-Stone Embankment in the Qinghai–Tibet Highway Using Ground-Penetrating Radar Imagery. *Remote Sens.* **2022**, *14*, 2950. [CrossRef]
5. Guo, S.; Xu, Z.; Li, X.; Zhu, P. Detection and Characterization of Cracks in Highway Pavement with the Amplitude Variation of GPR Diffracted Waves: Insights from Forward Modeling and Field Data. *Remote Sens.* **2022**, *14*, 976. [CrossRef]
6. Salman, M.; Mathavan, S.; Kamal, K.; Rahman, M. Pavement crack detection using the Gabor filter. In Proceedings of the 16th international IEEE Conference on Intelligent Transportation Systems (ITSC 2013), The Hague, The Netherlands, 6–9 October 2013; pp. 2039–2044. [CrossRef]
7. Ayenu-Prah, A.; Attoh-Okine, N. Evaluating pavement cracks with bidimensional empirical mode decomposition. *EURASIP J. Adv. Signal Process.* **2008**, *2008*, 861701. [CrossRef]
8. Majidifard, H.; Adu-Gyamfi, Y.; Buttlar, W.G. Deep machine learning approach to develop a new asphalt pavement condition index. *Constr. Build. Mater.* **2020**, *247*, 118513. [CrossRef]
9. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
10. Tran, V.P.; Tran, T.S.; Lee, H.J.; Kim, K.D.; Baek, J.; Nguyen, T.T. One stage detector (RetinaNet)-based crack detection for asphalt pavements considering pavement distresses and surface objects. *J. Civ. Struct. Health Monit.* **2021**, *11*, 205–222. [CrossRef]
11. Xiao, L.Y.; Li, W.; Yuan, B.; Cui, Y.Q.; Gao, R.; Wang, W.Q. Pavement Crack Automatic Identification Method Based on Improved Mask R-CNN Model. *Geomat. Inf. Sci. Wuhan Univ.* **2022**, *47*, 623–631. [CrossRef]
12. Xu, K.; Ma, R.G. Crack detection of asphalt pavement based on improved faster RCNN. *Comput. Syst. Appl.* **2022**, *31*, 341–348. [CrossRef]
13. Xu, X.; Zhao, M.; Shi, P.; Ren, R.; He, X.; Wei, X.; Yang, H. Crack detection and comparison study based on faster R-CNN and mask R-CNN. *Sensors* **2022**, *22*, 1215. [CrossRef]
14. Yan, K.; Zhang, Z. Automated asphalt highway pavement crack detection based on deformable single shot multi-box detector under a complex environment. *IEEE Access* **2021**, *9*, 150925–150938. [CrossRef]
15. Yokoyama, S.; Matsumoto, T. Development of an automatic detector of cracks in concrete using machine learning. *Procedia Eng.* **2017**, *171*, 1250–1255. [CrossRef]
16. Jiang, Y.T.; Yan, H.T.; Zhang, Y.R.; Wu, K.Q.; Liu, R.Y.; Lin, C.Y. RDD-YOLOv5: Road Defect Detection Algorithm with Self-Attention Based on Unmanned Aerial Vehicle Inspection. *Sensors* **2023**, *23*, 8241. [CrossRef] [PubMed]
17. Zhang, Y.; Zuo, Z.; Xu, X.; Wu, J.; Zhu, J.; Zhang, H.; Wang, J.; Tian, Y. Road damage detection using UAV images based on multi-level attention mechanism. *Autom. Constr.* **2022**, *144*, 104613. [CrossRef]
18. Zhou, Q.; Ding, S.; Qing, G.; Hu, J. UAV vision detection method for crane surface cracks based on Faster R-CNN and image segmentation. *J. Civ. Struct. Health Monit.* **2022**, *12*, 845–855. [CrossRef]
19. Xiang, X.; Hu, H.; Ding, Y.; Zheng, Y.; Wu, S. GC-YOLOv5s: A Lightweight Detector for UAV Road Crack Detection. *Appl. Sci.* **2023**, *13*, 11030. [CrossRef]
20. Wang, X.; Gao, H.; Jia, Z.; Li, Z. BL-YOLOv8: An Improved Road Defect Detection Model Based on YOLOv8. *Sensors* **2023**, *23*, 8361. [CrossRef] [PubMed]
21. Omobamije, O.; Omoniyi, T.M.; Musa, A.; Duna, S. An improved deep learning convolutional neural network for crack detection based on UAV images. *Innov. Infrastruct. Solut.* **2023**, *8*, 236. [CrossRef]
22. Zhao, Y.; Zhou, L.; Wang, X.; Wang, F.; Shi, G. Highway Crack Detection and Classification Using UAV Remote Sensing Images Based on CrackNet and CrackClassification. *Appl. Sci.* **2023**, *13*, 7269. [CrossRef]
23. Liu, K. Learning-based defect recognitions for autonomous uav inspections. *arXiv* **2023**, arXiv:2302.06093v1.
24. Zou, Z.; Chen, K.; Shi, Z.; Guo, Y.; Ye, J. Object detection in 20 years: A survey. *Proc. IEEE* **2023**, *111*, 257–276. [CrossRef]
25. Bubbliiing. Faster-RCNN-PyTorch[CP]. 2023. Available online: <https://github.com/bubbliiing/faster-rcnn-pytorch> (accessed on 26 January 2024).
26. Ultralytics. YOLOv5[CP]. 2020. Available online: <https://github.com/ultralytics/yolov5> (accessed on 26 January 2024).
27. Wong, K.Y. YOLOv7[CP]. 2023. Available online: <https://github.com/WongKinYiu/yolov7> (accessed on 26 January 2024).
28. Ultralytics. YOLOv8[CP]. 2023. Available online: <https://github.com/ultralytics/ultralytics> (accessed on 26 January 2024).
29. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 1137–1149. [CrossRef]



30. Buda, M.; Maki, A.; Mazurowski, M.A. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Netw.* **2018**, *106*, 249–259. [[CrossRef](#)]
31. Sami, A.A.; Sakib, S.; Deb, K.; Sarker, I.H. Improved YOLOv5-Based Real-Time Road Pavement Damage Detection i-n Road Infrastructure Management. *Algorithms* **2023**, *16*, 452. [[CrossRef](#)]
32. Faramarzi, M. Road damage detection and classification using deep neural networks (YOLOv4) with smartphone images. *SSRN* **2020**. [[CrossRef](#)]
33. *CJJ36-2016*; Technical Code of Maintenance for Urban Road. Ministry of Housing and Urban-Rural Development of the People's Republic of China: Beijing, China, 2017. Available online: [https://www.mohurd.gov.cn/gongkai/zhengce/zhengcefilelib/201702/20170228\\_231174.html](https://www.mohurd.gov.cn/gongkai/zhengce/zhengcefilelib/201702/20170228_231174.html) (accessed on 10 May 2023).
34. *JTG 5210-2018*; Highway Performance Assessment Standards. Ministry of Transport of the People's Republic of China: Beijing, China, 2018. Available online: [https://xxgk.mot.gov.cn/2020/jigou/glj/202006/t20200623\\_3313114.html](https://xxgk.mot.gov.cn/2020/jigou/glj/202006/t20200623_3313114.html) (accessed on 10 May 2023).

**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.