A Defect Detection Method of Drainage Pipe Based on Improved YOLOv5s

Yusheng Sun, Weibo Zhong(✉), Yuhua Li, Xiao Cui, Zhe Zhao, and Weihai Chen

Zhengzhou University of Light Industry, 136 Science Avenue, Zhengzhou 450000, China
2325882181@qq.com

Abstract. In response to the existing challenges associated with manual interpretation, low efficiency, high leakage, and misdetection rates in detecting defects in urban underground drainage pipes, this study presents a defect detection method of drainage pipe based on improved YOLOv5s. The proposed method improves the detection of large target defects and reduces the leakage detection rate by increasing a deep target detection layer. Additionally, the introduction of deformable convolutional networks (DCN) allows for more accurate feature extraction from targets with complex shapes. Furthermore, the loss function is improved by employing MPDIoU as the bounding box loss function, which not only accelerates the convergence speed of bounding boxes but also enhances target recognition accuracy. Experimental results demonstrate that the improved model surpasses the performance of the original YOLOv5s, exhibiting an improvement of 3.8% in accuracy, 1.9% in recall, and 2.1% in average precision. Additionally, the proposed method achieves an impressive inspection speed of up to 54.64 FPS (frames per second), enabling real-time and efficient drain defect detection. This method is highly practical as it provides technical support for the future deployment of CCTV pipeline robots.

Keywords: Drainage Pipe · Defect Detection · YOLOv5s · DCN · MPDIoU

1 Introduction

Drainage pipes within urban environments constitute an integral facet of the city's infrastructure. Their primary function lies in efficiently managing the treatment of sewage and rainwater, thereby ensuring unimpeded urban thoroughfares, facilitating convenient living conditions, and fostering sustainable urban development and social stability. Nonetheless, the presence of pipeline defects, such as cracks, misalignments, disconnections, and obstructions, has become apparent due to factors such as pipeline aging, urban waterlogging, and the execution of road and bridge construction projects. These defects bear the potential to instigate environmental contamination, waterlogging, and traffic predicaments, as well as pose a substantial threat to the structural integrity of buildings. Furthermore, they may engender nuisances such as odors and pest infestations, thereby significantly impinging upon the progress of cities and the quality of life for its inhabitants. Hence, it becomes imperative to promptly and effectively undertake pipeline defect detection measures to safeguard the integrity of urban construction.

D.-S. Huang et al. (Eds.): ICAI 2023, CCIS 2015, pp. 144–155, 2024.
https://doi.org/10.1007/978-981-97-0827-7_13

Presently, the common methods employed for unmanned detection of drainage pipe defects include sonar detection, periscope detection, and closed-circuit television detection (CCTV) [1]. Among them, CCTV detection stands as one of the most extensively utilized approaches in engineering sites. Common CCTV pipe robots are shown in Fig. 1. However, this method excessively relies on manual interpretation during the defect recognition process, which is not only a complicated process and a large workload, but also has a high misjudgment rate [2]. Consequently, in recent years, the realization of automatic identification of pipeline defects based on machine vision and artificial intelligence has become a research hotspot in this field. Dong [3] used support vector mechanism to build a multi-class classifier model to extract the parameters such as grayscale difference, equivalent area, and circularity from pipeline weld images to build a feature database. Through training, they achieved a remarkable accuracy of 90% in identifying weld defects using the MSVM (Multicategory Support Vector Machines) classifier. Hawari A [4] employed morphological processing, Gabor filtering, and elliptical fitting algorithms for detecting cracks, deformations, and deposits, respectively. Their average accuracy rate was recorded at 75%. Huang [5] combined two algorithms to extract gaps and crack locations in gas pipeline interfaces. By merging morphological processing with the open top-hat algorithm and the MSER (Maximally Stable Extremal Regions) algorithm, they achieved a segmentation accuracy of 61.5% for gap detection and an 86.7% accuracy rate for crack segmentation.

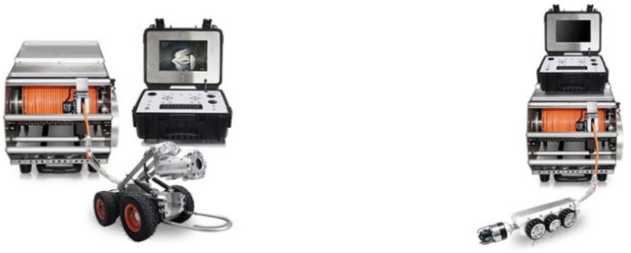


Fig. 1. Two different sizes of CCTV pipeline robots.

As deep learning continues to advance, a range of deep learning-based target detection algorithms have found application in pipeline defect detection research. These algorithms can be broadly classified into two types: one-stage and two-stage methods. Notable examples of one-stage target detection algorithms include the YOLO series [6–9] and the SSD network [10]. On the other hand, the two-stage algorithms include Faster R-CNN [11] and Mask R-CNN [12]. Wang [13] employed the Faster R-CNN network to detect and recognize six types of defects in underground drainage pipes with an average accuracy of 88.99%. Li [14] designed a new two-stage object detection algorithm for defect detection in underground drainage pipes. They utilized a multi-layer global feature fusion technique and achieved a model mAP value of 50.8%. Lu [15] employed StyleGAN2 for preprocessing the original images and made improvements to the feature fusion layer of YOLOX. They also modified the loss function to CIOU, resulting in an impressive mAP value of 68.76% for recognizing five types of defects.

The research achievements of these esteemed experts and scholars demonstrate a certain level of progress in pipeline defect detection through the integration of machine vision and artificial intelligence. However, it is worth noting that there are still some limitations, such as limited variety of defect recognition, lower accuracy rates, slower detection speed and other shortcomings.

To achieve a more efficient pipeline defect detection, the present study proposes a method that builds upon the improved YOLOv5s framework. The method incorporates several key enhancements. Firstly, a deep target detection layer is introduced to enhance the system's ability to detect large-area defects. Additionally, the introduction of DCN enables the extraction of more comprehensive feature information for irregular defects. Lastly, the optimization of the bounding box loss function to MPDIoU results in faster convergence speed and more accurate regression results. These advancements collectively contribute to a more efficient and accurate pipeline defect detection methodology.

2 Methods

The YOLOv5 algorithm framework comprises three components: the Backbone, Neck, and Head networks. The Backbone network is composed of the CBS, C3, and SPPF modules. The CBS module is responsible for extracting local features and performing downsampling operations. The C3 module utilizes residual structures to extract features while enhancing computational speed. The SPPF module achieves the fusion of local and global features. On the other hand, the Neck network incorporates the Pyramid Attention Network (PAN) to facilitate multi-scale feature fusion and enhancement. Lastly, the Head network performs prediction and filtering on the feature maps generated by the Neck network, ultimately enabling the detection of defect locations and categories.

At present, YOLOv5 has been updated to version 7.0, and it is divided into YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x based on the model's width and depth. To meet the requirements of the pipeline defects detection task and achieve faster detection results without compromising accuracy, this study selects YOLOv5s as the foundational model due to its smaller parameter count. The proposed improvements include the addition of a deep target detection layer, replacing the last C3 convolutional layer in the Backbone with deformable conv, and modifying the bounding box loss function to MPDIoU. These enhancements contribute to the development of a pipeline defect detection algorithm that combines high accuracy and efficiency. The overall network structure is illustrated in Fig. 2.

2.1 Addition of a Deep Target Detection Layer

The original YOLOv5s model consists of three detection heads in the Head layer, each responsible for detecting different sizes of objects: large, medium, and small. However, in pipeline defect images captured by CCTV pipe robots, we frequently encounter numerous large-sized defects. To improve the detection accuracy of the model for large targets, we introduce a deeper target detection layer to refine the YOLOv5s model. This improvement involves incorporating new CBS and C3 layers into the Backbone network,

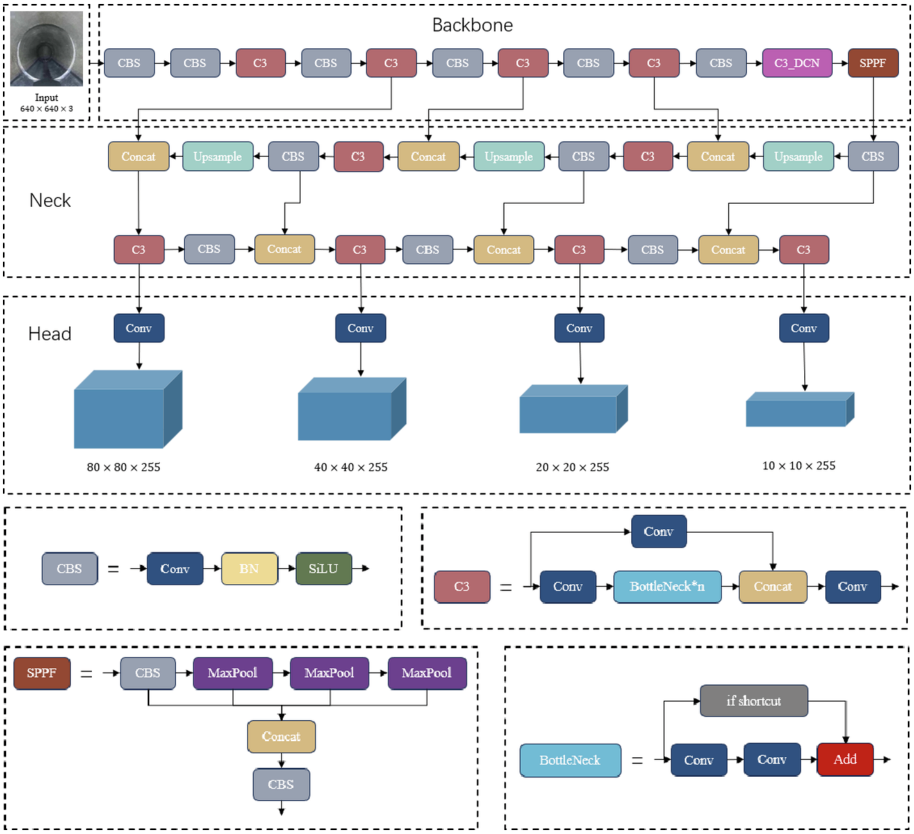


Fig. 2. Improved overall network structure.

increasing the final downsampling factor from 32x to 64x. As a result, the model becomes more adept at extracting feature information for larger-sized defects, thus improving its overall performance. Furthermore, an additional upsampling and convolution operation is introduced in the Neck network, resulting in four different sizes of feature maps: 80px × 80px, 40px × 40px, 20px × 20px, and 10px × 10px. Subsequently, these feature maps are inputted into the Head network for prediction and filtering. With the inclusion of the new detection layer, it is necessary to reconfigure the detection anchor boxes. In this study, the K-means clustering method is employed to obtain the priori anchor boxes for the dataset, and the specific configuration is outlined in Table 1.

Through the extraction of deeper features, we can achieve more precise capture of the features and intricacies of oversized targets, thereby significantly enhancing the overall accuracy of our detection system. This improvement enables the new model to better address larger defects such as disconnections and misalignments in pipeline defect detection tasks.

Table 1. Anchor box configuration corresponding to different size feature maps.

Feature map size	Receptive field size	Priori anchor box size
80px × 80px	small	(44,31) (132,56) (59,139)
40px × 40px	middle	(117,106) (71,293) (244,127)
20px × 20px	large	(132,257) (592,104) (283,245)
10px × 10px	extra large	(630,149) (625,209) (468,343)

2.2 Deformable Convolutional Network Module

The traditional convolution operation in CNN involves dividing the feature map into segments of the same size as the convolution kernel, with fixed positions for each segment on the feature map. However, due to the diverse shapes of pipe defects, this convolution method yields suboptimal results. To address this complexity in target types, this study introduces Deformable Convolution Networks (DCN) [16]. DCN incorporates a learnable offset to the sampling positions in standard convolution, enabling the convolution kernel to expand its range during the training process. This adjustment allows the kernel to better conform to the shape of the target, as depicted in Fig. 3. In Fig. 3, the green dots in (a) represent the standard convolution kernel, while the blue dots in (b), (c), and (d) represent the updated kernel positions after incorporating the offset. It is evident that the inclusion of the offset enables the kernel to adapt to various scenarios, including target movement, size scaling, rotation, and more.

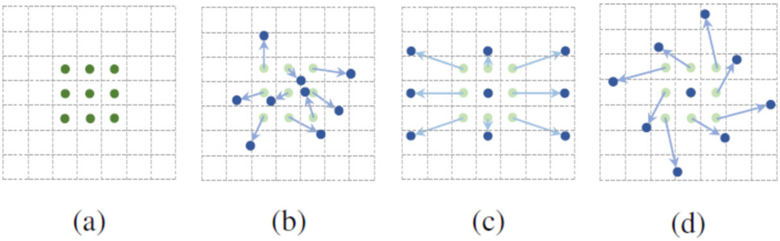


Fig. 3. Comparison of deformable convolution and standard convolution.

The traditional convolution structure can be defined by Formula 1, where p_0 represents each point in the output feature map, corresponding to the center point of the convolution kernel, and p_n represents each offset of p_0 within the range of the convolution kernel.

$$y(p_0) = \sum_{p_n \in R} w(p_n) \cdot x(p_0 + p_n) \tag{1}$$

In the case of deformable convolution, each point is introduced with an offset, which is generated by another convolution from the input feature map. This can be represented by Formula 2.

$$y(p_0) = \sum_{p_n \in R} w(p_n) \cdot x(p_0 + p_n + \Delta p_n) \tag{2}$$

As the introduced offsets in deformable convolution are typically non-integer values and do not correspond to actual pixel points on the feature map, it becomes necessary to employ bilinear interpolation to obtain the final pixel values after the offset. The diagram illustrating deformable convolution is depicted in Fig. 4.

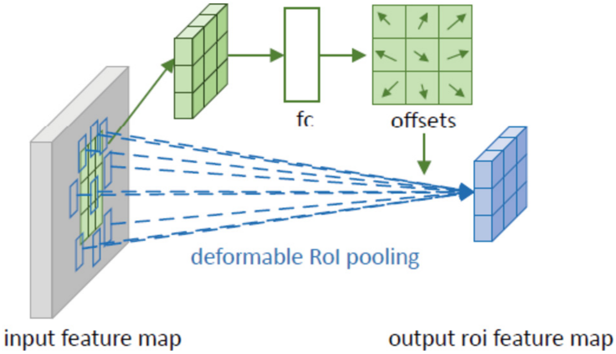


Fig. 4. Illustration of 3 × 3 deformable convolution.

The introduction of offsets and control points in deformable convolution allows for its adaptability to non-rigidly deformed targets, concurrently enhancing the receptive field, improving target localization accuracy, and reducing computational requirements. By precisely adjusting the position of the sampling point, deformable convolution can effectively capture the intricate details and boundaries of the target, thereby significantly improving the model's accuracy in detecting defective targets.

2.3 Improvement of Loss Function

The loss function of the YOLOv5s model comprises classification loss, bounding box loss and confidence loss, with the total loss being the sum of these three components. Within this framework, the localization loss function employs the CIoU (Complete Intersection over Union) metric, which considers parameters such as the distance between the predicted box and the real box, overlap rate, scale, penalty term, etc. However, this type of loss function struggles to optimize effectively when the predicted box and the ground truth bounding box have the same aspect ratio but vastly different width and height values.

To address the above issue, this study introduces a novel similarity comparison metric called Minimum Point Distance-based IoU (MPDIoU) as the measurement method for the new model's bounding box loss function. This metric considers the distance between the top-left and bottom-right points of the predicted box and the ground truth bounding box, in addition to the original IoU calculation, while simplifying the calculation process. The formula for MPDIoU is as follows:

$$MPDIoU = \frac{A \cap B}{A \cup B} - \frac{d_1^2}{w^2 + h^2} - \frac{d_2^2}{w^2 + h^2} \tag{3}$$

$$d_1^2 = \left(x_1^B - x_1^A\right)^2 + \left(y_1^B - y_1^A\right)^2 \tag{4}$$

$$d_2^2 = \left(x_2^B - x_2^A\right)^2 + \left(y_2^B - y_2^A\right)^2 \tag{5}$$

Let A and B represent the ground truth bounding box and predicted bounding box, respectively. The coordinates (x_1^A, y_1^A) and (x_2^A, y_2^A) represent the top-left and bottom-right coordinates of the ground truth bounding box, while (x_1^B, y_1^B) and (x_2^B, y_2^B) represent the top-left and bottom-right coordinates of the predicted bounding box. The MPDIoU metric incorporates all relevant factors considered in the existing loss function, including overlapping or non-overlapping regions, center point distance, and deviations in width and height.

Compared to traditional loss functions, MPDIoU offers a more efficient and concise computation process, effectively improving the accuracy and efficiency of bounding box regression in defect detection tasks.

3 Experimental Results and Analysis

3.1 Dataset Construction and Environment Configuration

The dataset used in this study is sourced from the Video Pipe ICPR2022 Video Pipeline Challenge. It consists of videos captured by CCTV pipeline robots in multiple urban underground drainage pipeline inspection projects, totaling 575 videos. From these videos, we extracted and selected 1659 images depicting various types of common pipeline defects, including misalignment (CK), crack (PL), leakage (SL), disconnection (TJ), shedding (TL), and obstruction (ZW).

To ensure the robustness and generalization capabilities of the model, we employed techniques such as rotation, flipping, and brightness adjustment to augment the dataset, resulting in a final count of 3000 defect images. The annotated samples were divided into training, validation, and test sets in an 8:1:1 ratio.

The specifications of the software and hardware devices employed in this experiment are presented in Table 2. Additionally, Table 3 provides an overview of the hyperparameters utilized during the training process.

Table 2. Software and Hardware Device Specifications.

Device	Environmental parameters
operating system	Ubuntu 20.04
CPU	Xeon(R) Platinum 8255C
GPU	NVIDIA RTX 3080 (10 GB)
memory	40 GB
programming Language	Python 3.8
deep learning framework	PyTorch 1.11.0、CUDA 11.3

Table 3. Training Hyperparameters.

Hyperparameters	Value
image size	640 × 640
epoch	100
batch size	16
initial learning rate	0.01
momentum	0.937

3.2 Evaluation Metrics

In order to assess and evaluate the performance of the improved YOLOv5s model, this study utilizes several performance evaluation metrics, including Precision (P), Recall (R), mean Average Precision (mAP) and Frames Per Second (FPS). The formulas for calculating these metrics are as follows:

$$P = \frac{T_p}{T_p + F_p} \tag{6}$$

$$R = \frac{T_p}{T_P + F_N} \tag{7}$$

$$mAP = \frac{\int_0^1 p(r)dr}{N_{classes}} \tag{8}$$

$$FPS = \frac{T_T}{N_{figure}} \tag{9}$$

Among them, T_{p} denotes the number of positive samples recognized correctly by the model, F_{p} denotes the number of positive samples recognized incorrectly by the model, F_{N} denotes the number of negative samples recognized incorrectly by the model, $p(r)$ is the PR curve, $N_{classes}$ denotes the number of classes of all defects, T_T denotes the total time of detection, and N_{figure} denotes the number of detection images.

3.3 Analysis of Experimental Results

Compared to the Original YOLOv5s. By inputting the pipeline defect dataset into the YOLOv5s algorithm model before and after the improvement and incorporating the pre-trained weights from YOLOv5s on the COCO dataset, we conducted training for 100 epochs. We obtained the mean Average Precision, Precision, and Recall curves, as depicted in Fig. 5. The figure shows that the improved mAP curve stabilizes after approximately 70 iterations, exhibiting an obvious enhancement compared to the pre-improvement stage. Ultimately, the improved YOLOv5s model achieved a mAP value of 88.0%, surpassing the original YOLOv5s model by 2.1%. Moreover, the precision and recall rates improved by 3.8% and 1.9% respectively.

Due to the addition of the deep target detection layer and DCN module in the model of this paper's algorithm, the size of the model weights is increased from 14.1 MB to 24.7 MB. Nevertheless, the detection speed of our algorithm is 54.6 FPS, which is comparable to the 54.9 FPS achieved by YOLOv5s. Both algorithms enable real-time detection capabilities.

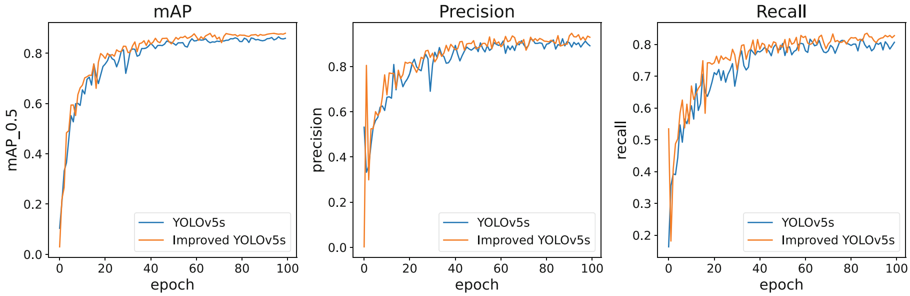


Fig. 5. Comparison of curves before and after algorithm model improvement.

Comparison of Other Algorithms. To further validate the superiority of the algorithm proposed in this study, it is compared with the common target detection algorithms Faster-RCNN, Mask-RCNN, YOLOv3, YOLOv4, and YOLOv7 on the same dataset and calculate the average accuracy of six types of defects separately. The detailed detection results are shown in Table 4. Notably, the improved algorithm showcased obvious enhancements in detecting three specific types of defects: misalignment, crack, and obstruction, with respective increases of 5.9%, 3.5%, and 3.1% compared to the top-performing alternatives. Moreover, the accuracy of detecting the remaining three defect types was comparable to the other leading algorithms. In terms of overall performance, our algorithm outperforms all others.

Table 4. Comparison results of different detection algorithms.

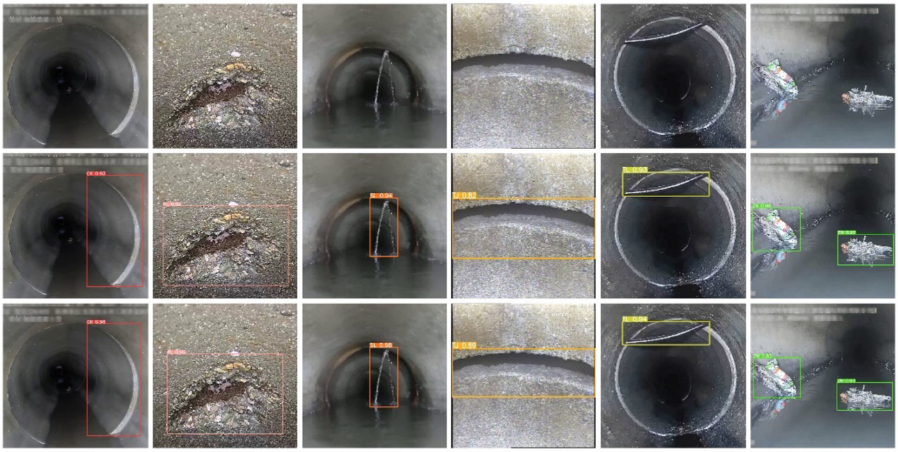
Model	AP/%						P (%)	R (%)	mAP (%)	FPS (f/s)
	CK	PL	SL	TJ	TL	ZW				
Faster-RCNN	81.5	57.8	82.2	66.7	76.9	90.3	82.3	69.7	75.9	54.2
Mask-RCNN	84.0	60.6	76.1	70.2	75.6	91.1	83.2	68.3	76.2	53.2
YOLOv3	85.8	53.8	81.5	68.8	77.3	92.1	84.5	70.2	76.6	55.8
YOLOv4	90.1	68.2	81.8	96.2	79.9	90.4	88.6	78.3	84.4	51.3
YOLOv7	86.9	71.6	82.9	96.6	80.7	89.1	90.0	77.6	84.6	51.6
Improved YOLOv5s	96.0	75.1	82.6	97.8	81.2	95.2	93.0	82.8	88.0	54.6

Ablation Experiments. In order to verify the effects of different improved methods on the model performance and experimental results, based on the original YOLOv5s model, this study conducted six sets of ablation experiments on the three improved methods, and the results are shown in Table 5. The mAP value of the original YOLOv5s model is 85.9%. By incorporating a deep object detection layer, the mAP improved by 1.0%. Introducing deformable convolution led to a further mAP improvement of 0.9%. By changing the bounding box loss function to MPDIoU, the mAP increased by 1.5%. Additionally, integrating the deep object detection layer with the deformable convolution module resulted in mAP improvement of 1.6%. Finally, when all three improvement methods were simultaneously applied to the YOLOv5s model, the mAP increased by 2.2%, demonstrating the best overall performance.

Table 5. Effect of different improvement methods on model performance.

Deep object detection layer	MPDIoU	DCN	P (%)	R (%)	mAP_0.5 (%)	FPS (f/s)
×	×	×	89.2	80.9	85.9	54.9
√	×	×	94.3	81.6	86.9	53.1
×	√	×	91.2	82.1	86.8	52.6
×	×	√	93.1	82.2	87.4	54.6
√	√	×	91.6	83.1	87.5	51.5
√	√	√	93.0	82.8	88.0	54.6

Visualization Result Analysis. To present a more intuitive illustration of the algorithm's performance before and after improvement in pipeline defect detection, this study conducted tests on the original YOLOv5s model and the improved YOLOv5s model using the test dataset. The detection results are depicted in Fig. 6. The first row showcases the original pipeline defect images, followed by the detection results using the YOLOv5s model in the second row, and finally the detection results using the improved YOLOv5s model as described in this paper, displayed in the third row. The figures are annotated with the detected defect types and corresponding confidence scores.



(a) CK (b) PL (c) SL (d) TJ (e) TL (f) ZW

Fig. 6. Comparison of detection effects.

The detection effect graphs clearly demonstrate that both the original YOLOv5s model and the enhanced model can effectively identify the correct pipeline defects. However, it is evident that the improved model yields higher confidence scores compared to the original model, and the improved model exhibits enhanced precision in localizing the detected defects.

4 Conclusion

To address the challenges in urban drainage pipeline defect detection, enhancing both speed and accuracy, this paper proposes an improved YOLOv5s algorithm model. This model enhances the precision of defect target detection and reduces the missed detection rate through means of incorporating a deep target detection layer, introducing deformable convolutions, and refining the loss function. As a result of these improvements, the enhanced model demonstrates a 3.8% increase in accuracy, a 1.9% increase in recall rate, and a 2.1% increase in mean average precision. And the detection speed is similar to that of the original model, meeting the standards for real-time detection. This algorithmic model effectively solves the problems of low detection accuracy, high missed detections, and false positives in drainage pipeline defect detection, making it highly practical. Moving forward, our future research will involve collecting a more diverse range of pipeline defect images to further expand the defect dataset. We will also focus on building a more lightweight network model, reducing parameter size and model complexity. Furthermore, we plan to deploy the model on CCTV pipeline robots to facilitate improved pipeline defect detection in collaboration with industry professionals.

Acknowledgment. This research is jointly supported by the National Natural Science Foundation of China (62072414).

References

1. Haurum, J.B., Moeslund, T.B.: Sewer-ML: a multi-label sewer defect classification dataset and benchmark. In: IEEE Computer Society. Virtual, Online, United States (2021)
2. Moradi, S., Zayed, T., Golkhoo, F.: Review on computer aided sewer pipeline defect detection and condition assessment. Infrastructures **4**(1) (2019)
3. Shaohua, D., Xuan, S., Shuyi, X., et al.: Automatic defect identification technology of digital image of pipeline weld. Nat. Gas Ind. B **6**(4) (2018)
4. Hawari, A., Alamin, M., Alkadour, F., et al.: Automated defect detection tool for closed circuit television (CCTV) inspected sewer pipelines. Autom. Constr. **89** (2018)
5. Huang, Y.L.: Research on pipeline crack defect detection method based on video images. Xi'an University of Technology (2018)
6. Redmon, J., Farhadi, A.: You only look once: unified, real-time object detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788 (2016)
7. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6517–6525 (2017)
8. Redmon, J., Farhadi, A.: YOLOv3: an incremental improvement (2018)

9. Bochkovskiy, A., Wang, C., Liao, H.M.: YOLOv4: optimal speed and accuracy of object detection. arXiv (2020)
10. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C.: SSD: single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
11. Liu, Y., Zhu, S., Qiu, W., et al.: A lightweight faster R-CNN for ship detection in SAR images. IEEE Geosci. Remote Sens. Lett. **19**, 1–5 (2022)
12. Yu, W., Ren, Y., Hu, C., et al.: Using the improved mask R-CNN and softer-NMS for target segmentation of remote sensing image. In: Proceedings of 2021 4th International Conference on Pattern Recognition and Artificial Intelligence (PRAI), pp. 1–6 (2021)
13. Wang, A.M., Lei, B.H., Chen, J.C.P.C.: Towards an automated condition assessment framework of underground sewer pipes based on closed-circuit television (CCTV) images. Tunnel. Underground Space Technol. **110** (2021)
14. Li, D., Xie, Q., Yu, Z., et al.: Sewer pipe defect detection via deep learning with local and global feature fusion. Autom. Constr. **129**(2), 103823 (2021)
15. Lu, Q.R., Ding, X., Liang, Y.W.: Underground drainage pipe defect recognition algorithm based on improved YOLOX. Electron. Meas. Technol. **45**(21), 161–168 (2022)
16. Dai, J., Qi, H., Xiong, Y., et al.: Deformable convolutional networks. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 764–773 (2017)