applied sciences

MDPI

Article

SGST-YOLOv8: An Improved Lightweight YOLOv8 for Real-Time Target Detection for Campus Surveillance

Gang Cheng *, Peizhi Chao, Jie Yang and Huan Ding

College Surveying & Land Information Engineering, Henan Polytechnic University, Jiaozuo 454000, China; 212204010025@home.hpu.edu.cn (P.C.); yangjie@hpu.edu.cn (J.Y.); dhuan898@163.com (H.D.)
* Correspondence: chenggang@hpu.edu.cn

Abstract: Real-time target detection plays an important role in campus intelligent surveillance systems. This paper introduces Soft-NMS, GSConv, Triplet Attention, and other advanced technologies to propose a lightweight pedestrian and vehicle detection model named SGST-YOLOv8. In this paper, the improved YOLOv8 model is trained on the self-made dataset, and the tracking algorithm is combined to achieve an accurate and efficient real-time pedestrian and vehicle tracking detection system. The improved model achieved an accuracy of 88.6%, which is 1.2% higher than the baseline model YOLOv8. Additionally, the mAP0.5:0.95 increased by 3.2%. The model parameters and GFLOPS reduced by 5.6% and 7.9%, respectively. In addition, this study also employed the improved YOLOv8 model combined with the bot_sort tracking algorithm on the website for actual detection. The results showed that the improved model achieves higher FPS than the baseline YOLOv8 model when detecting the same scenes, with an average increase of 3–5 frames per second. The above results verify the effectiveness of the improved model for real-time target detection in complex environments.

Keywords: YOLOv8; target tracking; Soft-NMS; GSConv; triplet attention

Citation: Cheng, G.; Chao, P.; Yang, J.; Ding, H. SGST-YOLOv8: An Improved Lightweight YOLOv8 for Real-Time Target Detection for Campus Surveillance. *Appl. Sci.* **2024**, *14*, 5341. https://doi.org/10.3390/app14125341

Academic Editor: Andrea Prati

Received: 16 May 2024
Revised: 18 June 2024
Accepted: 19 June 2024
Published: 20 June 2024

1. Introduction

In China, with the expansion of the number of students on campuses, campus security is becoming more and more important, which brings great challenges to the maintenance of students' safety and campus order. With the continuous development of deep learning, intelligent surveillance based on target recognition has gradually been introduced into campuses, providing new solutions for campus stability. However, current target detection algorithms available on the market tend to be bulky when deployed on the web, so the network structure of the deployment model needs to be optimized.

Real-time target detection is one of the key technologies in campus surveillance systems. It can automatically identify and track various targets on the campus, such as individuals and vehicles, based on their recognition IDs. However, traditional target detection algorithms such as HOG [1], SIFT [2], and DDPM [3] face performance challenges in campus surveillance scenarios, including low real-time capability and high computing resource consumption.

Yun Wei et al. proposed a two-step target detection algorithm by combining Haar and HOG features, which improved the performance of multi-vehicle target detection and tracking in complex urban environments. The results showed that compared to traditional methods, this algorithm achieves higher detection accuracy and time efficiency [4]. Hongzhi Zhou et al. conducted experiments by combining HOG and LTP features and weighted the features of the color map and depth map to provide richer visual information. The results showed that this method improved the accuracy and efficiency of pedestrian detection [5].

With the continuous development of target detection algorithms, such as the emergence of new algorithms such as YOLO, Faster R-CNN, SSD, and RetinaNet, remarkable breakthroughs and progress have been made in the field of target detection. Currently, target detection algorithms can generally be categorized into two main directions: two-stage

detection and one-stage detection. Two-stage detection algorithms consist of two main components: (1) candidate region extraction and target classification and (2) boundary box regression, such as Faster R-CNN, Cascade R-CNN, and Mask R-CNN. Faster R-CNN is a type of target detection model that introduces a region proposal network (RPN). This model can simultaneously generate target proposals and their corresponding class probabilities, thus improving detection accuracy [6]. Cascade R-CNN enhances the performance of target detection through a cascaded approach, progressively filtering out more accurate target bounding boxes using a cascaded structure [7]. Based on Faster R-CNN, Mask R-CNN adds support for object instance segmentation, which can predict the boundary box and pixel level mask of the object at the same time [8]. Two-stage detection algorithms perform well in terms of accuracy, but correspondingly sacrifice a certain amount of detection speed. Sirisha discussed the advantages and disadvantages of one-stage detection and two-stage detection through various variants of the current YOLO detector [9]. She found that in terms of detection accuracy, two-stage detection is generally superior to one-stage detection. However, they are slightly inferior in inference speed. Lingcai Zeng et al. combined the Adversarial Occlusion Network (AON) with the standard Faster R-CNN detection algorithm to detect complex underwater targets [10]. The results showed that joint training for target detection helps alleviate overfitting caused by fixed pre-generated data, providing a promising detection solution for underwater exploration. Lehai Zhong et al. integrated the bidirectional feature pyramid network (BiFPN) into the Cascade R-CNN to overcome errors and omissions in target occlusion and small target scenes [11]. The improved method achieved a 0.91 mAP accuracy on the wildlife video frame dataset, and each detection time was only 0.42 s. For the problems of poor quality and accuracy of multi-object segmentation and detection in complex traffic scenes, Shuqi Fang improved the MaskR-CNN: the path enhancement strategy is introduced, and the Efficient Channel Attention module (ECA) is added to optimize the semantic infographic; ResNet in the original backbone network is replaced by ResNet network with group convolution to enhance feature extraction ability. The results showed that the detection accuracy and segmentation accuracy of the improved Mask R-CNN algorithm increased by 4.73% and 3.96%, respectively.

One-stage detection algorithms mainly include YOLO, SSD, and RetinaNet. Compared with two-stage detection, these algorithms directly predict the category and bounding box of the target through a single neural network, omitting the process of candidate region extraction, which is faster. YOLO is a fast and accurate target detection algorithm with a single forward propagation [12]. SSD is another popular real-time target detection algorithm. It predicts bounding boxes and class probabilities for objects of different scales using multiple layers of feature maps [13]. Unlike the previous two, RetinaNet combines an efficient feature pyramid network and Focal Loss to solve the problem of category imbalance in target detection, thereby improving detection performance [14]. Therefore, one-stage detection algorithms can achieve real-time target detection, but compared to two-stage detection algorithms, they may have slightly lower accuracy. Heming Hu et al. achieved high accuracy in strawberry detection by combining two-stage detection (Mask R-CNN) and one-stage detection (YOLOv3) networks for training and recognition [15]. To solve multiple complex situations in actual traffic scenes, Yalin Miao et al. proposed a deep learning target detection network based on SI-SSD [16]. This network utilizes the feature pyramid network (FPN) and feature map fusion method to combine shallow and deep feature maps and enhance its sensitivity to small objects. Hong Liang et al. addressed the issue of insufficient information features for small objects by combining MFEM with RetinaNet [17]. They constructed a bidirectional feature pyramid network model, which prevented information loss and effectively integrated strong semantic information and high-resolution information.

Among them, the YOLO algorithm has achieved tremendous success in the field of target detection due to its unique concept and outstanding performance. More and more researchers are starting to explore the application of target detection algorithms in actual scenarios.

Yang Wu et al. proposed a lightweight and real-time method based on an improved instance segmentation model, which has achieved remarkable results in pixel-level crack detection [18]. Shichu Li et al. proposed a glove detection algorithm (YOLOv8-AFPN-M-C2F) based on YOLOv8 [19]. This algorithm replaces the head of YOLOv8 with the AFPN-M-C2f network, expanding the path of feature vector propagation and alleviating semantic differences between non-adjacent feature layers. At the same time, the surface feature information is enriched by introducing a superficial feature layer, and the sensitivity to small objects is improved.

To achieve real-time detection of students and passing vehicles in the complex campus environment, while addressing the challenges of lightweight model and high performance, the main contributions of this paper are as follows:

1. By introducing the Triplet Attention module, which reduces background interference by enhancing target representation and integrating context information. The results show that the introduction of the module can improve the performance of target detection tasks.
2. Using the GSConv + SlimNeck to replace Conv and C2f modules in the YOLOv8 model for lightweight operation. In addition, compared with the original NMS algorithm, the Soft-NMS algorithm is introduced to optimize potential batch omission issues that may arise in dense occlusion scenes.
3. The model was deployed into actual production, achieving a real-time monitoring FPS 3–5 frames higher than the baseline YOLOv8 model. The experiments demonstrate that in campus surveillance scenarios, SGST-YOLOv8 has a smaller model size and better detection capability.

2. Method

The real-time pedestrian and vehicle detection method proposed in this study is based on an improved version of the YOLOv8 model, named SGST-YOLOv8. This model is capable of real-time detection of vehicles and pedestrians. As shown in Figure 1, the study first performed data cleaning on the existing dataset, which includes thousands of images of vehicles and pedestrians. Subsequently, the dataset was divided into training, validation, and test sets. In this study, the training and verification sets were first used for the pre-training stage. Then, the pre-training weight results were transferred to the SGST-YOLOv8 model trained with data-enhanced pedestrian and vehicle datasets through transfer learning for training and verification.

2.1. Overview of YOLOv8

YOLOv8 is a novel target detection model created and maintained by the startup company Ultralytics, offering state-of-the-art performance in terms of accuracy and speed. YOLOv8 has been iterated and optimized based on the previous YOLO version, which has made it a new favorite in the target detection field [20]. The architecture of YOLOv8 is built upon earlier versions of the YOLO series, as shown in Figure 2.

YOLOv8 uses an anchor-free detection method to predict the target, which improves the detection speed and accuracy [21]. However, some issues remain in handling real-world campus surveillance scenarios, such as crowded crowds, mutual occlusion between members, and computational performance [22]. To solve these problems, this paper proposes a lightweight YOLOv8 model.

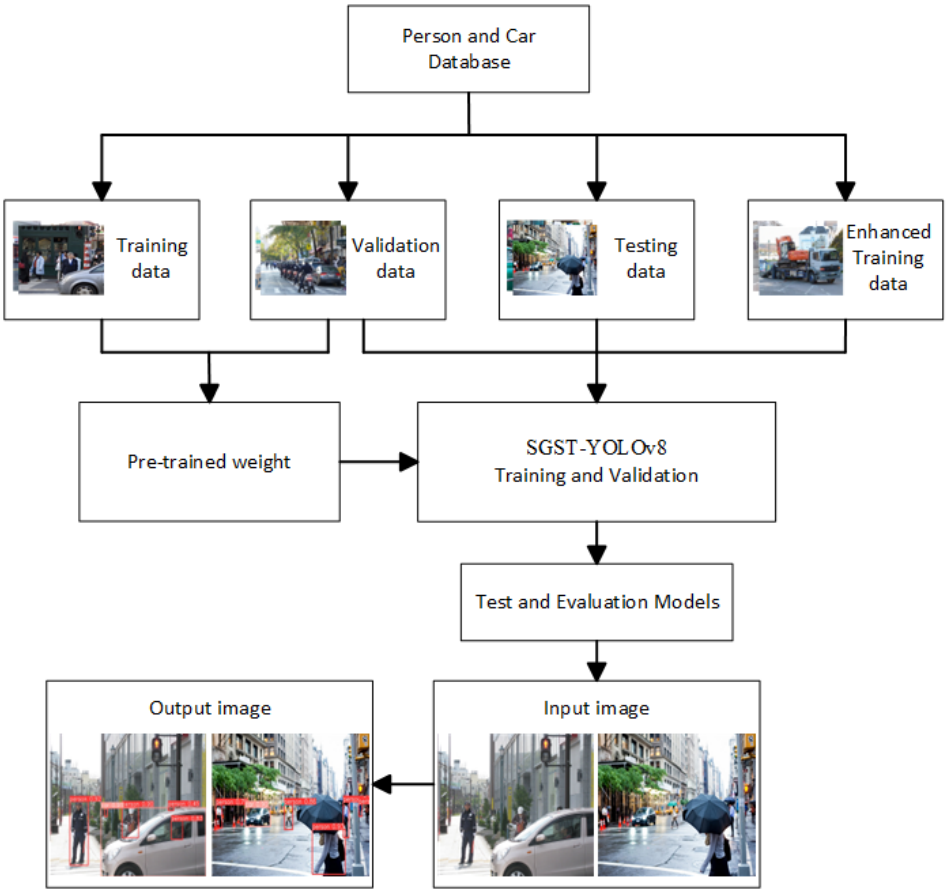


Figure 1. Research framework for the model improvement approach.

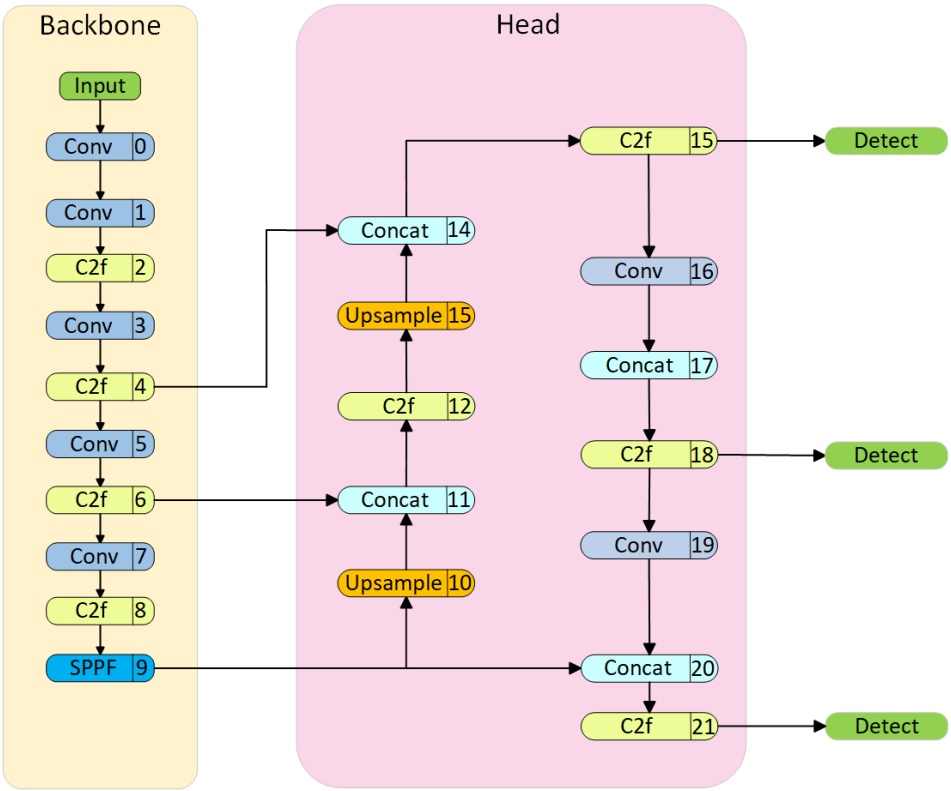


Figure 2. YOLOv8's network structure diagram.

2.2. SGST-YOLOv8: Lightweight Pedestrian and Vehicle Detection Network Architecture

SGST-YOLOv8 (Figure 3) has made some optimized modifications based on YOLOv8: Soft-NMS was introduced to replace the original NMS loss function to reduce the missed rate in complex scenarios. At the same time, the GSConv lightweight module and the generated VoV-GSCSP on this basis was introduced to replace the Conv and C2f modules of the original neck part. Additionally, the Triplet Attention module is added to the backbone network to enhance the dimensional interpretation and improve the accuracy of the model.

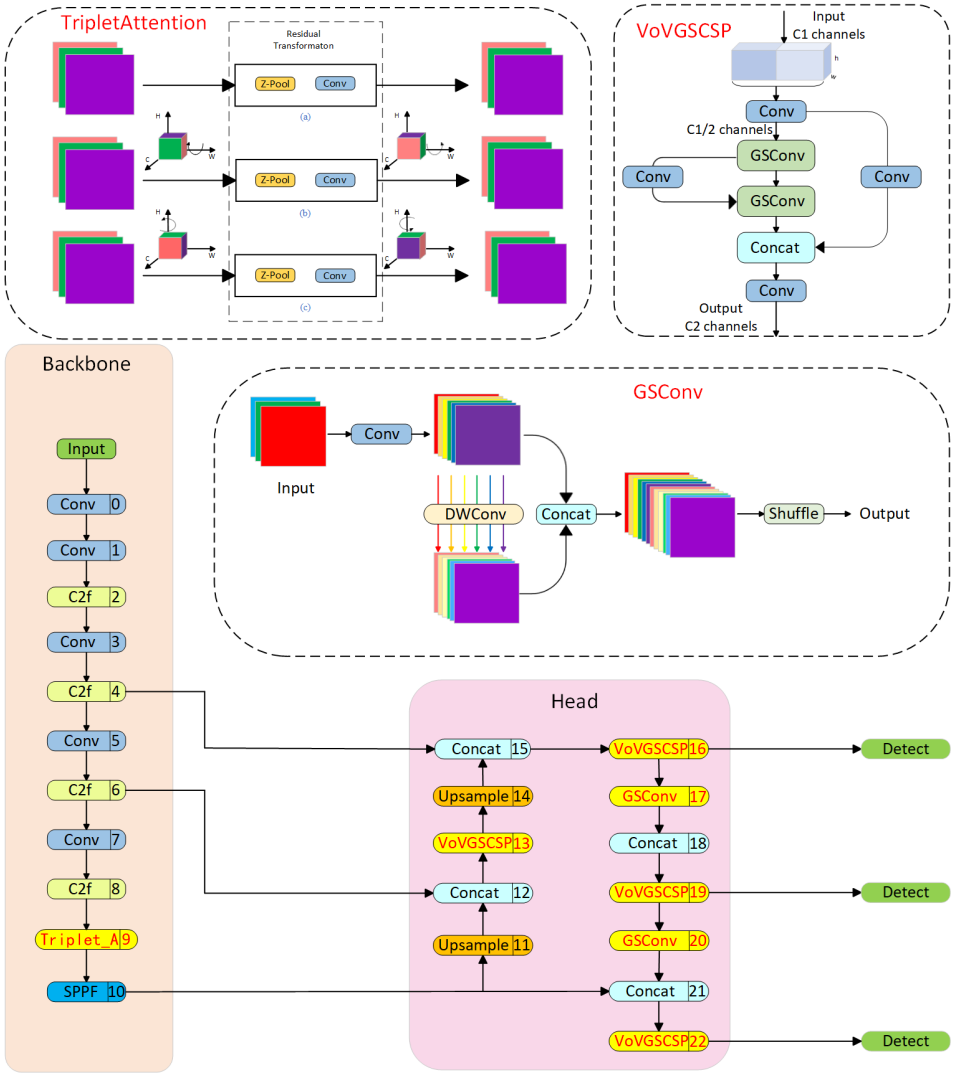


Figure 3. Network structure diagram of SGST-YOLOv8 with added modules. "(a)–(c)" They respectively represent three branches capturing cross-dimensional interaction information between different dimensions.

2.3. Lightweight Neck Network Module: GSConv

GSConv (Group Separable Convolution) is a unique convolution operation technique [23]. Its core objective is to enhance the performance of Depth-wise Separable Convolution (DSC) [24] by making its output closer to Standard Convolution (SC) [25]. The design concept of GSConv is to find an optimal balance between the accuracy of the model and the speed of calculation. By using GSConv, the computational cost of the model can be effectively reduced while maintaining or even improving the performance of the model.

In the improved model of this study, the Standard Convolution (SC) in the neck networks is replaced by GSConv. The reason for not using this method in the backbone network is that employing GSConv in the backbone may lead to excessively high computa-

tional complexity. Meanwhile, in the neck part, the feature map has become elongated, and hence, transformation is no longer needed [26]. Replace the formula as

$$YOLOv8(input) = GSConv(SC(C2/2 \times 2)) \quad (1)$$

Here, "input" refers to the input feature map, and (C2/2 × 2) represents the convolution results after two depth-wise convolutions. GSConv uses a combination of SC, DSC, and shuffle-mixed convolutions for dense convolution calculations, maximizing the retention of hidden connections between each channel. Consequently, it achieves the same output effect as SC with fewer computational costs. In this study, a cross-stage part network called VoV-GSCSP module was used, which is generated based on GSConv. The introduction of convolutions like GSConv and VoV-GSCSP increases the model's non-linear capability while maintaining parameter sharing, thereby enhancing the model's generalization ability [27]. It decreases computational complexity and simplifies network structure while preserving adequate accuracy. This helps to reduce the risk of overfitting in the model.

2.4. The Triplet Attention Module

In recent research, attention mechanism modules such as SE, CA, ECA, CBAM, GAM, etc., have appeared more frequently in computer vision tasks. While these attention mechanism modules have achieved significant success in enhancing the performance of deep learning models, they may pose risks such as large memory consumption or overfitting due to the complex network structure.

The Triplet Attention is a lightweight triple attention mechanism proposed by Diganta Misra et al. [28]. The attention mechanism module captures relevant information in three directions: horizontally, vertically, and across channels, in multiple dimensions. At the same time, weights are calculated and applied to enhance the characteristics of the target information. It has been proven by the results of previous experiments that Triplet Attention is effective and practical in target detection tasks and can capture cross-dimensional dependencies [29]. This innovative attention mechanism has brought breakthroughs and advancements to the field of target detection.

2.5. Soft-NMS Loss Function

Soft-NMS is an improved loss function used to suppress redundant bounding boxes more smoothly during the NMS process [30]. Previously, the traditional NMS method used a fixed threshold to determine if two bounding boxes overlap and to suppress them [31]. However, due to the fixed threshold, it may adapt to situations between different objects, resulting in some candidate boxes with low confidence but which incorrectly exclude significantly overlapping true objects. In contrast, Soft-NMS employs a smoother strategy to handle overlapping bounding boxes. For adjacent bounding boxes with an Intersection over Union (IoU) greater than the NMS threshold, traditional NMS methods set their scores to 0, effectively discarding them, which may lead to missing boundary boxes, especially in occlusion scenarios. However, Soft-NMS retains confidence information by reducing the scores of overlapping bounding boxes rather than immediately discarding them (Figure 4).

Specifically, in Soft-NMS, the scores of overlapping bounding boxes are adjusted based on their degree of overlap and confidence level. Bounding boxes with higher overlap and confidence levels will receive smaller score penalties, while those with lower overlap and confidence levels will receive larger score penalties.

Soft-NMS performs this by introducing a decay function. Soft-NMS can be more flexible in adjusting the weights of candidate boxes. For the candidate box that is highly overlapping with the selected box but has low confidence, there is still a chance for them to be retained, thus improving the accuracy of target detection.

Figure 4. Plot of the results of target recognition for dormitory students after the introduction of Soft-NMS.

2.6. Transfer Learning Strategy

Introducing a new backbone network and modifying the neck structure require training the new convolutional network architecture from scratch. This process is typically an iterative trial-and-error process, requiring continuous iteration and parameter-tuning of the network architecture and hyperparameters to find the optimal configuration [32]. However, surveillance devices used in campus environments often struggle to capture high-quality images. This presents a challenge for target detection tasks, as models need to accurately identify and locate objects in low-quality images. To solve these problems, transfer learning technology is introduced into the research, which uses prior knowledge and feature transfer to reduce training costs [33].

As shown in Figure 5, the model undergoes initial pre-training on a large dataset for feature learning. Then, feature transfer is performed, followed by secondary model training on the pedestrian and vehicle dataset. Using cross-domain transfer learning strategies to adjust models and transfer parameters can reduce the model's data dependency, improve its robustness, and lower the training cost.

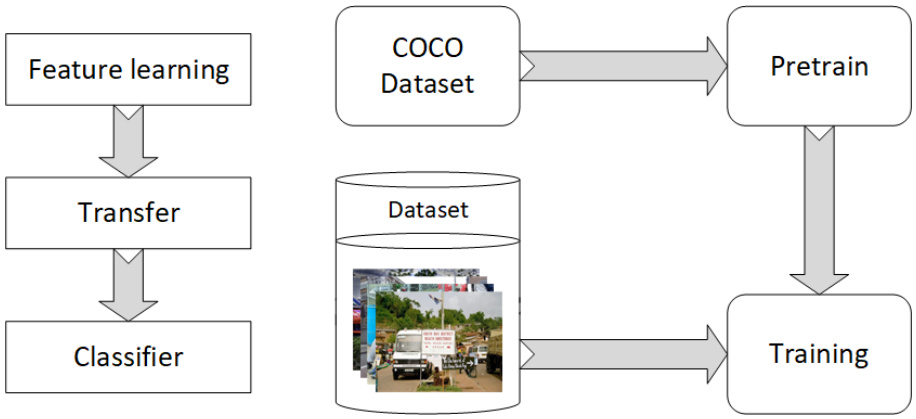


Figure 5. Basic road map for pre-training.

3. Experiments

3.1. Data Preprocessing and Enhancement

The dataset was downloaded from the Kaggle database and contains pedestrian and vehicle data. However, the initial data had some invalid areas and data format issues for this study. Therefore, all the data were screened, the invalid areas were removed, and noise removal and data balance were performed on the data. The study divided the original dataset into proportions of 7:1.5:1.5. To match the training of the YOLO model, the dataset format was converted to the COCO dataset format.

To improve the model's generalization and robustness, the study applied data enrichment to the training dataset. Several data enhancement strategies (Figure 6) were used in this study, including brightness adjustment, Gaussian noise adjustment, and contrast adjustment.

Figure 6. Data enhancement operations.

3.2. Experimental Environment and Training Parameters

The experimental environment of this research project is based on the cloud server platform 'Featurize'. The NVIDIA RTX A4000 graphics card with a total memory of 30.1 GB was used as the hardware platform for model training, with specific parameters as shown in Table 1. The programming work throughout this study was based on the Python 3.10.12 environment, using the PyTorch2.0.1 GPU version for deep learning model training. The model training and testing are performed strictly according to the above parameters.

Table 1. Experimental environments for model training and testing.

Scenario	Graphics Card Model	Random Access Memory (RAM)	Display Memory	Experimental Environment
Model training	NVIDIA RTX A4000	30.1	16.9	Python 3.10.12 torch2.0.1GPU
Model testing	NVIDIA GeForce RTX 3060	16	8	Python 3.8.1 torch2.0.1GPU

In terms of model training, the study specified an input image size of 640 × 640, a batch size of 53, and utilized 8 workers. The number of training epochs was determined through grid search, and early stopping epochs were adjusted based on parameter settings. For model inference and practical deployment testing, the experiment utilized the NVIDIA GeForce RTX 3060 graphics card and a Hikvision 4-million-pixel HD camera to verify the model's efficiency on the website.

3.3. Evaluation Index

In the experiments of this paper, Precision, Recall, mAP50, and mAP0.5–0.95 were used as accuracy metrics from the experimental results, while the model's parameters, GFLOPs, and the FPS of the deployed system are used as lightweight evaluation criteria. Precision, Recall and mAP are calculated as follows:

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

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

$$AP = \int_{r=0}^{1} P(r)dr \tag{4}$$

$$mAP = \frac{1}{K}\sum_{i=1}^{K} AP_i \tag{5}$$

TP, FP, and FN represent the number of correctly identified true samples, the number of incorrect identifications, and the number of correct samples missed, respectively.

4. Results and Validation

4.1. Experimental Results

4.1.1. Pre-Training Results

To explore and verify the possibility of model improvement more quickly, we conducted pre-training on the original pedestrian and vehicle dataset. As shown in Figure 7, the model closely approximates the fitting accuracy of the baseline model while showing significant improvement in training efficiency.

By observing the model accuracy and convergence efficiency (Table 2), it can be inferred that this YOLOv8 model may outperform the baseline YOLOv8 on medium- to large-sized objects.

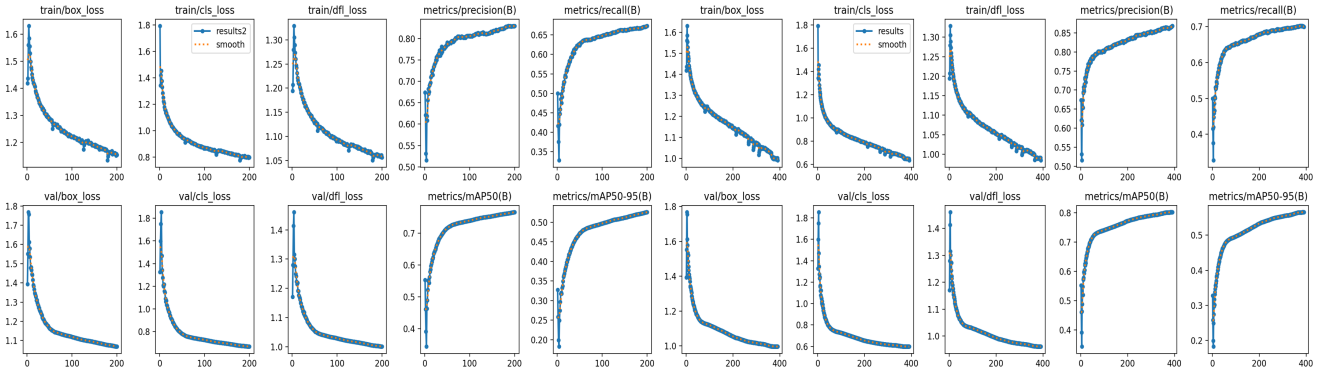


Figure 7. Plot of pre-training results of the improved model and baseline model.

Table 2. Accuracy results for the three types of mAP targets.

Epoch	Model	AP_s	AP_m	AP_l
Model training	YOLOv8	0.485	0.583	0.810
	SGST-YOLOv8	0.471	0.579	0.811

4.1.2. Further Training

The study conducted further training on the augmented dataset. The training results of the improved model on the pedestrian and vehicle dataset are shown in Figure 8: as the number of epochs increases, each loss and Precision changes. It can be observed that during the initial epochs, both training loss and validation loss first increase and then decrease rapidly because of the transfer of training parameters from the pre-trained model to continued training. Subsequently, both losses decrease as the training epochs increase. Meanwhile, Precision, Recall, mAP0.5, and mAP0.5:0.95 exhibit an increasing trend.

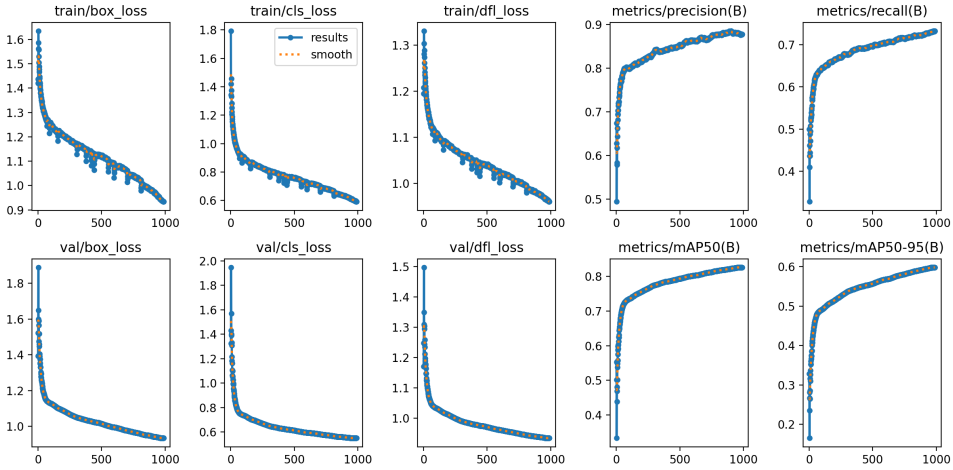


Figure 8. Plot of training results for the improved model.

The horizontal coordinate in Figure 8 is the number of training cycles of the model, and the vertical coordinate is the error size and parameter accuracy. The smaller the loss and the higher the Precision, the better the accuracy of the model.

4.2. Ablation Experiment

Follow-up studies performed ablation experiments and obtained the following data results (Table 3). The experimental dataset used the same pedestrian and vehicle dataset to ensure that the experiments were credible. The inclusion of the Soft-NMS technique led to a significant improvement in accuracy compared to the baseline YOLOv8 model. It is worth noting that when only Soft-NMS technology was introduced into the baseline model,

the Recall increased by 0.8%, while mAP0.5 and mAP0.5: 0.95 increased by 0.5 percent and 9 percent, respectively. Additionally, the introduction of the GSConv lightweight module alone also resulted in a certain degree of improvement in mAP. The combination of GSConv, Triplet Attention, and Soft-NMS reduced the model parameters by 5.6% and GFLOPs by 7.9%. Meanwhile, SGST-YOLOv8 improved by 1.2% in Precision, 3.1% in Recall, 2.0% in mAP0.5, and 3.2% in mAP0.5:0.95.

Table 3. Accuracy results for each scenario of the ablation experiment.

Scenario	Model	Precision	Recall	mAP0.5	mAP0.5:0.95	Parameter	GFLOPs
1	YOlOv8	0.874	0.711	0.811	0.577	3,157,200	8.9
2	+Soft-NMS	0.862	0.718	0.816	0.586	3,157,200	8.9
3	+GSConv	0.874	0.710	0.812	0.580	2,947,792	8.1
4	+Triplet_A [1]	0.877	0.705	0.810	0.578	3,190,168	9.0
5	1 + 3 + 4 [2]	0.875	0.735	0.826	0.598	2,980,760	8.2
6	1 + 2 + 3 + 4	0.886	0.742	0.831	0.609	2,980,760	8.2

[1] +Triplet_A is short for Triplet Attention. [2] Here is the baseline model YOLOv8 introducing modules such as GSConv and Triplet Attention.

Compared to Scenario 3, which contains the GSConv module, Scenario 6 of the improved model has a slightly higher number of parameters. However, overall, the model combination in Scenario 6 enhances the comprehensive capability of the model. Although there is a slight decrease in performance after introducing the GSConv convolution, the combination of GSConv and Triplet Attention has improved the accuracy compared to the baseline YOLOv8 model. This is because these two techniques complement each other in terms of performance. The Triplet Attention module reduces information loss by enhancing feature representation, thereby improving accuracy. It compensates for the minor impact on performance resulting from the introduction of the GSConv convolution and VoV-GSCSP lightweight module. At the same time, convolutional operations like GSConv and VoV-GSCSP can reduce the complexity of calculations, thereby improving the model's computational efficiency. This makes the model more efficient during both training and inference stages.

4.3. Evaluation Experiment

To test the model's efficiency and experiment, the experiment uses sample test images to make predictions. Figure 9 shows the improved and baseline models' detection results. The figure shows that the improved model exhibits better detection performance for pedestrians and vehicles in crowded environments. Even in cases where vehicles and pedestrians are partially occluded, the improved model still performs remarkably well. Furthermore, it can be observed that the improved model exhibits higher sensitivity to variations in image scale. This result is particularly crucial for subsequent deployment on the web for real-time monitoring.

To further validate the theoretical effectiveness of the model in practice, this study employs Grad-CAM for heatmap visualization. Grad-CAM is a computer vision technique for interpretability. It generates a gradient-weighted-class activation map to represent the key contributing regions of the neural network's prediction on the image.

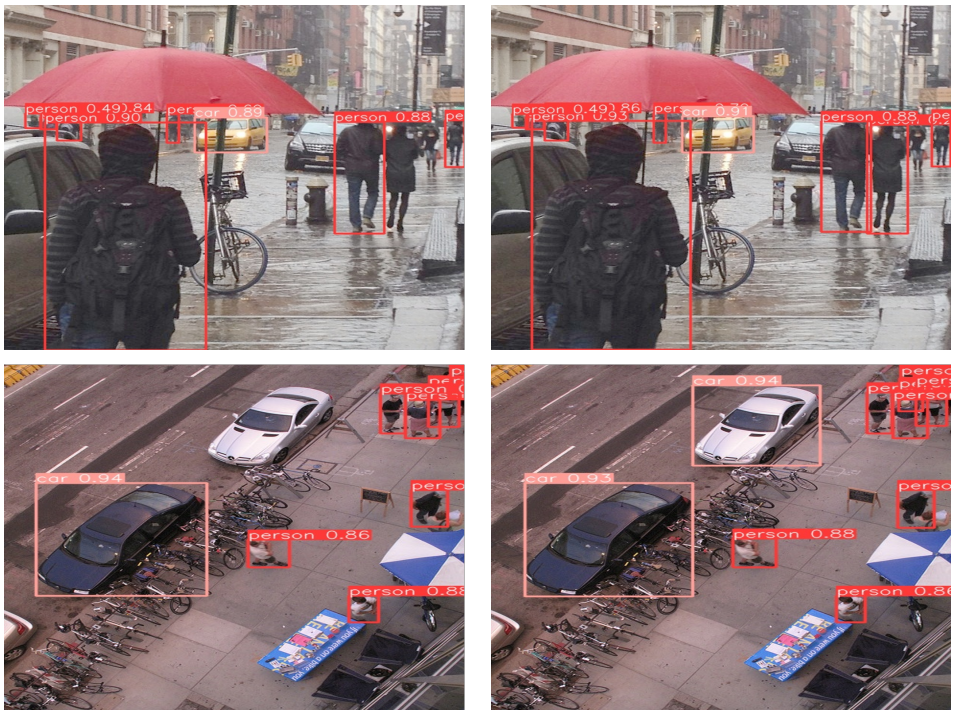
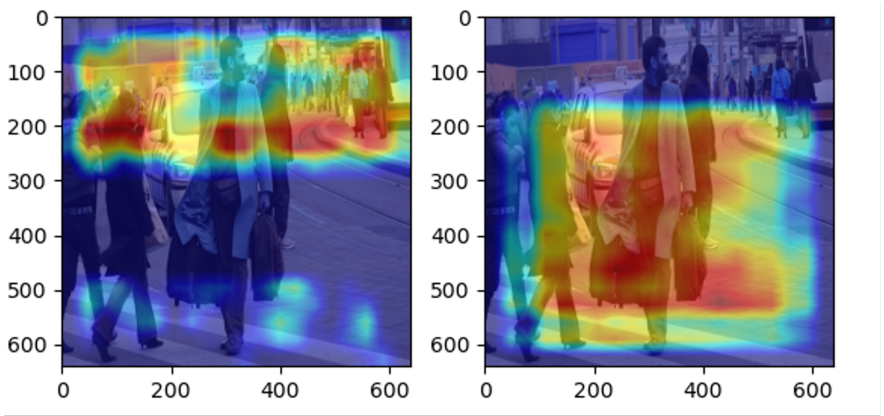


Figure 9. Validation result chart.

Utilizing Grad-CAM for analysis, it is evident that in complex regions where targets are detected, the improved model shows greater sensitivity within corresponding heatmap areas than the baseline YOLOv8 model (Figure 10).

The improved model can analyze images of varying scales and extract useful information from images to the greatest extent. In contrast, the baseline model focuses on large areas containing vehicles and pedestrians in complex scenes and fails to capture information about pedestrians in distant regions. The results showed that the improved model is more effective in capturing the characteristics of the detection target in a complex environment.



(a) SGST-YOLOv8 (b) Baseline model YOLOv8

Figure 10. SGST-YOLOv8 and baseline model YOLOv8 visualization results.

4.4. Web Deployment Testing

As long as the target vehicles and pedestrians can be successfully detected, the experiment can deploy the tracking algorithm for further tracking. The study used two commonly used tracking algorithms for experimentation and testing, that is, BoT-SORT and ByteTrack. Figure 11 shows the algorithm model's web deployment and testing workflow. The BoT-SORT algorithm is an improved version of the SORT algorithm, which enhances tracking

capabilities in complex scenarios by integrating appearance and motion information. Especially when targets are occluded or crossed, its deep learning appearance model, combined with a refined cost function and rapid motion model, can optimize data and enhance the accuracy and robustness of target identification. It is applicable in areas such as video surveillance and autonomous driving. The ByteTrack algorithm improves the detection and tracking of small targets and performs excellently in handling target occlusion and interactions. A low-confidence detection box processing mechanism effectively enhances the tracking capability of small and temporarily occluded targets.

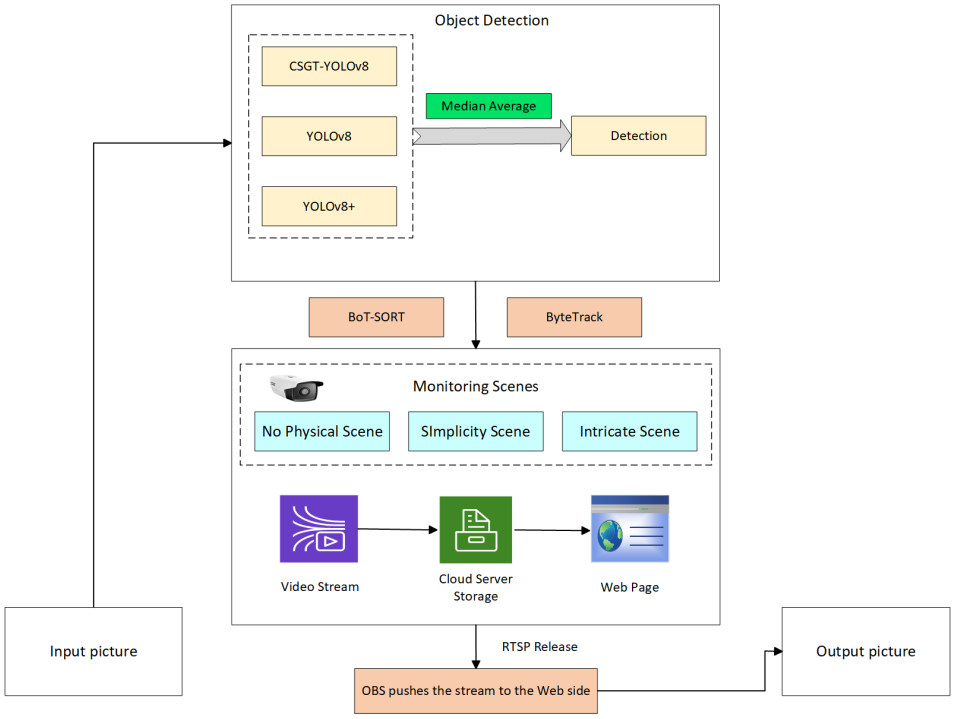


Figure 11. Baseline model YOLOv8 and SGST-YLOV8 visualization results.

This study evaluated the models by obtaining the FPS from three different surveillance footage scenarios (Table 4). To accurately evaluate the operating efficiency of the model, multiple counts were performed and the median average method was used to determine the final result of each scenario. This method can better capture the performance of the model in different scenarios. During the evaluation process, the experiment selected three representative monitoring screen scenarios and recorded the FPS values for each scenario. These scenarios include complex outdoor environments, simple outdoor scenes, and static scenes without physical objects for monitoring. By conducting multiple counts for each scenario, the experiment was able to obtain more reliable results and eliminate the influence of occasional outliers on the evaluation outcomes. It can be seen from the results that in the static scene without physical objects, the FPS difference between the improved YOLOv8 model, other enhanced models, and the baseline YOLOv8 model is minimal, with detection efficiency remaining relatively consistent. When transitioning to a simple scenario, a certain decrease in the detection efficiency of the model can be observed. After median averaging, the FPS of the improved model was 2.6 FPS higher than the baseline model. However, when entering the complex outdoor scene, the model's ability to handle complexity began to weaken. Compared to the simple scene, the FPS generally dropped by around 50%. The improved SGST-YOLOv8 model that takes GSConv+Triplet Attention as its core has better performance than other models. This result is crucial for practical applications, as it indicates that the model can effectively detect targets even in real-world environments.

Table 4. Scenario PFS for each model in the three scenarios.

Scenario	Model	FPS
No Physical Scene	CSGT-YOLOv8	44.7
	YOLOV8	44.4
	YOLOV8+ [1]	42.9–43.6
Simple Scene	CSGT-YOLOv8	34.9
	YOLOV8	32.3
	YOLOV8+	28.1–31.4
Intricate Scene	CSGT-YOLOv8	22.9
	YOLOV8	17.6
	YOLOV8+	15.8–16.8

[1] YOLOv8+ means that the baseline model introduces modules such as other attention mechanisms such as CNeB, ECA-Net, and GAM.

5. Discussion

This research is based on campus safety hazards, using object detection and object tracking algorithms as technical methods to achieve the combination of theoretical algorithm improvements and practical applications. Looking at the existing literature, this is the first research to use an improved YOLOv8 model combined with a target tracking algorithm for real-time detection and recognition of pedestrians and vehicles based on web-based campus surveillance. Typically, using algorithms for object recognition and tracking in surveillance helps managers efficiently identify abnormal situations. However, the accuracy of the object detection algorithm and the inference speed after deployment can affect the results.

This paper presents an improved lightweight model named SGST-YOLOv8, which replaces the original module with a GSConv backbone to reduce the overall complexity of the model. At the same time, adding the Triplet Attention mechanism module to the head section enhances the target representation and integrates contextual information, reducing background interference to improve classification accuracy. The method achieved an accuracy of 88.6% on the pedestrian and vehicle dataset and an FPS increase of 3–5 fps compared to the baseline model on real-time surveillance videos. The model is superior to YOLOv8n in parameters, GFLOPS, and Precision, indicating that the model is a practical real-time pedestrian and vehicle detection algorithm. Therefore, the SGST-YOLOv8 model and deployment method framework proposed in this study can help subsequent scholars to carry out related research and provide ideas and directions for them.

Although this improved model reduces the overall complexity of the target monitoring model and adds the Triplet Attention mechanism to improve the classification accuracy, the model's ability to monitor small targets still needs to be improved. In future research, follow-up experiments will further optimize the algorithm to improve the accuracy and efficiency of the model, and explore its integration with tasks such as target tracking and semantic segmentation to meet the evolving application requirements.

The current work's main contributions are compared with those reported in existing technologies, as shown in Table 5.

From the table, it can be seen that most of the current research in this field focuses on offline processing and object recognition for static targets, while there has been little in-depth exploration of subsequent target path tracking. In terms of practical application, previous research primarily conducted in-depth studies in the scientific domain without performing relevant experiments for actual deployment. This can be seen from the deployment types and detection targets. Previous research has proposed many excellent and practical improved algorithms. However, new algorithms continue to emerge in the field of technology and object detection. Therefore, this study presents a lightweight improved model, SGST-YOLOv8, providing a new algorithm direction for future research.

Table 5. Compare the results of this experimental method with contributions from the current stage.

Scenario	Method	Type of Deployment	Target of Detection
City street	Existing methods such as RANSAC	Offline processing (not real-time)	Static (no tracking)
Campus road	YOLOv7 target detection algorithm	Offline processing (not real-time)	Static and Dynamic (no tracking)
City street	Improved Siamese tracking algorithm	Offline processing (not real-time)	Static and Dynamic (tracking)
Campus road	Improved YOLOv8 [1] bot_sort	Online processing (real-time)	Static and Dynamic (tracking)

[1] This scenario provides the experimental methodology and related techniques for this study.

6. Conclusions

Although the SGST-YOLOv8 improved model performs well in terms of accuracy and lightweight design, its effectiveness in practical deployment is not outstanding. The improved model only considers the detection of medium to large objects such as vehicles and pedestrians on campus, without testing and optimizing its capability for small object detection. Additionally, the experiment used a single dataset of pedestrians and vehicles, without utilizing a large number of public datasets to further validate the model's generalization ability.

Therefore, future research will further optimize the algorithm to enhance the model's accuracy and efficiency, as well as improve its capability to detect small objects. Additionally, we will explore integration with other tasks, such as target tracking and semantic segmentation, to meet the evolving application needs. Subsequent experiments will also include multiple datasets to test and evaluate the model, enhancing its generalization and reliability.

Author Contributions: Conceptualization, P.C. and G.C.; methodology, P.C. and G.C.; validation, P.C.; formal analysis, P.C.; investigation, P.C.; writing—original draft preparation, P.C.; writing—review and editing, G.C.; visualization, P.C.; supervision, H.D.; project administration, G.C. and J.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This study was supported by the Fundamental Research Funds for the Universities of Henan Province (NSFRF180329), the MOE (Ministry of Education in China) Project of Humanities and Social Sciences (15YJCZH018), and the Science and Technology Project of Henan Province (162102210063).

Data Availability Statement: The data used in this study can be obtained from the first author at 17739147215@163.com with a reasonable request.

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

References

1. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–26 June 2005; Volume 1, pp. 886–893.
2. Lowe, D.G. Object recognition from local scale-invariant features. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999; Volume 2, pp. 1150–1157.
3. Ho, J.; Jain, A.; Abbeel, P. Denoising diffusion probabilistic models. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 6840–6851.
4. Wei, Y.; Tian, Q.; Guo, J.; Huang, W.; Cao, J. Multi-vehicle detection algorithm through combining Harr and HOG features. *Math. Comput. Simul.* **2019**, *155*, 130–145. [CrossRef]
5. Zhou, H.; Yu, G. Research on pedestrian detection technology based on the SVM classifier trained by HOG and LTP features. *Future Gener. Comput. Syst.* **2021**, *125*, 604–615. [CrossRef]
6. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1137–1149. [CrossRef] [PubMed]
7. Cai, Z.; Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162.

8. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
9. Sirisha, U.; Praveen, S.P.; Srinivasu, P.N.; Barsocchi, P.; Bhoi, A.K. Statistical analysis of design aspects of various YOLO-based deep learning models for object detection. *Int. J. Comput. Intell. Syst.* **2023**, *16*, 126. [CrossRef]
10. Zeng, L.; Sun, B.; Zhu, D. Underwater target detection based on Faster R-CNN and adversarial occlusion network. *Eng. Appl. Artif. Intell.* **2021**, *100*, 104190. [CrossRef]
11. Zhong, L.; Li, J.; Zhou, F.; Bao, X.; Xing, W.; Han, Z.; Luo, J. Integration Between Cascade Region-Based Convolutional Neural Network and Bi-Directional Feature Pyramid Network for Live Object Tracking and Detection. *Trait. Du Signal* **2021**, *38*, 1253. [CrossRef]
12. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
13. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland, 2016; pp. 21–37.
14. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
15. Hu, H.; Kaizu, Y.; Zhang, H.; Xu, Y.; Imou, K.; Li, M.; Huang, J.; Dai, S. Recognition and localization of strawberries from 3D binocular cameras for a strawberry picking robot using coupled YOLO/Mask R-CNN. *Int. J. Agric. Biol. Eng.* **2022**, *15*, 175–179. [CrossRef]
16. Miao, Y.; Zhang, S.; He, S. Real-time detection network SI-SSD for weak targets in complex traffic scenarios. *Neural Process. Lett.* **2022**, *54*, 3235–3247. [CrossRef]
17. Liang, H.; Yang, J.; Shao, M. FE-RetinaNet: Small target detection with parallel multi-scale feature enhancement. *Symmetry* **2021**, *13*, 950. [CrossRef]
18. Wu, Y.; Han, Q.; Jin, Q.; Li, J.; Zhang, Y. LCA-YOLOv8-Seg: An improved lightweight YOLOv8-Seg for real-time pixel-level crack detection of dams and bridges. *Appl. Sci.* **2023**, *13*, 10583. [CrossRef]
19. Li, S.; Huang, H.; Meng, X.; Wang, M.; Li, Y.; Xie, L. A glove-wearing detection algorithm based on improved YOLOv8. *Sensors* **2023**, *23*, 9906. [CrossRef]
20. Song, X.; Cao, S.; Zhang, J.; Hou, Z. Steel Surface Defect Detection Algorithm Based on YOLOv8. *Electronics* **2024**, *13*, 988. [CrossRef]
21. Ma, S.; Lu, H.; Liu, J.; Zhu, Y.; Sang, P. LAYN: Lightweight Multi-Scale Attention YOLOv8 Network for Small Object Detection. *IEEE Access* **2024**. . [CrossRef]
22. Hoang, M.L. Smart Drone Surveillance System Based on AI and on IoT Communication in Case of Intrusion and Fire Accident. *Drones* **2023**, *7*, 694. [CrossRef]
23. Li, H.; Li, J.; Wei, H.; Liu, Z.; Zhan, Z.; Ren, Q. Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles. *arXiv* **2022**, arXiv:2206.02424.
24. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
25. Chen, J.; Wang, X.; Guo, Z.; Zhang, X.; Sun, J. Dynamic region-aware convolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 8064–8073.
26. Jiang, T.; Chen, S. A Lightweight Forest Pest Image Recognition Model Based on Improved YOLOv8. *Appl. Sci.* **2024**, *14*, 1941. [CrossRef]
27. Li, M.; Chen, S.; Sun, C.; Fang, S.; Han, J.; Wang, X.; Yun, H. An Improved Lightweight Dense Pedestrian Detection Algorithm. *Appl. Sci.* **2023**, *13*, 8757. [CrossRef]
28. Misra, D.; Nalamada, T.; Arasanipalai, A.U.; Hou, Q. Rotate to attend: Convolutional triplet attention module. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 5–9 January 2021; pp. 3139–3148.
29. Qiang, H.; Tao, Z.; Ye, B.; Yang, R.; Xu, W. Transmission Line Fault Detection and Classification Based on Improved YOLOv8s. *Electronics* **2023**, *12*, 4537. [CrossRef]
30. Bodla, N.; Singh, B.; Chellappa, R.; Davis, L.S. Soft-NMS–improving object detection with one line of code. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5561–5569.
31. Symeonidis, C.; Mademlis, I.; Pitas, I.; Nikolaidis, N. Neural attention-driven non-maximum suppression for person detection. *IEEE Trans. Image Process.* **2023**, *32*, 2454–2467. [CrossRef]
32. Alom, M.Z.; Taha, T.M.; Yakopcic, C.; Westberg, S.; Sidike, P.; Nasrin, M.S.; Hasan, M.; Van Essen, B.C.; Awwal, A.A.; Asari, V.K. A state-of-the-art survey on deep learning theory and architectures. *Electronics* **2019**, *8*, 292. [CrossRef]
33. Narejo, S.; Pandey, B.; Esenarro Vargas, D.; Rodriguez, C.; Anjum, M.R. Weapon detection using YOLO V3 for smart surveillance system. *Math. Probl. Eng.* **2021**, *2021*, 9975700. [CrossRef]