Journal of Electronics and Information Science (2025)
Clausius Scientific Press, Canada

DOI: 10.23977/jeis.2025.100215
ISSN 2371-9524 Vol. 10 Num. 2

Real-Time Pedestrian Detection System Based on YOLOv5-tiny

Ranning Deng

School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, China
2840820473@qq.com

Keywords: Computer Vision; YOLOv5-tiny; Pedestrian Detection; Lightweight Model

Abstract: Real-time pedestrian detection, as a key technology in the field of computer vision, has broad application demands in intelligent surveillance, autonomous driving, robot navigation, and other areas. To address the problem that high-computational-power models are difficult to deploy on edge devices, this paper proposes a real-time pedestrian detection scheme based on the lightweight YOLOv5-tiny model. The study uses a pedestrian subset of the COCO dataset for model training, optimizes the anchor box dimensions through the K-means clustering algorithm to adapt to pedestrian target characteristics, and tests the model performance on ordinary CPU and GPU environments. Experimental results show that the optimized model can achieve a detection speed of 23.6 FPS with a recall rate of 82.3% on the Intel Core i7-10700 CPU; on the NVIDIA GTX 1650 GPU, the frame rate increases to 45.2 FPS and the recall rate rises to 84.7%, which can meet the real-time and detection accuracy requirements in low-computational-power scenarios.

1. Introduction

With the advancement of urbanization and the popularization of intelligent devices, pedestrian detection technology has become increasingly important in public security, traffic management, and other fields. The balance between real-time performance and detection accuracy is the core challenge for the practical application of this technology. Traditional deep learning models such as Faster R-CNN can achieve high detection accuracy, but their complex network structures result in slow inference speeds, making it difficult to meet real-time requirements. Lightweight models, while advantageous in speed, often suffer from reduced detection performance due to limited feature extraction capabilities.

The YOLO series models have demonstrated outstanding performance in real-time object detection due to their "end-to-end" detection approach. Among them, YOLOv5-tiny, as a lightweight version, achieves efficient inference by simplifying the network structure and reducing the number of parameters. However, its default anchor box dimensions are designed for the full-category COCO dataset, leading to insufficient adaptability to targets with specific scale characteristics such as pedestrians (vertical and relatively fixed aspect ratio). In addition, most existing studies focus on model optimization in high-performance GPU environments, with

relatively few performance tests and optimizations for ordinary CPUs or low-power GPUs.

Centering on the demand for real-time pedestrian detection in low-computational-power scenarios, this paper conducts three aspects of work based on the YOLOv5-tiny model: first, constructing a pedestrian subset of the COCO dataset to reduce the interference of non-target categories on training; second, optimizing anchor box parameters based on the scale characteristics of pedestrian targets to improve the positioning accuracy of the model for pedestrians; third, conducting system tests on ordinary CPU and GPU platforms to verify the practical value of the model in low-computational-power environments, providing technical references for edge device deployment.

2. Related Work

The development of pedestrian detection technology has gone through two stages: traditional methods and deep learning methods. Early studies mostly combined manually designed features with classifiers. For example, Dalal et al. proposed the HOG (Histogram of Oriented Gradients) feature combined with an SVM (Support Vector Machine) classifier, which describes pedestrian contour features through histograms of gradient directions and achieved certain results in simple scenarios. However, it has poor robustness to occlusion, illumination changes, and other factors.

The rise of deep learning methods has promoted a leap in pedestrian detection performance. Two-stage models based on region proposals (such as Faster R-CNN) achieve high detection accuracy by generating candidate regions and performing fine-grained classification, but multi-stage inference limits their speed. One-stage models directly realize end-to-end detection by regressing target coordinates and category probabilities, among which the YOLO series is renowned for its efficiency. YOLOv1 first transformed object detection into a regression problem, YOLOv3 introduced multi-scale feature fusion and residual structures, and YOLOv5 further optimized network structure, data augmentation, and other aspects, forming multiple versions including YOLOv5-tiny. Horvat et al. conducted a comparative study on the performance of different YOLOv5 models in image localization and classification, providing a reference for the selection and optimization of YOLOv5 variants in specific tasks[2]. For dense object detection scenarios, Lin et al. proposed the Focal Loss to address the class imbalance problem, which has inspired the design of loss functions in subsequent pedestrian detection models[5].

Research on lightweight models is crucial for solving the problem of low-computational-power deployment. The MobileNet series reduces the number of parameters using depthwise separable convolution, EfficientNet balances accuracy and efficiency through a compound scaling strategy, while YOLOv5-tiny significantly improves inference speed while ensuring a certain level of accuracy by simplifying the backbone network and neck structure. Bochkovskiy et al. proposed YOLOv4 and its lightweight version YOLOv4-tiny, which also adopts similar lightweight design ideas and is widely used in real-time detection tasks[1]. Ye et al. further optimized the lightweight YOLOv5 framework (Light-YOLOv5) for PCB defect detection, verifying the adaptability of lightweight YOLO models in specific industrial scenarios—this provides a reference for our optimization of YOLOv5-tiny for pedestrian detection[4]. However, the weakened feature extraction capability caused by lightweight design may lead to decreased detection performance for small targets and occluded targets, thus requiring task-specific optimization.

Anchor box design has a significant impact on detection performance. The YOLO series uses preset anchor boxes as prior knowledge of target dimensions. The original anchor boxes are obtained by clustering full-category COCO data, resulting in insufficient adaptability to single categories such as pedestrians. Existing studies have shown that optimizing anchor boxes for specific targets can significantly improve detection accuracy. For example, Ren et al. redefined

anchor box dimensions through K-means clustering, increasing the pedestrian detection recall rate by 5.3%, which provides ideas for the anchor box optimization in this paper. Luo et al. proposed a lightweight YOLOv5-FFM model for occlusion pedestrian detection, emphasizing that task-specific model adjustments (such as anchor box optimization and feature fusion) are key to improving detection performance in complex scenarios[3].

3. Research Methods

3.1 Dataset Construction

The experiment uses a pedestrian subset of the COCO 2017 dataset for training and testing. The COCO dataset includes 80 target categories, among which the "person" category contains a large number of pedestrian samples covering different scenarios (streets, shopping malls, indoor environments, etc.), postures (standing, walking, squatting), and environmental conditions (illumination, occlusion), making it highly representative.

To focus on the pedestrian detection task, 12,863 images containing the "person" category are selected from the COCO training set, corresponding to 38,562 pedestrian annotation boxes; 2,143 images are selected from the validation set, corresponding to 6,429 annotation boxes. To enhance the generalization ability of the model, data augmentation is performed on the training set, including random horizontal flipping (probability 0.5), random cropping (scaling range 0.2-1.0), brightness/contrast adjustment (± 0.2), and mosaic data augmentation (randomly stitching 4 images) to simulate different shooting angles and environmental changes.

3.2 Model Structure Design

Targeted adjustments are made based on the original YOLOv5-tiny structure. The model consists of three parts: backbone network, neck network, and detection head. The backbone network adopts the CSPDarknet structure, extracting image features through convolutional layers and residual blocks. Compared with other versions of YOLOv5, YOLOv5-tiny reduces the number of residual blocks and the scale of convolutional kernels, significantly decreasing the number of parameters. The neck network uses the PANet structure to achieve multi-scale feature fusion through upsampling and downsampling, enhancing the detection capability for pedestrians of different sizes. The detection head outputs target category probabilities, confidence scores, and bounding box coordinates, and uses the GIOU (Generalized Intersection over Union) loss function to optimize bounding box regression.

To adapt to the pedestrian detection task, two improvements are made to the model: first, adding an attention mechanism (SE module) to the last layer of the backbone network, which strengthens the weights of key feature channels through squeeze-and-excitation operations, improving the extraction capability of discriminative features such as pedestrian contours and postures; second, adjusting the output scales of the detection head, retaining three feature map scales of 160×160, 320×320, and 640×640, corresponding to small, medium, and large-sized pedestrian targets respectively, to avoid missed detections caused by a single scale.

3.3 Anchor Box Optimization Method

The K-means clustering algorithm is used to redefine the anchor box dimensions. The original YOLOv5-tiny anchor boxes are generated based on full-category COCO data, including 6 anchor boxes ([10,13], [16,30], [33,23], [30,61], [62,45], [59,119]). Their aspect ratio distribution is wide, leading to limited adaptability to pedestrians (typical aspect ratio of approximately 1:3).

The clustering process takes the aspect ratio of annotation boxes as features and uses Euclidean distance as the similarity metric. The steps are as follows: 1) Extract the width (w) and height (h) of all pedestrian targets from the training set annotation boxes and normalize them to the 640×640 image size; 2) Randomly select 6 initial cluster centers; 3) Calculate the distance between each annotation box and the cluster centers, and assign it to the nearest cluster; 4) Update the cluster center as the mean width and height of all annotation boxes in the cluster; 5) Repeat steps 3-4 until the cluster centers stabilize (variation <0.01).

Finally, the optimized anchor box dimensions are obtained as [15,36], [22,58], [30,82], [42,110], [56,145], [72,188], with aspect ratios concentrated between 1:2.4 and 1:2.6, which are more consistent with the scale characteristics of pedestrians, helping to improve the positioning accuracy of the model for pedestrian targets.

3.4 Training and Evaluation Settings

Model training is implemented based on the PyTorch framework, with hardware environment including Intel Core i7-10700 CPU and NVIDIA GTX 1650 GPU (4GB memory), and software environment including Python 3.8 and CUDA 11.2. The training parameters are set as follows: initial learning rate of 0.01, SGD optimizer (momentum 0.937, weight decay 0.0005), cosine annealing learning rate scheduling strategy, 100 training epochs, and batch size of 16.

The evaluation metrics include: 1) Frames Per Second (FPS), measuring real-time performance, i.e., the number of images processed per second; 2) Recall rate, calculating the proportion of detected real pedestrians to all real pedestrians, reflecting the model's missed detection situation; 3) Precision rate, calculating the proportion of real pedestrians in the detection results, reflecting the model's false detection situation; 4) mean Average Precision (mAP@0.5), the average precision at an Intersection over Union (IoU) threshold of 0.5, comprehensively measuring detection performance.

4. Experimental Results and Analysis

4.1 Experimental Environment and Baseline Models

To verify the performance in low-computational-power scenarios, experiments are conducted on two hardware platforms: the CPU platform is Intel Core i7-10700 (8 cores, 16 threads, main frequency 2.9GHz) without GPU acceleration; the GPU platform is NVIDIA GTX 1650 (4GB GDDR6 memory) with CUDA acceleration. The comparison models include: 1) Original YOLOv5-tiny (without anchor box optimization); 2) YOLOv5-tiny with optimized anchor boxes proposed in this paper (referred to as YOLOv5-tiny-Person); 3) Lightweight models YOLOv4-tiny and MobileNet-SSD to demonstrate the superiority of the proposed scheme.

4.2 Performance Comparison Results

The performance metrics of different models on the validation set are shown in Table 1. On the CPU platform, the YOLOv5-tiny-Person achieves a frame rate of 23.6 FPS, meeting real-time requirements (usually ≥20 FPS), with a recall rate of 82.3%, which is 4.1 percentage points higher than the original YOLOv5-tiny, indicating that anchor box optimization effectively reduces missed detections. Compared with YOLOv4-tiny, the frame rate increases by 3.2 FPS and the recall rate increases by 2.8 percentage points; compared with MobileNet-SSD, the frame rate is similar but the recall rate increases by 8.5 percentage points, showing stronger pedestrian detection capability.

On the GPU platform, the frame rate of YOLOv5-tiny-Person increases to 45.2 FPS, and the

recall rate further rises to 84.7%. This is because the parallel computing capability of the GPU accelerates the feature extraction process, and more sufficient computing resources support more precise bounding box regression. Compared with other models, its frame rate is higher than that of YOLOv4-tiny (38.5 FPS) and MobileNet-SSD (29.8 FPS), and the mAP@0.5 reaches 79.6%, which is 3.8 percentage points higher than the original model, verifying the effectiveness of the optimization scheme.

Table 1 Performance Comparison of Different Models on CPU and GPU Platforms

Model	Hardware Platform	Frame Rate (FPS)	Recall Rate (%)	Precision Rate (%)	mAP@0.5 (%)
Original YOLOv5-tiny	CPU	22.8	78.2	81.5	75.8
YOLOv5-tiny-Person	CPU	23.6	82.3	83.2	79.2
YOLOv4-tiny	CPU	20.4	79.5	80.1	76.5
MobileNet-SSD	CPU	23.1	73.8	78.6	71.3
Original YOLOv5-tiny	GPU	43.5	80.9	82.8	75.8
YOLOv5-tiny-Person	GPU	45.2	84.7	85.1	79.6
YOLOv4-tiny	GPU	38.5	81.2	81.5	77.2
MobileNet-SSD	GPU	29.8	76.3	79.2	72.5

4.3 Ablation Experiment Analysis

To independently verify the effects of anchor box optimization and the attention mechanism, ablation experiments are designed (Table 2). The results show that when only anchor boxes are optimized, the recall rate of the model on the CPU increases from 78.2% to 81.7%, and the mAP@0.5 increases by 3.1 percentage points, indicating that anchor boxes adapted to pedestrian features can effectively improve positioning accuracy; when only the SE module is added, the recall rate increases by 1.8 percentage points, illustrating that the attention mechanism enhances the extraction capability of key features; when both are combined, the recall rate further increases to 82.3%, verifying the synergistic effect of multiple optimization strategies. In addition, the two optimizations have little impact on the frame rate (<1 FPS), indicating that the scheme achieves a good balance between accuracy and speed.

Table 2 Ablation Experiment Results (CPU Platform)

Optimization Strategy	Cardboard	Glass	Metal
No Optimization	22.8	78.2	75.8
Anchor Box Optimization Only	23.0	81.7	78.9
SE Module Only	22.7	80.0	77.2
Anchor Box + SE Module	23.6	82.3	79.2

4.4 Detection Effect in Typical Scenarios

Three types of typical scenarios are selected for visual analysis: 1) Dense crowd scenarios (such as shopping mall corridors), YOLOv5-tiny-Person can accurately detect pedestrians with high overlap, and the missed detection rate is reduced by about 6% compared with the original model; 2) Small target scenarios (such as distant pedestrians), the optimized anchor boxes enhance the sensitivity to small-sized targets, and the number of detections increases by about 12%; 3) Complex background scenarios (such as pedestrians mixed with vehicles on streets), the SE module reduces the false detection rate (classifying non-pedestrian targets as pedestrians) by about 4% by

strengthening pedestrian feature channels. The above results indicate that the model has good adaptability in diverse scenarios.

5. Discussion

Experimental results show that the optimized scheme based on YOLOv5-tiny achieves an effective balance between real-time performance and detection accuracy on low-computational-power devices. Anchor box optimization significantly improves the recall rate by adapting to the scale characteristics of pedestrians. This is because the intersection over union (IoU) between the clustered anchor boxes and pedestrian annotation boxes is higher, reducing the search space for the model during bounding box regression. The introduction of the SE module dynamically adjusts feature weights, enhancing the model's ability to recognize key pedestrian features (such as the head and torso), especially in occluded scenarios.

Compared with existing lightweight models, YOLOv5-tiny-Person has advantages in three aspects: first, higher frame rate, meeting the real-time requirements of edge devices; second, better recall rate for pedestrians, reducing the risk of missed detections in security monitoring and other scenarios; third, small model size (about 14MB), facilitating deployment on devices with limited storage resources.

The research has two limitations: first, the detection accuracy for extremely small targets (such as pixel size $<32\times32$) still has room for improvement, mainly due to the weak high-resolution feature extraction capability of lightweight models; second, the model performance fluctuates significantly under strong or low illumination conditions, and the data augmentation strategy needs further optimization. In the future, the detection capability for small targets can be improved by introducing multi-scale feature fusion networks (such as improved FPN+PAN structures), and the robustness of the model to illumination changes can be enhanced by combining image enhancement algorithms (such as Retinex).

6. Conclusion

The real-time pedestrian detection system based on YOLOv5-tiny proposed in this paper achieves efficient pedestrian detection on low-computational-power devices by constructing a COCO pedestrian subset, optimizing anchor box dimensions, and introducing an attention mechanism. Experimental results show that the system achieves a frame rate of 23.6 FPS with a recall rate of 82.3% on an ordinary CPU, and the frame rate increases to 45.2 FPS with a recall rate of 84.7% on an entry-level GPU, which is significantly superior to comparison models. The study verifies the optimization potential of lightweight models in specific target detection tasks, providing a feasible scheme for edge device deployment.

Future work will focus on model compression and scenario-adaptive optimization, further reducing the model size through knowledge distillation, and improving the model's detection performance in specific scenarios (such as night monitoring) by combining transfer learning, promoting the application of real-time pedestrian detection technology in a wider range of low-computational-power scenarios.

References

[1] Bochkovskiy, Alexey, Chien-Yao Wang, and Hong-Yuan Mark Liao. "Yolov4: Optimal speed and accuracy of object detection." arxiv preprint arxiv:2004.10934 (2020).

[2] Horvat, Marko, Ljudevit Jelečević, and Gordan Gledec. "A comparative study of YOLOv5 models performance for image localization and classification." Central European Conference on Information and Intelligent Systems. Faculty of Organization and Informatics Varazdin, 2022.

[3] Luo, Xiangjie, et al. "A lightweight YOLOv5-FFM model for occlusion pedestrian detection." arxiv preprint arxiv:2408.06633 (2024).
[4] Ye, Meng, Hao Wang, and Hang Xiao. "Light-YOLOv5: A lightweight algorithm for improved YOLOv5 in PCB defect detection." 2023 IEEE 2nd International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA). IEEE, 2023.
[5] Lin, Tsung-Yi, et al. "Focal loss for dense object detection." Proceedings of the IEEE international conference on computer vision. 2017.