

International Journal of Intelligent Systems Technologies and Applications

ISSN online: 1740-8873 - ISSN print: 1740-8865
<https://www.inderscience.com/ijista>

Detection of waterlogging in urban road traffic based on improved YOLOv5-seg and ellipse fitting algorithm

Jianqiang Liu, Rui Chen, Xiaoyan Zhao, Xingyao Li, Yujie Shang, Peng Geng

DOI: [10.1504/IJISTA.2025.10069443](https://doi.org/10.1504/IJISTA.2025.10069443)

Article History:

Received:	18 July 2024
Last revised:	27 September 2024
Accepted:	07 November 2024
Published online:	30 September 2025

Detection of waterlogging in urban road traffic based on improved YOLOv5-seg and ellipse fitting algorithm

Jianqiang Liu, Rui Chen, Xiaoyan Zhao,
Xingyao Li, Yujie Shang and Peng Geng*

School of Information and Communication Engineering,

Nanjing Institute of Technology,

Nanjing, 211167, China

Email: 19895597781@163.com

Email: chenrui@njit.edu.cn

Email: zhaoxiaoyan@njit.edu.cn

Email: 2434090340@qq.com

Email: 806387795@qq.com

Email: gengpeng@njit.edu.cn

*Corresponding author

Abstract: This article proposes an innovative method for acquiring precise waterlogging depth data utilising images from traffic surveillance systems. Initially, the YOLOv5 algorithm identifies the vehicle type and determines its tire specifications accordingly. Subsequently, an enhanced version of the YOLOv5-seg model segments and masks the tire instances, while an ellipse fitting algorithm extracts the geometric parameters of the submerged tires to shape a complete ellipse. With the vehicle tires as benchmarks, a mathematical model for waterlogging depth is formulated, which computes the depth using crucial parameters from the ellipse. The experimental outcomes demonstrate that this algorithm achieves an average localisation accuracy of 96.4%, a mask segmentation accuracy of 95.6%, and maintains a detection error within 5 cm for 90% of the waterlogged depths measured. These findings confirm that the image-based tire detection method for waterlogging measurement is both effective and practical.

Keywords: waterlogging depth detection; deep learning; ellipse fitting; YOLOv5-seg.

Reference to this paper should be made as follows: Liu, J., Chen, R., Zhao, X., Li, X., Shang, Y. and Geng, P. (2025) 'Detection of waterlogging in urban road traffic based on improved YOLOv5-seg and ellipse fitting algorithm', *Int. J. Intelligent Systems Technologies and Applications*, Vol. 23, No. 3, pp.272–294.

Biographical notes: Jianqiang Liu is an undergraduate majoring in Information Engineering in Nanjing Institute of Technology. During the school, he has been participating in the teacher's scientific research projects. His major research interests include deep learning and image processing.

Rui Chen received her BE and ME from Southeast University, Nanjing, China, in 1991 and 1996, respectively. She received PhD from the Nanjing University of Post and Telecommunications in 2013. Her major research interests include deep learning and image processing.

Xiaoyan Zhao received her BE and PhD from the Southeast University, Nanjing, China, in 2008 and 2014, respectively. She is currently working as a Lecturer at Nanjing Institute of Technology. Her major research interests include signal processing and deep learning.

Xingyao Li is an undergraduate majoring in Information Engineering in Nanjing Institute of Technology. During the school, he has been participating in the teacher's scientific research projects. His major research interests include deep learning and image processing.

Yujie Shang is an undergraduate majoring in Information Engineering in Nanjing Institute of Technology. During the school, he has been participating in the teacher's scientific research projects. His major research interests include deep learning and image processing.

Peng Geng received his Master's in Communication and Information System in 2007 from the Guilin University of Electronic Science and Technology, Guilin, Guangxi, China. Currently, he is an Associate Professor of Nanjing Institute of Technology. His research interests include complex systems and deep learning.

1 Introduction

Amidst the escalating impacts of global climate change, there has been a marked escalation in the incidence and intensity of extreme precipitation events across various urban areas. This trend has resulted in significant disruptions to transportation networks and inundation of residential and public infrastructure, posing substantial threats to urban safety and the daily livelihoods of residents (Lin et al., 2018). In tandem with these climatic shifts, the pace of urbanisation has surpassed the capacity to develop sophisticated urban drainage systems. The consequence has been an expansion of impervious surfaces, thereby amplifying the susceptibility to urban flooding and its attendant challenges (Zhang et al., 2012). To address these issues proactively, it is imperative to implement robust monitoring of urban road traffic flooding. Such monitoring will not only mitigate the risks associated with flood events but also minimise the potential hazards to public travel safety and urban traffic flow (Jiang et al., 2019). This paper, therefore, proposes a novel approach to urban floodwater monitoring, emphasising the integration of advanced deep learning techniques to enhance the precision and efficiency of flood detection and measurement.

Currently, there are three main methods for urban flood depth detection: manual measurement, obtaining scale data from images with water level lines, and monitoring the water level through sensors (Liu et al., 2015). The first of these methods relies on manual measurements in flood-prone areas, with the advantage of being intuitive and flexible, and the disadvantage of being labour-intensive, inefficient, and posing certain safety hazards to detectors. The second method requires special images with water level lines, while only a few detection images have labelled water level lines, and the dataset of labelled water level line images is very small, so this method cannot be widely used. The third is to rely on the deployment of water level sensors to realise the real-time acquisition and transmission of waterlogging data, the advantage of which is the real-time acquisition and transmission of measurement data, the advantage of high measurement

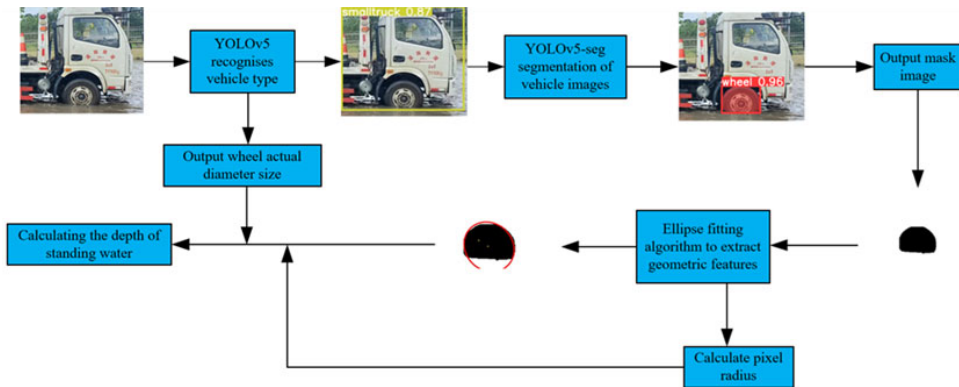
accuracy, convenient and fast, the disadvantage of the instrument is the high cost, the need for regular maintenance, and easy to be subjected to external interference, it is difficult to comprehensively carry out the work of urban traffic and road waterlogging detection, and it is difficult to meet the needs of urban road traffic. The limitations of the above three methods are mainly focused on efficiency, cost, safety and applicability. These limitations make them difficult to meet the comprehensive needs of urban flood monitoring in terms of real-time, accuracy and economy.

With the continuous development of deep learning, it also provides new ideas for waterlogging detection. In this technology, people focus on two issues. One is what object to use as a reference? The other is what algorithm to use for object detection? Traditional waterlogging detection selects some common objects in the city for target detection, so as to serve as a reference for measuring the depth of waterlogging, including but not limited to: garbage cans, houses, and sidewalk fences. However, in highly urbanised areas, vehicles are the most common reference, and the fixed tire sizes of the vehicles can serve as a waterlogging depth measure. Therefore, in this article, we use wheels as the reference for target detection. Common target detection algorithms can be categorised into single-stage algorithms represented by YOLO series and two-stage algorithms based on convolutional neural network (CNN), such as R-CNN (Ren et al., 2016), fast-RCNN (Zhang et al., 2024), faster-RCNN (Dollar et al., 2011), and mask-RCNN (He et al., 2017). In urban waterlogging detection application scenarios, there is a high demand for low complexity and real-time performance. Therefore, the YOLO series is more appropriate. Among them, the lightweight characteristics of YOLOv5 (Dong et al., 2022) make it easy to deploy on embedded devices. This article selects it as the object detection algorithm.

As shown in Figure 1, the proposed waterlogging depth measuring involves three tasks: identifying vehicles, detecting and segmenting tires, and calculating the waterlogging depth. Vehicle tires are accurately identified and segmented through deep learning technology, and then the geometric parameters of tires covered by water are extracted. Taking the vehicle tires as a reference, a mathematical model is constructed using ellipse fitting algorithm, so as to accurately calculate the waterlogging depth. The proposed method not only provides new technical ideas for urban flood monitoring, but also is expected to provide important technical support and decision-making basis for disaster emergency management and urban flood prevention and drainage.

The main contributions of this article are as follows:

- 1 Employs the lightweight Ghostnet architecture in place of the conventional YOLOv5-seg network. This substitution results in a substantial decrease in computational load and parameter count, subsequently enhancing the model's inference speed to better align with the demands of real-time detection capabilities.
- 2 Leveraging urban road surveillance cameras, this article integrates an enhanced YOLOv5-seg for vehicle tire detection with an ellipse fitting algorithm to extract comprehensive geometrical parameters of the tires. The integration of computer vision techniques offers a breakthrough from the limitations of traditional sensing devices, which are not only costly but also impractical for large-scale deployment. This innovative approach presents a novel solution for urban waterlogging monitoring, providing a more efficient and cost-effective method for assessing road conditions during flood events.

Figure 1 Framework of the proposed method (see online version for colours)

The rest of the article is structured as follows. The related work is presented in Section 2. Section 3 describes the mask extraction of vehicle tires using an improved YOLOv5-model structure. Section 4 describes the proposed waterlogging depth detection method, including the wheel geometric feature parameter extraction and the mathematical model construction based on ellipse fitting algorithm. Section 5 presents the testing and calculation results of the proposed algorithm. Finally, conclusions as well as future work are drawn in Section 6.

2 Related work

The urban waterlogging detection research is moving towards integration and intelligence, and the integration of deep learning, computer vision, and mathematical modelling technologies has brought new technological breakthroughs and application prospects to the field.

In the aspect of tire detection and segmentation, the YOLO series of algorithms, since the initial proposal of the YOLOv1 model by Redmon et al. in 2016 (Redmon et al., 2016), has significantly improved detection speed by transforming the object detection problem into a regression issue. The subsequent YOLOv2 introduced batch normalisation (BN) and a high-resolution classifier, further enhancing model performance (Redmon and Farhadi, 2017). YOLOv3 incorporated a feature pyramid network (FPN) and multi-scale predictions, significantly improving the detection of small objects (Al-Haija et al., 2021; Farhadi and Redmon, 2018). YOLOv4 built upon YOLOv3 by integrating various optimisation strategies, such as using CSPDarknet53 as the backbone network, to further enhance detection performance and accuracy (Lin et al., 2017; Wang et al., 2020). YOLOv5 iterated on YOLOv4, improving recognition speed and accuracy, making it a strong candidate for real-world engineering applications (Dong et al., 2022). YOLOv6 (Li et al., 2022) and YOLOv7 (Wang et al., 2023) further optimised upon previous versions, offering higher detection accuracy and speed. The latest YOLOv8 (Sohan et al., 2024) offers five scaled versions and supports multiple vision tasks such as object detection, segmentation, pose estimation, tracking, and classification, achieving excellent performance in both accuracy and speed.

In the aspect of waterlogging depth monitoring, in addition to traditional sensor-based methods, computer vision-based methods are gaining attention. Huang et al. (2020) constructed a waterlogging detection model based on Mask-RCNN, using tires as a reference, and calculated road waterlogging depth using the height difference method and the Pythagorean theorem. However, due to the large differences in tire sizes among different vehicles, this method has limitations in practical applications. Meng et al. (2019) used the human body as a reference to build a detection model, but due to differences in height, the accuracy of waterlogging depth measurement was affected. Chaudhary et al. (2019) used vehicles and houses as references and classified the proportion of flooded references into 11 categories to roughly estimate the proportion of waterlogging, but they could not accurately calculate the actual depth of waterlogging.

In the realm of ellipse fitting algorithm research, a method combining algebraic distance with the RANSAC algorithm has been proposed in Barath and Matas (2021), which pre-processes data through linear transformation and minimises the Euclidean distance between the ellipse and discrete points, thereby enhancing noise resistance and fitting accuracy. Addressing scenarios where multiple ellipses either do not intersect or intersect with each other, Scitovski et al. (2023) presents an extension of the hybrid ellipse fitting algorithm to accommodate multiple ellipses, introducing a novel globally search-based method for fitting concentric ellipses that adheres to the constraints required for concentricity. Cao and Li (2017) suggests an optimisation algorithm for ellipse fitting based on the Letts criterion, which aims to mitigate the interference of isolated points on fitting through iterative approaches and employs an extended Kalman filter to achieve more precise ellipse fitting. Furthermore, Mei et al. (2021) tackles the challenge of fitting ellipses to weak signals by proposing an improved algorithm capable of accurately demodulating weak vibration signals, subsequently improving detection resolution.

Inspired by these studies, this article proposes a new method, which first distinguishes between vehicle types, and divides the size of wheels according to different vehicle types, so as to make the measurement standard more diversified, and to a certain extent, reduce the error caused by the calculation using a single wheel size, and then use YOLOv5-seg to segment the wheels by instances, and extract the mask image of the wheels, and then use the ellipse detection algorithm, the geometric parameters of the wheel are calculated, and the mathematical model of waterlogging depth is constructed so as to calculate the waterlogging depth.

3 Mask extraction deep learning network of tires

3.1 YOLOv5-seg model structure

YOLOv5-seg represents a significant evolution over the YOLOv5-s model by incorporating image segmentation capabilities alongside its robust object detection functionality. YOLOv5-s itself is a sophisticated deep learning model, engineered for high-efficiency and precision in object detection. It achieves this through a refined network architecture and sophisticated inference mechanisms that maintain accuracy while accelerating processing speed. Building upon this foundation, YOLOv5-seg introduces a segmentation head, enabling the model to perform concurrent object detection and image segmentation, as depicted in Figure 2.

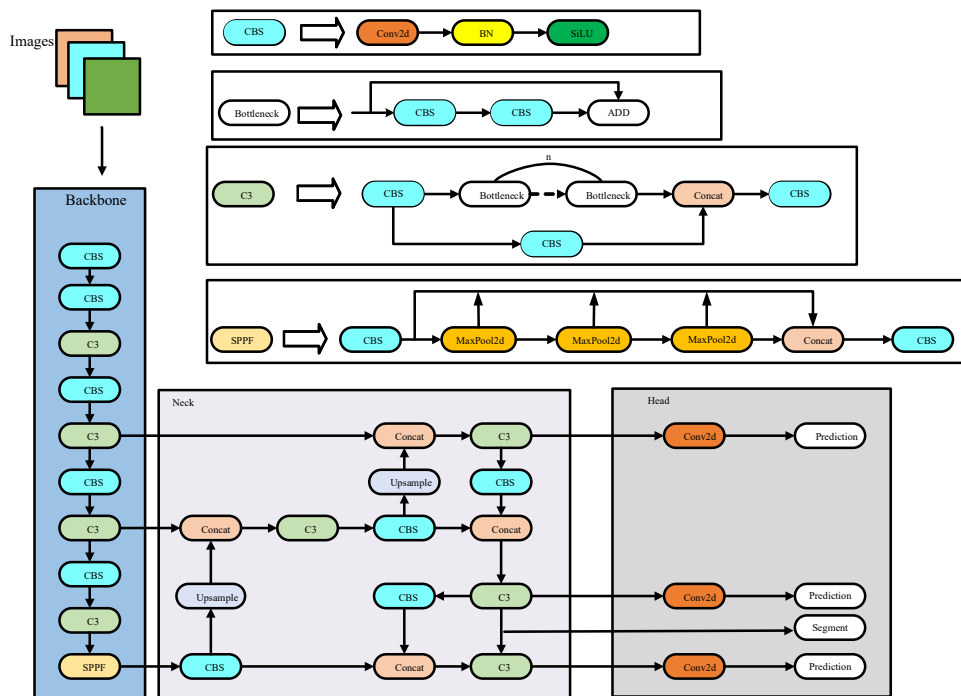
The architecture of YOLOV5-seg is meticulously organised into four core components: the input, backbone, neck, and head (Shi et al., 2021). The input phase is dedicated to data acquisition and pre-processing, employing techniques such as adaptive anchoring, image scaling, and Mosaic data augmentation to standardise the input for optimal processing by the subsequent layers.

The backbone, which plays a pivotal role in feature extraction, consists of three advanced modules: CBS (Conv + BatchNorm + SiLu), C3, and spatial pyramid pooling fast (SPPF) (Zhang et al., 2023). These modules are meticulously designed to work in unison, efficiently capturing and refining the critical features from the input images.

The neck of the model, which is crucial for feature fusion, utilises an FPN + PAN (feature pyramid network + pyramid attention network) structure (Yang et al., 2017). It adeptly performs down-sampling and up-sampling operations, effectively reducing the network's complexity while simultaneously enhancing the richness of feature representation.

The final stage, the head component, receives the feature maps from the neck and executes the ultimate predictions, culminating in the output of detection results. This integrated structure endows YOLOV5-seg with exceptional performance in both detection and segmentation tasks, solidifying its position as a formidable tool in the realm of computer vision.

Figure 2 YOLOv5s-seg network structure diagram (see online version for colours)



YOLOv5-seg uses three loss functions in the head part, which are classification loss, localisation loss, and confidence loss. The classification loss evaluates the model's accuracy in identifying and classifying image objects. It measures the discrepancy

between predicted and actual labels, driving the model to refine its categorisation capabilities throughout training; the localisation loss assesses the accuracy of the model's bounding box predictions compared to the actual boxes. It quantifies the positional and dimensional discrepancies, guiding the model to enhance its prediction accuracy for object locations and dimensions throughout training; the confidence loss evaluates the model's certainty in its predictions, comparing the predicted confidence scores against the actual presence of objects. This function aids in refining the model's capacity to accurately assess object existence and the precision of its bounding boxes throughout training. Both the classification loss function and the confidence loss function use the binary cross-entropy loss function as

$$L_{BCE} = -\frac{1}{n} \sum_x [y \ln a + (1-y) \ln(1-a)] \quad (1)$$

where x is the sample, y is the label score, a is the probability predicted by the model, and n is the number of samples.

The localisation loss function is

$$L_{IoU} = 1 - IoU + \frac{|C - (A \cup B)|}{|C|} \quad (2)$$

where A denotes the prediction box, B denotes the ground truth (GT), C is the smallest outer rectangle between the prediction box and GT, and $IoU = \frac{A \cap B}{A \cup B}$ is the intersection and concatenation of the predicted box compared to GT.

YOLOv5-seg incorporates an additional segmentation loss function known as L_{Focal} (Rezatofighi et al., 2019), which quantifies the variance between the model's predicted segmentation mask and the actual mask. The segmentation loss function is

$$L_{Focal} = -\alpha * (1-p)^\gamma * \log(p) \quad (3)$$

where α denotes the moderating factor for positive samples, p denotes the probability of positive sample, and γ is the weighting factor.

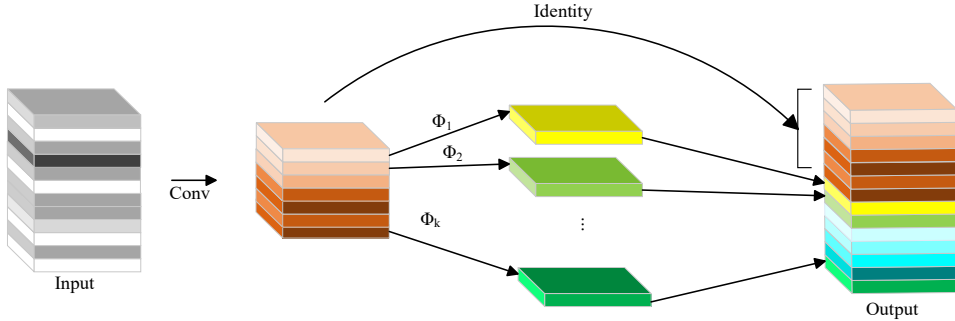
3.2 Improved YOLOv5-seg model

As shown in Figure 2, the backbone part uses the C3 module as one of the basic structures, which incorporates the diversion idea and residual structure of CSPNet. The C3 module contains three standard convolutional layers and multiple residual structures, which can be stacked to increase the nonlinear representation ability of the model and make it more suitable for handling complex detection tasks, but it also increases the computational complexity of the model, which leads to the trend of degradation of the performance in resource-limited environments.

In order to solve this problem, this article adopts the Ghostnet (Han et al., 2020) network to replace the backbone network structure. Ghostnet is a lightweight neural network architecture proposed by Huawei's Noah's Ark Lab team, which consists of multiple Ghost modules. Ghost module generates feature maps in two steps. As shown in Figure 3, few feature maps are firstly obtained by regular convolution, then more feature maps are obtained by layer-by-layer convolution of the feature maps obtained in the first

step through constant mapping. The final feature maps are combined by the features obtained in the two steps. This reduces the number of parameters and the amount of computation, improves the efficiency of the network, and makes it easier to deploy and apply under limited resources.

Figure 3 Ghost module structure (see online version for colours)



The first step of Ghost module calculation process is

$$Y' = Xf' \quad (4)$$

where Y' denotes the output feature map, X denotes the input feature map, and f' denotes convolution kernel.

The second step is

$$y_{i,j} = \Phi_{i,j}(y'_i) \quad i = 1, \dots, s \quad (5)$$

where $y_{i,j}$ denotes the j^{th} Ghost feature map generated from the i^{th} original feature map, $\Phi_{i,j}$ denotes the constant mapping of the feature map, and y'_i denotes the i^{th} feature map in Y' .

In the traditional YOLOv5-seg, the backbone portion of the network uses the CBS module as another network basic structure. In this article, we combine the Ghost module with the CBS module to form a new CBS module, the GhostConv module. GhostConv can reduce the number of parameters and computations to further reduce the network size. Its structure is shown in Figure 4.

The GhostBottleneck in Ghostnet is differentiated into two structural variants based on their step sizes, as depicted in Figure 5. In the GhostBottleneck configuration designated as step size 1, the process follows a familiar pattern to that of ResNet's residual blocks (He et al., 2016), where the feature map undergoes an initial upscaling prior to being downscaled. The GhostBottleneck is composed of a pair of nested Ghost modules, each with a distinct function. The initial Ghost module serves to expand the channel count, effectively broadening the feature representation. Conversely, the subsequent Ghost module is designed to condense the channel dimensions, aligning them with the shortcut connection. This ensures that the input and output of the GhostBottleneck are smoothly bridged by the shortcut, maintaining feature integrity across the module. Drawing inspiration from MobileNetV2 (Sandler et al., 2018), the GhostBottleneck's initial Ghost module incorporates BN (Ioffe and Szegedy, 2015) and a ReLU activation function to introduce non-linearity and stabilise training. However, the

second Ghost module omits the ReLU activation to prevent undue complexity. The step size 2 variant of the structure mirrors the step size 1, with the key distinction being the interposition of a depthwise convolution that operates on a stride of 2, enabling efficient spatial reduction between the Ghost modules.

Figure 4 GhostConv structure (see online version for colours)

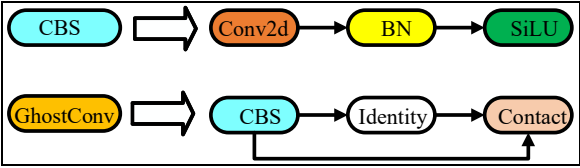


Figure 5 GhostBottleneck structure diagram

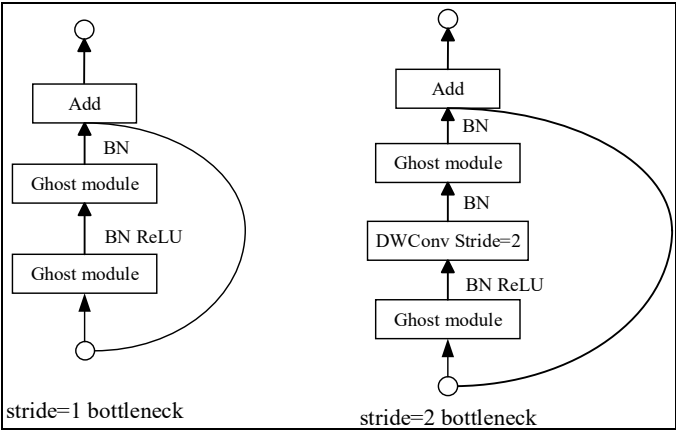


Figure 6 C3Ghost structure (see online version for colours)

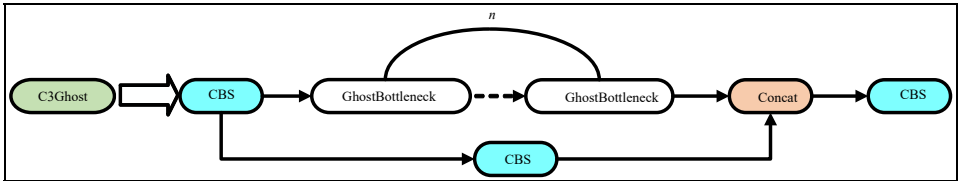
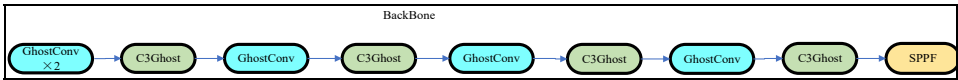


Figure 7 Ghostnet-YOLOv5-seg Backbone structure (see online version for colours)



In this article, GhostBottleneck is used to replace the residual module in the C3 module to form a new C3 module called C3Ghost, which can further reduce the number of parameters and the amount of computation, thus reducing the memory occupation and speeding up the inference. Its structure is shown in Figure 6.

The final Ghostnet-YOLOv5-seg Backbone that incorporates the YOLOv5-seg network structure of C3Ghost and GhostConv is shown in Figure 7.

4 Proposed waterlogging depth detection algorithm

4.1 Reference selection for the proposed algorithm

YOLO in image recognition is mainly used in the target object localisation and segmentation, and cannot directly detect the depth of standing water, so we need to select the object as the depth of standing water reference. The selection of the reference generally has the following characteristics.

- 1 Common daily, especially when the occurrence of waterlogged city roads appears frequently.
- 2 Easy to recognise, must be through the deep learning model can be more accurately recognised objects.
- 3 Regular shape, with more obvious geometric features, and able to calculate the submergence ratio according to the parameters of the shape features.
- 4 Fixed size, it is generally desirable to have a clearer size, so as to estimate the depth of waterlogging through the comparison of the actual size and the size in the image.

Based on the above criteria, this article takes the wheel as the identifier, the wheel has a high enough common degree of shape for the regular circle, and generally has a clear tire specification, suitable for the construction of the waterlogging depth calculation model as the object.

4.2 Dataset construction

In order to ensure the training effect of the model, a rich image dataset needs to be collected. The dataset sources in this article mainly include two aspects. In one part, the video is recorded through the road surveillance camera and the key frame images are extracted from it. Due to the limitation of recording time, the number of acquired images is relatively limited. The dataset enhancement technique is used to expand the dataset by rotating the images. In the other part, relevant image data is collected by itself. Similarly, in order to enhance the diversity and generalisation of the dataset, rotation operation is also performed on these images as well. As shown in Figure 8. After the above steps, a total of 3,900 images were finally obtained. Finally, the wheel data is labelled by Labelme to produce the training dataset. As shown in Figure 9.

4.3 Ellipse fitting algorithm

Ellipse fitting algorithm research is an important direction in the field of computer vision, scholars at home and abroad have so far proposed a variety of ellipse fitting methods, in this article, we use the high-quality ellipse detection ellipse fitting algorithm proposed by Lu et al. (2019), the core idea of the algorithm is to extract and analyse arc-supporting line segments in the image to simplify complex curves, while retaining general properties

including convexity and polarity, so as to realise ellipse fitting, the main steps are as follows.

Figure 8 Dataset (see online version for colours)



Figure 9 Labelme labelling (see online version for colours)



Step 1 Arc support segment extraction

The algorithm extracts arc-support line segments from the image, which represent local curves in the image that may belong to the ellipse. This step is based on the line segment detector (LSD) algorithm, which analyses the gradient information of the image to identify possible elliptical arcs.

Step 2 Arc support group formation

The algorithm forms arc support groups by iteratively and robustly linking neighboring arc support line segments. These groups represent potential elliptic curves and ensure that they belong to the same ellipse by considering the continuity and convexity of the line segments.

Step 3 Initial elliptic set generation

The algorithm uses two complementary approaches to generate the initial elliptic set; the local approach focuses on selecting arc support groups with high significance, while the global approach traverses all possible pairs of arc support groups. These two approaches combine the superposition principle of ellipse fitting with a set of geometric constraints to ensure that the initial ellipse sets generated are both comprehensive and accurate.

Step 4 Ellipse clustering

To deal with duplicate terms in the initial ellipse set, the algorithm devises a hierarchical clustering method based on mean drift. This method decomposes the 5-dimensional ellipse parameter space (including the ellipse centre, direction and semi-axis) into three subspaces and performs clustering on each of these three subspaces, which effectively compresses the ellipse candidate set and improves the efficiency of fitting.

Step 5 Verification and optimisation

In order to ensure the accuracy of ellipse fitting, the algorithm validates and optimises the fitting results, which includes the evaluation of the fitting quality of the ellipse and the verification of the ellipse geometrical properties to ensure that the detected ellipse is accurate and reliable.

4.4 Wheel geometric feature parameter extraction based on ellipse fitting algorithm

Given that the wheel is partially submerged by water, only the mutilated part of the wheel exposed to the water can be recognised. For this reason, this article uses the improved YOLOv5-seg model to recognise this part of the mutilated wheel image and outputs the corresponding mask image. Subsequently, the high-quality ellipse detection ellipse fitting algorithm is applied to extract the geometric feature parameters of these mask images, which in turn restores the complete wheel shape and calculates the submerged water depth. However, in practical application scenarios, the shooting angle of the surveillance camera may be directly facing the front of the wheel, or it may be deviated from the front. For these two different situations, this article adopts two processing methods respectively:

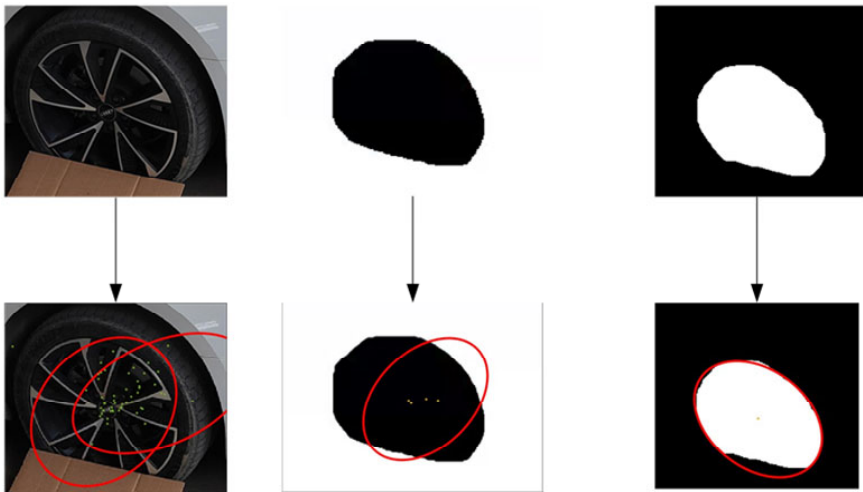
- 1 When the camera is facing the front of the wheel, the detection rectangular box inferred from the improved YOLOv5-seg model is first utilised to accurately represent the dimensions of the wheel, i.e., the length and height of the rectangular box correspond to the width and height of the wheel, respectively. Next, the high-quality ellipse detection ellipse fitting algorithm is applied to the improved

YOLOv5-seg output of the residual wheel mask image in order to fit the geometric feature parameters of the wheel to obtain the pixel radius of the wheel. The specific process is shown in Figure 10.

Figure 10 Positive wheel (see online version for colours)



Figure 11 Experimental comparison chart (see online version for colours)



- 2 When the camera is deviated from the front of the wheel, the detection rectangle box inferred by the improved YOLOv5-seg model cannot accurately reflect the actual size of the wheel. Therefore, firstly, it is necessary to detect the minimum outer rectangular box of the crippled wheel mask image using OpenCV. Secondly, in order to ensure the accuracy of wheel geometric parameter extraction, we conducted comparison experiments on the original wheel image, the mask image outputted by the improved YOLOv5-seg, and the wheel image binarised by OpenCV, respectively, which were all processed using the high-quality ellipse detection ellipse fitting algorithm. The experimental results are shown in Figure 11. Through comparison, it is found that applying the high-quality ellipse detection ellipse fitting algorithm to the original wheel image and the mask image output by the improved YOLOv5-seg is not satisfactory, and the long and short axes of the fitted ellipse cannot be used as the pixel radius of the wheel image. As for the wheel image processed by OpenCV binarisation, although its boundary features are not obvious, resulting in the actual mask image of the hub, but because the wheel and the hub have the same location of the centre of the circle, we can fit the location of the centre of the circle by the high-quality ellipse detection ellipse fitting algorithm, and equate it to the centre of the circle of the wheel. Therefore, in this article, we choose to

apply the high-quality ellipse detection ellipse fitting algorithm to the wheel image that has been binarised by OpenCV to obtain the centre of circle position of the wheel. Finally, the distance from the centre of the circle to the top edge of the smallest outer rectangular box is calculated using OpenCV, which is used as the pixel radius of the wheel image. The specific process is shown in Figure 12.

Figure 12 Failure to square the wheel (see online version for colours)



4.5 Mathematical modelling of waterlogging depth

The fitting parameters based on ellipse fitting are extracted by detecting the minimum outer rectangular frame of the wheel, assuming that the waterlogged surface is an equiaxial plane, and taking the bottom edge of the outer rectangle of the wheel mask image output from the improved YOLOv5-seg as the location of the water level line, and the height of the outer rectangle as the pixel height of the part of the wheel that is exposed to the water surface, h , as the perpendicular distance from the centre of the circle of the wheel after the wheel is fitted to the top of the minimum outer rectangle of the wheel mask image H_r , which can be regarded as the pixel radius length of the wheel, and the actual diameter of the wheel is H . The construction of the specific model is shown in Figure 13. Finally, the depth of standing water is calculated as

$$H_d = \left(1 - \frac{h}{2H_r}\right)H \quad (6)$$

where H_d is the detected type mask depth, i.e., the depth of water on the road surface; H is the actual type size of the wheel; h is the pixel height of the exposed part of the wheel, in terms of the height of the outer rectangle of the wheel mask image output from the improved YOLOv5-seg; H_r is the pixel radius of the wheel, in terms of the perpendicular distance from the centre of the fitted circle after ellipse fitting of the wheel image to the top edge of the smallest outer rectangle of the wheel mask.

From equation (6), it can be seen that in the case of obtaining the height h and the pixel radius H_r of the wheel exposed part of the water surface by image processing, the calculation of the road water depth H_d using the wheel as an identifier can be realised as long as the actual size H of the vehicle tire is determined.

Considering the large number of vehicles that exist in the real environment, each of which may use tires of different specifications, a single tire size cannot accurately reflect the actual situation. In order to ensure the accuracy of waterlogging depth calculation, this article classifies different vehicle types and selects a representative tire specification for each type as the actual wheel size. Therefore, three common vehicle types are tested, i.e., cars, SUVs and buses, and the corresponding tire specifications are shown in Table 1. In this table, the three types, i.e., 205/55 R16, 225/60 R18, 245/70 R17 and 275/70 R22.5, and their corresponding wheel sizes are listed. These representative tire specifications can be substituted into equation (6) for calculation.

Figure 13 Mathematical model for calculating the depth of standing water (see online version for colours)

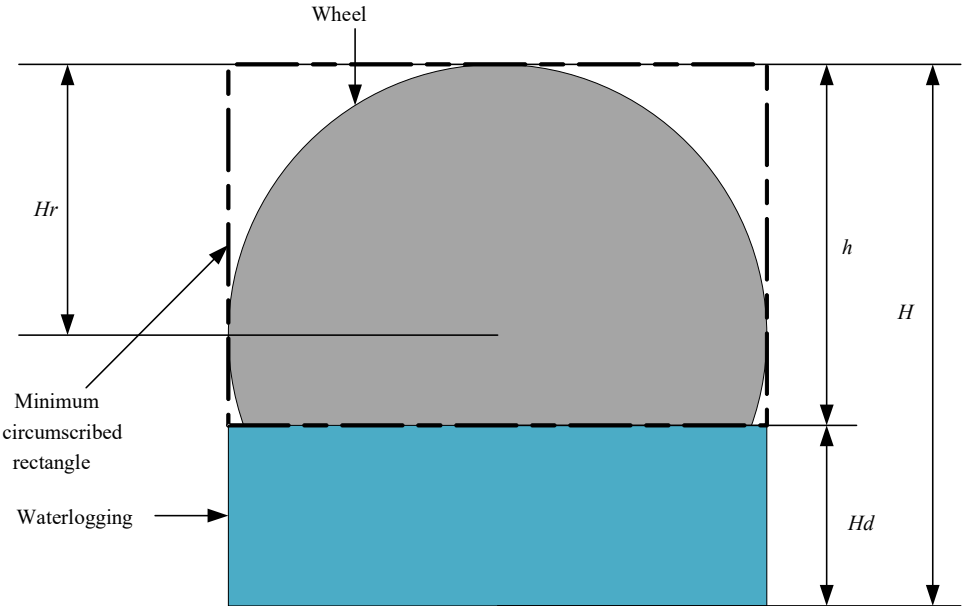


Table 1 Representative wheel sizes for different vehicle types

<i>Vehicle type</i>	<i>Representative wheel size</i>	<i>Representative wheel size/cm</i>
Car	205/55 R16	63.19
SUV	225/60 R18	72.72
Trunk	245/70 R17	77.48
Bus	275/70 R22.5	95.65

5 Model evaluation

5.1 Experimental platform

The experimental setup is configured on a Windows 10 operating system, equipped with an NVIDIA GeForce RTX 4090 GPU and a 13th Generation Intel Core i9-13900KF processor running at 3.00 GHz. The development is carried out using PyTorch as the framework, Python 3.9 as the coding language, and CUDA 11.3 for GPU acceleration. Training parameters for YOLOv5 are detailed in Table 2.

5.2 Evaluation indicators

To objectively and accurately assess the performance of a trained model, key metrics include the mean average precision (mAP), as well as the precision and recall rates. The precision rate is calculated using the formula:

$$Precision = \frac{TP}{TP + FP} \quad (7)$$

where TP represents the number of samples correctly identified as positive, FN denotes the number of samples that were incorrectly identified as negative when they are actually positive.

Table 2 Parameter settings for the training process

Training parameter	Parameter values
momentum	0.937
weigh_decay	0.0005
lr0	0.01
lrf	0.01
batch-size	16
epochs	100

The recall rate is calculated as

$$Recall = \frac{TP}{TP + FN} \quad (8)$$

where FN is the count of false negatives, which are the instances that should have been classified as positive but were incorrectly classified as negative.

The average precision mean mPA is calculated as

$$mPA = \frac{1}{n} \sum_{i=1}^n P_{A,i} \quad (9)$$

where $PA = \int_0^1 P(R)dR$ denotes the average accuracy, and n denotes the total number of categories.

5.3 Ablation experiment

To substantiate the superiority of the refined approach, this study meticulously partitions the dataset into validation, training, and testing subsets with a ratio of 2:7:1. Subsequent to this, ablation comparison experiments are meticulously conducted on the test set, with outcomes systematically detailed in Table 3. The analysis reveals that the enhanced YOLOv5-seg notably boosts its localisation detection precision over the baseline model. While the average precision for localisation detection remains stable, there is a noticeable improvement in the mean accuracy of mask segmentation. This enhancement is achieved without a compromise in computational efficiency; in fact, there is a significant reduction in computational and parameter overhead. The GFLOPs are downsized by 20.2%, from 25.7 to 20.5, and the parameter count is trimmed by 27%, from 7.4×10^6 to 5.4×10^6 .

To assess the model's inference velocity, experiments are conducted on a Windows 10 OS, powered by an NVIDIA GeForce RTX 4090 GPU and a 13th generation Intel Core i9-13900KF processor running at 3.00 GHz. The development ecosystem includes

PyTorch as the framework, Python 3.9 as the programming language, and CUDA 11.3 for GPU acceleration. With a test batch of 300 images, the original YOLOv5-seg requires 12.8 seconds, translating to an average processing time of 43 ms per sample, and an inference rate of approximately 23 samples per second. In contrast, the improved YOLOv5-seg accomplishes this in 7.8 seconds, with an average processing time of 26 ms per sample, and an inference rate of about 38 samples per second, marking a commendable enhancement of 39.5%.

The detection precision-recall curve is delineated in Figure 14, while the segmentation precision-recall curve is presented in Figure 15. A selection of the detection results is showcased in Figure 16. These visual representations of the data underscore the substantial gains achieved by the improved YOLOv5-seg, reaffirming its potential as a formidable tool in high-precision object detection and segmentation.

Table 3 Parameters for experimental evaluation

	<i>YOLOv5-seg</i>	<i>The improved YOLOv5-seg</i>
BOX_P	94.1%	94.5%
BOX_R	92.2%	91.4%
BOX_mAP50	96.4%	96.4%
Mask_P	94.8%	94.2%
Mask_R	87.8%	89.4%
Mask_mAP50	95.2%	95.6%
GFLOPs	25.7	20.5%
parameters	7.4×10^6	5.4×10^6

Figure 14 Detection precision-recall curve (see online version for colours)

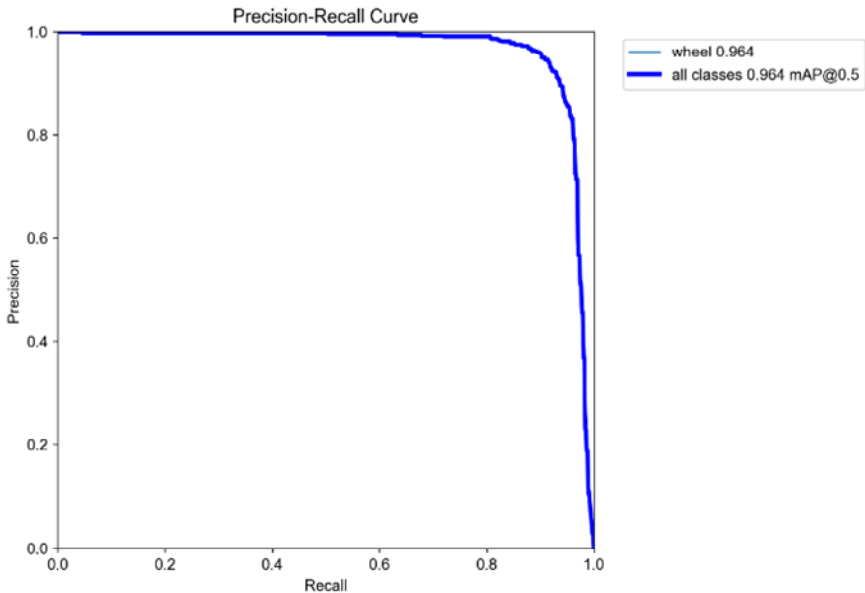
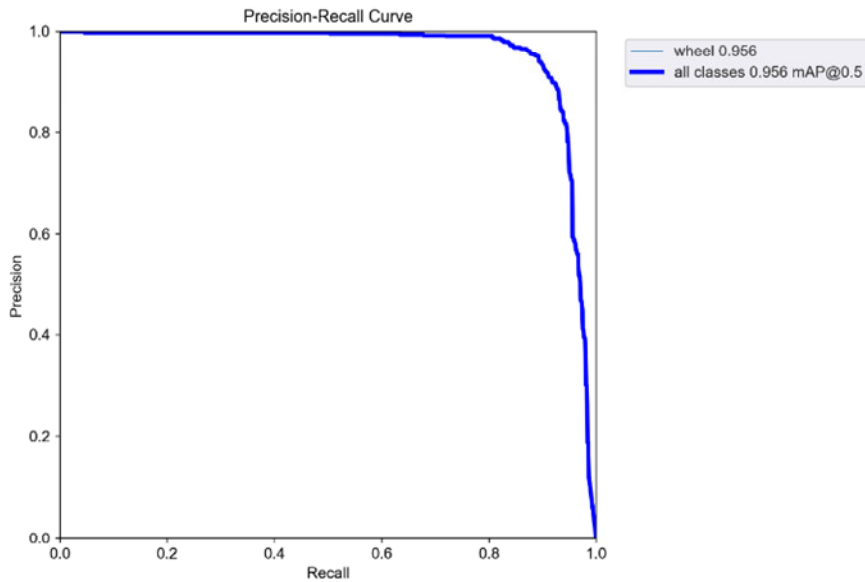
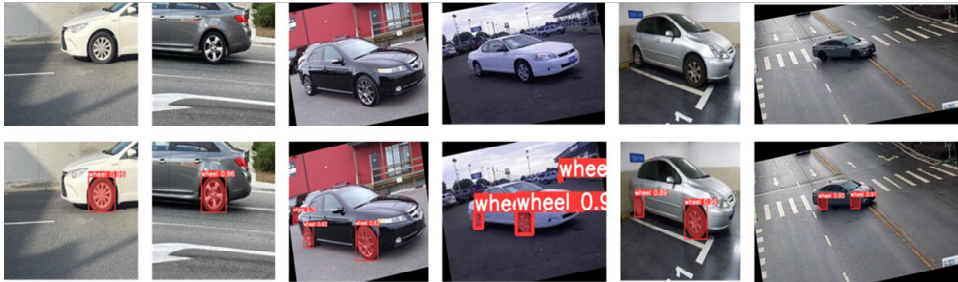


Figure 15 Split precision-recall curve (see online version for colours)**Figure 16** Detection effect display diagram (see online version for colours)

5.4 Validation of waterlogging detection model

In this article, the reliability of the waterlogging detection model is verified by simulating the waterlogging depth on different types of vehicle tires. The actual depth of the wheel submerged in water is taken as the waterlogging depth, and the depth of waterlogging is measured as the experimental standard. In this article, simulation experiments of 5 cm, 8 cm, 15 cm, 24 cm, and 32 cm are conducted for the wheels respectively. To enhance the model's generalisation, a camera is applied to capture images of each case from various angles, and 25, 30, 27, 27, and 24 images are collected for each case, resulting in a total of 133 test data. As depicted in Figure 17, the experimental data are presented in Table 4. The average error of the model when the depth is 5 cm is 2.70 cm, and the mean square deviation is 2.43 cm. The average error of the model when the depth is 8 cm is 2.38 cm, and the mean square deviation is 2.66 cm. The average error of the model when the depth is 15 cm is 2.35 cm, and the mean square deviation is 2.84 cm. The average error of the model when the depth is 24 cm is 3.22 cm, and the mean square deviation is

3.25 cm. The average error of the model when the depth is 32 cm is 2.38 cm, and the mean square deviation is 2.85 cm. The above results show that the improved YOLOv5-seg network model and the high-quality ellipse detection ellipse fitting algorithm basically satisfy the error range of the recognised ponding depth less than 5 cm, and thus the ponding depth can be detected more accurately.

Figure 17 Sample presentation of the test dataset (see online version for colours)



Comparison of the above different depths reveals that the model has poorer detection effect at a depth of 24 cm, which is analysed and found to be due to the fact that the model's recognition efficiency at the near point is better than that at the far point, and the recognition performance of the wheel is better than the wheel detection performance, and the wheel image undergoes deformation when the angle of deviation is too large, which results in the wheel output from YOLOv5-seg network to lose its shape. The ellipse mask image loses its elliptical geometric properties, which leads to the High-quality Ellipse Detection ellipse fitting algorithm cannot accurately recognise the elliptical geometric

parameters, and nearly cannot be fitted to the complete ellipse shape. Therefore, the detection effect of the experiment in the depth of 24 cm is relatively poor, and a large error is formed between the detected depth of standing water and the real value.

Table 4 Experimental data results

<i>Actual depth/cm</i>	<i>Total number of samples</i>	<i>Mean error/cm</i>	<i>Root mean square error/cm</i>	<i>Sample size</i>		<i>Proportion/percentage</i>	
				<i>Error less than 3 cm</i>	<i>Error less than 5 cm</i>	<i>Error less than 3 cm</i>	<i>Error less than 5 cm</i>
5	25	2.70	2.43	16	24	64.0	96.0
8	30	2.38	2.66	26	27	86.6	90.0
15	27	2.35	2.84	21	25	77.8	92.6
24	27	3.22	3.86	14	20	51.9	83.3
32	24	2.38	2.85	15	22	62.5	91.7

6 Conclusions

To address the issue of waterlogging depth detection in urban flood events, this article presents a novel algorithm for calculating the accurate waterlogging depth using images from traffic surveillance video systems. Compared to the original YOLOv5-seg model, the improved YOLOv5-seg demonstrates superior performance in terms of average localisation detection accuracy (96.4%) and average mask segmentation accuracy (95.6%). Moreover, the amount of model computation and parameter counts are significantly reduced, with the GFLOPs decreasing from 25.7 to 20.5, representing a 20.2% reduction, and the parameters decreasing from 7.4×10^6 to 5.4×10^6 , representing a 27% reduction, meeting the requirement of real-time detection. The simulation results indicate that the utilisation of a high-quality ellipse detection ellipse fitting algorithm can achieve more accurate fitting of ellipse geometric parameters, and also exhibits a better detection effect in practical applications.

However, there are still some deficiencies in this article that can be studied in depth, for example, The accuracy of model detection is associated with the camera's shooting angle, when the camera is facing the wheel detection effect is better than that of not facing the wheel detection effect, and when the camera deviates from the angle is too large, the mask image output from YOLOv5-seg does not meet the geometric characteristics of the ellipse, and then the ellipse fitting algorithm cannot be utilised to fit the ellipse. In addition, when the resolution of the image is not high, the boundary features of the wheels will be obviously decreased, the accuracy of detecting wheels using YOLOv5-seg will be decreased, and the output mask image is incomplete, and the subsequent consideration will be given to using the hub of the car with obvious boundary features as the reference to train the YOLOv5-seg deep learning model.

The next step will be to investigate how to accurately detect the depth of waterlogging even with a large camera deviation angle while ensuring the geometrical characteristics of the wheel, so as to further improve the accuracy of the model and reduce the error range. Meanwhile, there are fewer kinds of datasets for waterlogging detection, and in the future, we will take advantage of big data to further improve the

datasets and provide reference for the research. Furthermore, considering the harsh nature of the application environment, it is necessary to further evaluate the algorithm's ability to handle image noise, complex occlusion, and different lighting conditions.

Acknowledgements

This work is supported by the Jiangsu Provincial College Students Innovation and Entrepreneurship Training Plan Project (Grant No. 202311276103Y), Nanjing Institute of Technology fund for Research Startup Projects of Introduced talents (Grant No. YKJ202019), National Natural Science Foundation of China (Grant No. 41972111), National Natural Science Foundation of China (NSFC) (Grant No. 62101244), and the Second Tibetan Plateau Scientific Expedition and Research Program (STEP) (Grant No. 2019QZKK020604).

References

- Al-Haija, Q.A., Smadi, M. and Al-Bataineh, O.M. (2021) 'Identifying phasic dopamine releases using DarkNet-19 convolutional neural network', *2021 IEEE International IoT, Electronics and Mechatronics Conference (IEMTRONICS)*, IEEE, pp.1–5.
- Barath, D. and Matas, J. (2021) 'Graph-cut RANSAC: local optimization on spatially coherent structures', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 44, No. 9, pp.4961–4974.
- Cao, J. and Li, J. (2017) 'Improved ellipse fitting algorithm based on Letts criterion', *Journal of Computer Applications*, Vol. 37, No. 1, pp.273–277, Shanghai University School of Mechatronic Engineering and Automation.
- Chaudhary, P., D'Aronco, S., Moy de Vitry, M. et al. (2019) 'Flood-water level estimation from social media images', *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 4, No. 2/W5, pp.5–12.
- Dollar, P., Wojek, C., Schiele, B. et al. (2011) 'Pedestrian detection: an evaluation of the state of the art', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 34, No. 4, pp.743–761.
- Dong, X., Yan, S. and Duan, C. (2022) 'A lightweight vehicles detection network model based on YOLOv5', *Engineering Applications of Artificial Intelligence*, Vol. 113, No. 8, p.104914.
- Farhadi, A. and Redmon, J. (2018) 'YOLOv3: an incremental improvement', *Computer Vision and Pattern Recognition*, Springer, Berlin/Heidelberg, Germany, Vol. 1804, pp.1–6.
- Han, K., Wang, Y., Tian, Q. et al. (2020) 'Ghostnet: more features from cheap operations', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.1580–1589.
- He, K., Gkioxari, G., Dollár, P. et al. (2017) 'Mask R-CNN', *Proceedings of the IEEE International Conference on Computer Vision*, pp.2961–2969.
- He, K., Zhang, X., Ren, S. et al. (2016) 'Deep residual learning for image recognition', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.770–778.
- Huang, J., Kang, J., Wang, H. et al. (2020) 'A novel approach to measuring urban waterlogging depth from images based on mask region-based convolutional neural network', *Sustainability*, Vol. 12, No. 5, p.2149.
- Ioffe, S. and Szegedy, C. (2015) 'Batch normalization: accelerating deep network training by reducing internal covariate shift', *International Conference on Machine Learning*, PMLR, pp.448–456.

- Jiang, J., Liu, J., Cheng, C. et al. (2019) 'Automatic estimation of urban waterlogging depths from video images based on ubiquitous reference objects', *Remote Sensing*, Vol. 11, No. 5, p.587.
- Li, C., Li, L., Jiang, H. et al. (2022) *YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications*, arXiv preprint arXiv:2209.02976.
- Lin, T., Liu, X., Song, J. et al. (2018) 'Urban waterlogging risk assessment based on internet open data: a case study in China', *Habitat International*, Vol. 71, No. 1, pp.88–96.
- Lin, T.Y., Dollár, P., Girshick, R. et al. (2017) 'Feature pyramid networks for object detection', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.2117–2125.
- Liu, Y., Du, M., Jing, C. et al. (2015) 'Design and implementation of monitoring and early warning system for urban roads waterlogging', *Computer and Computing Technologies in Agriculture VIII: 8th IFIP WG 5.14 International Conference, CCTA 2014*, Springer International Publishing, Beijing, China, 16–19 September, Revised Selected Papers No. 8, pp.610–615.
- Lu, C., Xia, S., Shao, M. et al. (2019) 'Arc-support line segments revisited: an efficient high-quality ellipse detection', *IEEE Transactions on Image Processing*, Vol. 29, No. 8, pp.768–781.
- Mei, Z., Lü, H., Wen, X. et al. (2021) 'Modified ellipse fitting algorithm and phase demodulation of vibration sensing', *Acta Optica Sinica*, Vol. 41, No. 24, pp.110–115.
- Meng, Z., Peng, B. and Huang, Q. (2019) 'Flood depth estimation from web images', *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Advances on Resilient and Intelligent Cities*, pp.37–40.
- Redmon, J. and Farhadi, A. (2017) 'YOLO9000: better, faster, stronger', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.7263–7271.
- Redmon, J., Divvala, S., Girshick, R. et al. (2016) 'You only look once: unified, real-time object detection', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.779–788.
- Ren, S., He, K., Girshick, R. et al. (2016) 'Faster R-CNN: towards real-time object detection with region proposal networks', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 39, No. 6, pp.1137–1149.
- Rezatofighi, H., Tsoi, N., Gwak, J.Y. et al. (2019) 'Generalized intersection over union: a metric and a loss for bounding box regression', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.658–666.
- Sandler, M., Howard, A., Zhu, M. et al. (2018) 'Mobilenetv2: inverted residuals and linear bottlenecks', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.4510–4520.
- Scitovski, R., Sabo, K., Nikić, P. et al. (2023) 'A new efficient method for solving the multiple ellipse detection problem', *Expert Systems with Applications*, Vol. 222, No. 7, p.119853.
- Shi, X., Hu, J., Lei, X. et al. (2021) 'Detection of flying birds in airport monitoring based on improved YOLOv5', *2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP)*, IEEE, pp.1446–1451.
- Sohan, M., Sai Ram, T., Reddy, R. et al. (2024) 'A review on YOLOv8 and its advancements', *International Conference on Data Intelligence and Cognitive Informatics*, Springer, Singapore, pp.529–545.
- Wang, C.Y., Bochkovskiy, A. and Liao, H.Y.M. (2023) 'YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.7464–7475.
- Wang, C.Y., Liao, H.Y.M., Wu, Y.H. et al. (2020) 'CSPNet: a new backbone that can enhance learning capability of CNN', *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp.390–391.
- Yang, J., Fu, X., Hu, Y. et al. (2017) 'PanNet: a deep network architecture for pan-sharpening', *Proceedings of the IEEE International Conference on Computer Vision*, pp.5449–5457.

- Zhang, H., Li, W., Qi, Y. et al. (2023) 'Dynamic fry counting based on multi-object tracking and one-stage detection', *Computers and Electronics in Agriculture*, Vol. 209, No. 6, p.107871.
- Zhang, H., Shao, F., Chu, W. et al. (2024) 'Faster R-CNN based on frame difference and spatiotemporal context for vehicle detection', *Signal, Image and Video Processing*, Vol. 18, No. 10, pp.7013–7027.
- Zhang, X., Hu, M., Chen, G. et al. (2012) 'Urban rainwater utilization and its role in mitigating urban waterlogging problems – a case study in Nan**g, China', *Water Resources Management*, Vol. 26, No. 7, pp.3757–3766.