Dense Pedestrian Detection Based on YOLO-V4 Network Reconstruction and CIoU Loss Optimization

Guoliang Zhang¹, Zexu Du¹, Weijiang Lu¹, Xiaoyan Meng¹,*

¹Global Energy Interconnection Research Institute Co.Ltd, Artificial Intelligence on Electric Power System State Grid Corporation Joint Laboratory, Beijing, China
*email: duzexu@geiri.sgcc.com.cn

Abstract. Online object detection is a fundamental problem in time-critical video analysis applications. Due to the performance limitations of one-stage object detection algorithms in dense pedestrian occlusion, we have improved YOLO-V4 in this paper, including network structure optimization, more efficient multi-scale feature fusion strategy formulation, and more specialized network loss function design. First, a single-output YOLO-V4 network structure is proposed, which integrates image information from multiple scales through the designed ladder fusion strategy. This not only ensures that the aspect ratio estimation of anchors is still driven by training data but also solves the invalid anchor distribution problem of the original network for objects with approximate size. Second, we adjust the resolution ratio of the network output feature map to the original input image to reduce the label rewriting cases of training samples. Finally, the concept of repulsive force is introduced to optimize the bounding box regression loss function, which improves the robustness of the model to the detection of densely occluded pedestrians, and enhances the practical value of YOLO-V4 in actual application scenarios.

1. Introduction
Pedestrian detection is a more specific object detection problem. The objects of this task are pedestrians in a traffic environment. It can be combined with technologies such as pedestrian tracking and pedestrian re-recognition, and applied to auto-driving, passenger flow statistics, intelligent transportation and other fields.

At present, deep learning-based models are mainly divided into anchor-based methods and anchor-free methods. Both models include a backbone network for feature extraction and a head network for classification and regression. The backbone network is generally some typical SOTA networks, such as ResNet [1] or ResNext [2], etc., which need to be pre-trained on ImageNet or Open Images. In addition, there are some methods [3,4] that are trained from scratch. Considering the real-time requirements in the actual scene, this paper is mainly based on the one-stage target detection algorithm to solve the problem of intensive pedestrian detection.

OverFeat [5] is one of the earliest one-stage object detectors based on deep learning. It is based on a multi-scale sliding window strategy and uses only one shared CNN to simultaneously complete image classification, object positioning and object detection tasks. Later, Redmon et al. [6] regards object detection as a regression problem and proposed a real-time detector YOLO (You Only Look Once, YOLO), which can achieve 45FPS inference speed. If a simpler backbone network is used, the inference speed can reach 155FPS. Liu et al. [7] proposed a network structure called SSD (Single-Shot Mulibox Detector, SSD) by learning from the ideas of MultiBox [8], area proposal network and multi-scale output, which is more accurate and efficient than YOLO. In addition, Zhang et al. [9] proposed the RefinedDet architecture, which performs a two-step regression on the bounding box, and the second step regression is used to optimize the results of the first step, thereby obtaining more...
accurate detection results, especially for small objects.

Based on the one-stage object detection framework, the main contributions and innovations of this paper are summarized as follows:

- We designed a single-output YOLO-V4 network structure to solve the problem of label rewriting and invalid Anchor assignment when the original algorithm is applied to the pedestrian detection tasks.
- In order to achieve the integration of the output feature maps at different stages to enrich the pedestrian information contained in a single output layer, a multi-scale ladder fusion strategy on the feature maps is designed.
- A more specific network loss function is designed according to the detection requirements of actual traffic scene for densely occluded pedestrians.

2. Network structure limitations for YOLO-V4

There are two obvious problems in applying YOLO-V4 directly to pedestrian detection tasks, namely label rewriting and invalid anchor allocation. Solving the above problems is essential to improve the performance of YOLO-V4 in pedestrian detection.

1) Problem of label rewriting

Label rewriting means that due to the unique anchor mechanism of YOLO-V4, if two targets with similar width and height are located in the same grid and are assigned to the same anchor box, the label of the previous target will be rewritten by the next target, and one of the targets will be unable to participate in the training process because the label is rewritten into the background. This kind of processing will make the model learn wrong image information and ignore some dense objects, resulting in a decrease in the number of true positive detection results. The effect of the label rewriting problem is shown in Figure 1. The red bounding box in the figure indicates that the target label is rewritten, and the green indicates that the label is reserved, 13 of the 26 labels in the figure are rewritten.

2) Problem of invalid anchors distribution

According to the anchor mechanism of YOLO-V4, the clustering algorithm is used to obtain 9 anchor boxes that conform to a specific sample distribution, and a group of three aspect ratios is used as the default anchor for large output images, medium output images, and small output layers. In the inference stage, objects of different sizes are assigned to different output layers for prediction. However, unlike the rich scale diversity of the COCO data-set for natural scenes, the size of pedestrians in traffic scenes is usually similar, or there are only a few specific scales. In this case, the application of the above anchor mechanism will result in the similar targets are forced to be assigned to different output layers for prediction, which is obviously unreasonable.

![Fig. 1 Diagram of label rewriting in YOLO-V4 for pedestrian detection](image)

3. Pedestrian detection in traffic scene based on improved YOLO-V4

Here, we will optimize the network structure for the label rewriting and invalid anchor distribution by YOLO-V4 in pedestrian detection, and design a multi-scale feature fusion strategy and a more specific network loss function based on the requirements of actual traffic scene detection.

3.1. Network structure optimization

We will optimize the overall network structure for the label rewriting and invalid anchor distribution problems in the algorithm. The specific considerations are as follows.

1) Output feature map resolution setting

Assuming that the ratio of the network output image resolution to the input image resolution is $s_k$, there will be no label rewriting problem when $s_k=1$. This situation usually exists in segmented neural networks based on encoder-decoders [10]. Since this type of structure contains more deconvolution
operations, the computational efficiency is relatively low and cannot meet the real-time requirements of pedestrian detection. Besides, increasing the resolution of the output feature map can also alleviate label rewriting effectively. In the case of \( s_k < 1 \). Since increasing the resolution of the input image to indirectly increase the resolution of the output feature map will consume more computing resources and time costs in both training and inference, it is planned to directly increase the resolution of the output feature map to perform the detection network reconfiguration. The values of \( s_k \) corresponding to the three output layers of YOLO-V4 are \( s_1 = 1/8 \), \( s_2 = 1/1 \), and \( s_3 = 1/3 \). Then the optimized single network output layer \( s_k \) value will be set to 1/4.

2) Network output layer number setting

There are two solutions to the invalid anchor distribution problem. The first solution is to convert anchor box clustering from data-driven to problem-driven. First, define three scale ranges according to the receptive field of the network output layer, and set a threshold to forcibly discretize and separate the three scales. Then, the form of selecting bounding boxes that meet a specific range on each output scale to perform separate clustering replaces the non-scale difference clustering of the bounding boxes of the entire dataset. This solution can effectively avoid the problem of small targets being assigned to small output feature maps for training and large targets being assigned to large output feature maps for training. However, the object size in pedestrian detection is usually close, and only one output layer is assigned. In the target prediction task, the other two output scales cannot function, resulting in a waste of network resources. The second solution is that the network uses only one output layer to complete the prediction of all targets. The solution is simple and effective, and can significantly reduce the parameter amount of the model, which is more in line with the actual needs of pedestrian detection.

In summary, this article intends to use a single output layer with a zoom ratio of 1/4 to associate with all anchor boxes to solve the problems of label rewriting and invalid anchor distribution. The specific network structure is shown in Figure 2.

![Fig. 2](image)

**Fig. 2** The network structure of the improved YOLO-V4

3.2. Multi-scale feature fusion

To realize the integration of the output feature maps at different stages to enrich the pedestrian information contained in a single output layer, a multi-scale ladder fusion strategy on the feature maps is designed. Assuming that \( O \) is a set of feature maps, \( u(\cdot;\omega) \) represents the up-sampling of the input image by a factor of \( \omega \), \( m(\cdot) \) is the function converts the feature map of size \( a \times b \times \delta \) into the feature map of size \( a \times b \times \delta \), \( \delta \) is a constant, \( g(O_1,O_2,...,O_n) \) is an \( n \)-ary ensemble function, the final fusion feature map can be expressed as formula (1). Here, the add operation is used.
It can be seen from formula (1) that $O_2$ only needs to perform up-sampling with a factor of 2, while $O_k$ needs to perform up-sampling with a factor of $2^{k-1}$. There is an obvious imbalance in the quality of the feature maps generated by them, and the latter will generate a lot of information redundancy. To this end, we design a ladder fusion strategy as shown in Figure 3. Compared with the direct up-sampling method, the computational complexity of the model remains unchanged, but the output feature map obtained is smoother. The formula (1) can be modified into the form of the formula (2).

3.3. Loss function design for occlusion problem

Comparing with general target detection tasks, occlusion problems are more common in pedestrian detection tasks. The test results in [11] on the CityPersons [12] dataset show that 48.8% of the pedestrians in the 3157 pedestrian labelling boxes have an IoU higher than 0.1, and 26.4% of the pedestrians have an IoU higher than 0.3. To this end, this article will focus on solving the problem of model detection performance degradation caused by pedestrian self-occlusion from the perspective of the loss function based on the network architecture designed above.

The bounding box regression loss function proposed in this paper is shown in formula (3), $\lambda$ and $\theta$ are weight coefficients, and the first term $Loss_{Selcl}$ is the CIoU loss, which is used to make the prediction box closer to the target groundtruth (TG). The second item $L_{RepGT}$ is responsible for keeping the prediction box away from other surrounding groundtruth (SG) as far as possible. The third item $L_{RepBox}$ is responsible for keeping the prediction box as far away as possible from the prediction boxes of other surrounding SGs. Assuming that the prediction box is $P$, $P_z$ represents the prediction boxes set, the real target box is $G$, $g=\{G\}$ represents the real target boxes set, $G^r_{clu}$ is the real target box $G$ that matches $P$, $G^p_{Rep}$ is the $G$ with the largest IoU between $g$ and $P$ except $G^r_{clu}$, then $G^p_{Rep}$ and $L_{RepGT}$ can be expressed in the form of formulas (4) and (5), respectively.

$$\begin{align*}
    Loss_{reg} &= Loss_{Selcl} + \lambda \times L_{RepGT} + \theta \times L_{RepBox} \\
    G^p_{Rep} &= \arg \max_{G \in \{G\}} IoU(G, P) \\
    L_{RepGT} &= \frac{\sum_{P_z \in \{P|\} \text{Smooth}_{n}(IoG(P, G^p_{Rep}))}}{\text{Smooth}_{n}} \\
    \text{Smooth}_{n} &= \begin{cases} 
    -\ln(1-x) & x \leq \eta \\
    x - \eta - \ln(1-\eta) & x > \eta
    \end{cases} \\
    IoG(P, G) &= \frac{\text{area}(P \cap G)}{\text{area}(G)}
\end{align*}$$

The essence of $L_{RepGT}$ is to punish the overlapping relationship between $P$ and $G^p_{Rep}$. When the prediction frame is large enough, the use of IoU as a metric will directly lead to the decrease of $L_{RepGT}$, which is inconsistent with the original intention of designing $L_{RepGT}$. Therefore, the overlap relationship between $P$ and $G^p_{Rep}$ is redefined as IoG, as shown in formula (7), the denominator is the real bounding box area that cannot be changed, and only the numerator can be optimized, thus avoiding the drawbacks of using IoU directly. Formula (6) is continuously differentiable within (0,1), and $\eta \in [0,1]$ is a smoothing parameter used to adjust the sensitivity of repulsive force to outliers. When the overlap relationship between a prediction box and a non-matching target box is closer, the $L_{RepGT}$ can bring greater loss to the bounding box regression by formula (5), and effectively prevent the prediction box from moving to adjacent objects that are not its target.

For $L_{RepBox}$ we divide $P_z$ into $|g|$ subsets. Considering that the essence of $L_{RepBox}$ is to make the overlap relationship of prediction boxes sampled from any two subsets as weak as possible, $L_{RepBox}$ can be expressed as formula (8). $A$ is an identity function, and $\epsilon$ is a constant used to prevent the division by zero. When the IoU area between two prediction boxes that match different targets is smaller, the $L_{RepBox}$ will also decrease. To a certain extent, this prevents the predicted bounding boxes of two
different targets from being filtered out by the NMS algorithm due to the close distance and makes the
model more robust to the detection of dense crowds.

\[ L_{reg} = \frac{\sum_{ij} \text{Smooth}(IoU(P_i, P_j))}{\sum_{ij} [IoU(P_i, P_j) > 0] + \epsilon} \]  

(8)

4. Experiment

We use the standard CSPDarknet53 pre-trained on the ImageNet dataset released by YOLO-V4 as the
backbone network for all experiments, and update the overall network parameters of Im-YOLOV4 on
2 GPUs based on the SGD solver. During the training phase, specific parameters are set as follows:
batch size=64, subdivisions=16, momentum=0.949, weight decay=0.0005, max batches is related to
the maximum epoch=500. The CityPersons [12] are set to 24,000 according to the dataset size,
COCOPersons [13] takes 500,500, CrowdHuman [14] takes 120,000. The learning rate adjustment
strategy is “steps”, and the initial learning rate is 0.001. When the number of iterations reaches 300
and 400 epochs, the learning rate will decay to \(10^{-4}\) and \(10^{-6}\), respectively.

All experiments are run on Ubuntu 16.04 system, the computer hardware environment is 2.20
GHz×40 Intel Xeon(R) Silver 4114 CPU, 64 GB RAM and NVIDIA Titan Xp GPU×2; the network
operating environment is Python 3.7.9, OpenCV 4.4.0 and Darknet framework.

Tab. 1 Comparison of overall performance between YOLO-V4 and Im-YOLOV4

| Dataset        | YOLO-V4 | Im-YOLOV4 |
|----------------|---------|-----------|
|                | AP↑     | MR2↓      | AP↑     | MR2↓      |
| CrowdHuman     | 82.4%   | 49.5%     | 87.8%   | 44.6%     |
| CityPersons    | 93.7%   | 14.7%     | 95.5%   | 11.6%     |
| COCOPersons    | 82.9%   | 43.0%     | 84.4%   | 40.9%     |

Tab. 2 Ablation experiments and parameter sensitivity statistics on CrowdHuman and CityPersons

| Structure & Feature fusion? | Crowd Loss? | \( \lambda \) | \( \theta \) | CrowdHuman | CityPersons |
|----------------------------|-------------|---------------|--------------|------------|-------------|
|                            |             |               |              | AP↑        | MR2↓        |
| Baseline                   |             |               |              | 82.4%      | 49.5%       |
| \( \checkmark \)          | \( \checkmark \) | 0.3          | 0.7          | 86.9%      | 45.4%       |
| \( \checkmark \)          | \( \checkmark \) | 0.4          | 0.6          | 86.3%      | 45.6%       |
| \( \checkmark \)          | \( \checkmark \) | 0.5          | 0.5          | 87.3%      | 44.9%       |
| \( \checkmark \)          | \( \checkmark \) | 0.6          | 0.4          | 87.8%      | 44.6%       |

4.1. Overall performance testing

This section will test the performance of the proposed Im-YOLOV4 algorithm on CrowdHuman [14],
CityPersons [12], and COCOPersons [13]. The CrowdHuman dataset contains 15,000 training samples,
4,370 validation samples, and 5,000 testing samples. The CityPersons dataset contains a total of 5,000
images with a resolution of 1024×2048, including 2,975 training samples, 500 validation samples,
and 1,525 testing samples. For CrowdHuman and CityPersons, the training set and validation set are taken
in the experiment for end-to-end model training, and then the performance test is performed on the
testing set. COCOPersons is a subset of the MS COCO dataset, and its samples are the images with the
“person” frame in the original COCO. After filtering out the samples in the original dataset that are
only related to the remaining 79 categories, this paper uses the remaining 64,115 training samples for
model training, and 2,639 validation set samples for performance testing. TABLE 1 shows the
comparison of the test results between the proposed Im-YOLOV4 and the original YOLO-V4.
Im-YOLOV4 surpassed the original YOLO-V4 in both AP and MR2 indicators. The improved AP on
the CrowdHuman, CityPersons and COCOPersons datasets was increased by 5.4%, 1.8% and 1.5%
respectively. MR2 was reduced by 4.9%, 3.1%, and 2.1%, respectively.
4.2. Ablation study and parameter sensitivity analysis
We conducted ablation experiments and parameter sensitivity analysis of the loss function on the improved scheme of YOLO-V4 mentioned in Sec. III. The specific results are shown in TABLE II. The values of $\lambda$ and $\theta$ in the ablation experiment part in TABLE II can make the model performance the best. Considering that $MR^2$ has stronger versatility in the field of pedestrian detection, it is determined that the model performance is the best when $MR^2$ is the minimum value. In general, the model with “Structure & Feature fusion” improves the performance of “Basline” (i.e., YOLO-V4) more than “Crowd Loss”. On the CrowdHuman and CityPersons datasets, the former has increased by 3.1% and 1.3% in AP, and has decreased by 3.7% and 2.6% in $MR^2$. The latter increased by 2.4% and 0.9% in AP, and decreased by 2.2% and 1.9% in $MR^2$. When both “Structure & Feature fusion” and “Crowd Loss” are used at the same time, the performance of Im-YOLOV4 is improved compared to using either of them alone. For the parameter sensitivity, the influence of $\lambda$ and $\theta$ on the performance of the model lacks regularity, which may be related to the sample distribution of different datasets. For the CrowdHuman dataset, the corresponding optimal $\lambda$ and $\theta$ values are 0.6/0.4. For the CityPersons dataset, the corresponding optimal $\lambda$ and $\theta$ values are 0.5/0.5.

4.3. Comparison with the state-of-the-art algorithms
The performance of the proposed Im-YOLOV4 is compared with the state-of-the-art pedestrian detection algorithms, and the results are shown in TABLE III. Im-YOLOV4 has a higher AP than the five algorithms participating in the comparison on the CrowdHuman dataset, but the $MR^2$ is lower than CascadeR-CNN [15]. On the CityPersons and COCOPersons datasets, the performance of Im-YOLOV4 is worse than that of CrowdToOwn [14]. This is because the algorithm uniformly uses CrowdHuman samples to train the pre-training model, and then uses the training set samples of CityPersons and COCOPersons to fine-tune the parameters. This trick has been proved to be more conducive to the effective detection of dense objects.

| Methods          | CrowdHuman | CityPersons | COCOPersons |
|------------------|------------|-------------|--------------|
|                  | AP↑ | MR\text{-}2 | AP↑ | MR\text{-}2 | AP↑ | MR\text{-}2 |
| FPN[16]          | 85.6% | 55.9% | 94.4% | 14.8% | 83.8% | 41.9% |
| RetinaNet[17]    | 77.2% | 65.5% | -    | -    | -    | -    |
| RelationNet[18]  | 81.6% | 48.2% | -    | -    | -    | -    |
| GossipNet[15]    | 80.4% | 49.4% | -    | -    | -    | -    |
| CrowdToOwn[14]   | -    | -    | 95.6% | 10.7% | 85.0% | 39.8% |
| CascadeR-CNN[15] | 86.2% | 40.2% | -    | -    | -    | -    |
| RFRNN[15]        | -    | -    | 95.0% | 11.6% | -    | -    |
| Im-YOLOV4        | 87.8% | 44.6% | 95.5% | 11.6% | 84.4% | 40.9% |

Fig. 4 Comparison of visualization results for YOLO-V4 and Im-YOLOV4 on the CrowdHuman
4.4. Qualitative analysis
We have performed a qualitative analysis of the visual detection results of the proposed Im-YOLOV4 and original YOLO-V4 algorithms on the three datasets, and the display threshold of the bounding box is all set to 0.3. Due to space limitations, this paper only lists the comparison results of the algorithm on CrowdHuman in Figure 4. From Figure 4, the first and third lines are the original YOLO-V4 test results, and the second and fourth lines are the Im-YOLOV4 test results. The areas where the performance is significantly improved are marked with red circles.

5. Conclusion
In view of the problems faced by the YOLO-V4 algorithm in the task of pedestrian detection, we improved the original network structure and designed a multi-scale feature fusion strategy and a more specific network loss function. The experimental results of the constructed Im-YOLOV4 pedestrian detection framework on 3 benchmark datasets show that the Im-YOLOV4 algorithm can effectively improve the pedestrian detection performance of the original YOLO-V4, and has higher robustness and promotion value in actual application scenarios.

Acknowledgments
This work is supported by State Grid Corporation of China (SGCC) Science and Technology Research Project (Research on Basic Technology of Video Behavior Analysis of Power Grid Operation Based on Artificial Intelligence, 5500-202058318A-0-0-00).

References
[1] He K, Zhang X, Ren S, Sun J 2016 Deep residual learning for image recognition IEEE Conference on Computer Vision and Pattern Recognition pp 770-778
[2] Xie S, Girshick R, Dollár P, et al. 2017 Aggregated residual transformations for deep neural networks IEEE Conference on Computer Vision and Pattern Recognition pp 1492-1500
[3] Shen Z, Liu Z, Li J, et al. 2019 Object detection from scratch with deep supervision[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence 42(2) pp 398-412
[4] Zhu R, Zhang S, Wang X, et al. 2019 Scratchdet: Training single-shot object detectors from scratch IEEE Conference on Computer Vision and Pattern Recognition pp 2268-2277
[5] Sermanet P, Eigen D, Zhang X, et al. 2013 Overfeat: Integrated Recognition, Localization and Detection Using Convolutional Networks arXiv preprint arXiv:1312.6229
[6] Redmon J, Divvala S, Girshick R, et al. 2016 You only look once: Unified, real-time object detection IEEE Conference on Computer Vision and Pattern Recognition pp 779-788
[7] Liu W, Anguelov D, Erhan D, et al. 2016 Ssd: Single shot multibox detector European Conference on Computer Vision Springer pp 21-37
[8] Erhan D, Szegedy C, Toshev A, et al 2014 Scalable object detection using deep neural networks IEEE Conference on Computer Vision and Pattern Recognition pp 2147-2154
[9] Zhang S, Wen L, Bian X, et al. 2018 Single-shot refinement neural network for object detection IEEE Conference on Computer Vision and Pattern Recognition pp 4203-4212
[10] Ronneberger O, Fischer P, Brox T. 2015 U-net: Convolutional networks for biomedical image segmentation International Conference on Medical Image Computing and Computer-assisted Intervention pp 234-241
[11] Wang X, Xiao T, Jiang Y, et al. 2018 Repulsion Loss: Detecting Pedestrians in a Crowd IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 7774-7783
[12] Zhang S, Benenson R, Schiele B. 2017 Citypersons: a Diverse Dataset for Pedestrian Detection arXiv preprint arXiv: 1702.05693
[13] Lin T, Maire M, Belongie S, et al 2014 Microsoft coco: Common objects in context European Conference on Computer Vision Springer pp 740-755
[14] Shao S, Zhao Z, Li B, et al. 2018 Crowdhuman: A Benchmark for Detecting Human in a Crowd arXiv preprint arXiv:1805.00123
[15] Chu X 2020 Detection in Crowded Scenes: One Proposal, Multiple Predictions IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 12211-12220
[16] Lin T Y, Doll’ar P, Girshick R B, et al. 2017 Feature pyramid networks for object detection
IEEE Conference on Computer Vision and Pattern Recognition pp 936-944
[17] Danelljan M. 2016 Beyond correlation filters: Learning continuous convolution operators for visual tracking European Conference on Computer Vision Berlin Springer pp 472-488
[18] Yang F, Zhang L, Xiang T, et al. 2018 Learning to compare: Relation network for fewshot learning IEEE Conference on Computer Vision and Pattern Recognition pp 1199-1208