Learning Hierarchical Graph for Occluded Pedestrian Detection

Gang Li[*‡]
Nanjing University of Science and Technology
gang.li@njust.edu.cn

Jian Li[*]
Youtu Lab, Tencent
swordli@tencent.com

Shanshan Zhang[†‡]
Nanjing University of Science and Technology
shanshan.zhang@njust.edu.cn

Jian Yang[‡]
Nanjing University of Science and Technology
csjyang@njust.edu.cn

ABSTRACT

Although pedestrian detection has made significant progress with the help of deep convolution neural networks, it is still a challenging problem to detect occluded pedestrians since the occluded ones can not provide sufficient information for classification and regression. In this paper, we propose a novel Hierarchical Graph Pedestrian Detector (HGPD), which integrates semantic and spatial relation information to construct two graphs named intra-proposal graph and inter-proposal graph, without relying on extra cues w.r.t visible regions. In order to capture the occlusion patterns and enhance features from visible regions, the intra-proposal graph considers body parts as nodes and assigns corresponding edge weights based on semantic relations between body parts. On the other hand, the inter-proposal graph adopts spatial relations between neighbouring proposals to provide additional proposal-wise context information for each proposal, which alleviates the lack of information caused by occlusion. We conduct extensive experiments on standard benchmarks of CityPersons and Caltech to demonstrate the effectiveness of our method. On CityPersons, our approach outperforms the baseline method by a large margin of 5.24*pp* on the heavy occlusion set, and surpasses all previous methods; on Caltech, we establish a new state of the art of 3.78% MR. Code is available at https://github.com/ligang-cs/PedestrianDetection-HGPD.

CCS CONCEPTS

• **Computing methodologies** → *Object detection.*

KEYWORDS

computer vision, object detection, occluded pedestrian detection, graph neural networks

ACM Reference Format:
Gang Li, Jian Li, Shanshan Zhang, and Jian Yang. 2020. Learning Hierarchical Graph for Occluded Pedestrian Detection. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413983

1 INTRODUCTION

Pedestrian detection is a popular topic in computer vision. On the one hand, it has extensive applications, such as autonomous driving, video surveillance, and robotics. On the other hand, it also serves as a fundamental step for some other vision-based tasks, e.g. person re-identification [26, 39], pose estimation [10, 19], etc. Recently, significant improvements have been achieved with the development of deep convolution neural networks. Although some state-of-the-art detectors can provide reasonable detection results for non-occluded or partially occluded pedestrians, the performance for heavily occluded pedestrians is still far from satisfactory.

Occlusion occurs frequently in real-world applications, and thus it is an important problem to solve. Though quite some efforts have been made for handling occlusion, it is still far from being solved. By analyzing the occlusion issue, we assume the difficulty mainly comes from the following two reasons: (1) lack of human body information from invisible parts; (2) background noise inside the detection window of occluded pedestrians. We show one example in Figure 1.

To improve the model's discrimination ability for occluded pedestrians, an intuitive way is to enhance human body features from visible regions and suppress noisy features from occluded regions. To this end, we need to predict visible parts and then perform feature aggregation or re-weighting based on the prediction. For example, OR-CNN [37] divides each proposal into five pre-defined parts and aggregates features from these parts with predicted visibility scores; MGAN [21] produces a pixel-wise attention map based on visible regions, and then gives higher attention weights to features from visible regions; Zhang *et al.* [38] propose to use the predicted visible box or part detection results as guidance to conduct channel-wise feature selection. All of the above works rely on either visible box annotations as additional supervision at training time or an external body part detector. However, visible boxes are not provided on some large-scale pedestrian datasets, such as EuroCity Persons [1] and NightOwls [18], as they require extensive human labor; running an additional body part detector is computationally expensive at inference.

[*] Both authors contributed equally to this research.
[†] The corresponding author is Shanshan Zhang
[‡] Gang Li, Shanshan Zhang and Jian Yang are with PCA Lab, Key Lab of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education, and Jiangsu Key Lab of Image and Video Understanding for Social Security, School of Computer Science and Engineering, Nanjing University of Science and Technology

MM '20, October 12–16, 2020, Seattle, WA, USA

ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413983

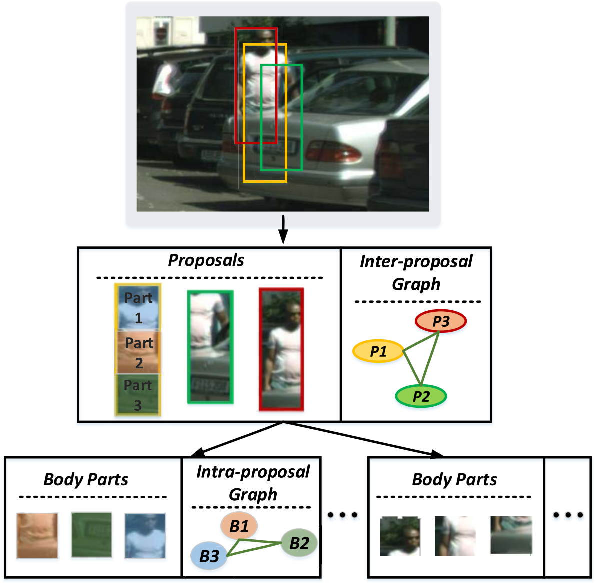

Figure 1: Illustration of our proposed hierarchical graph.

To overcome the above limitations, in this work we aim to model occlusion patterns and enhance features for occluded pedestrians without relying on extra annotations or cues. In order to diminish the effects of noisy features inside occluded pedestrian windows, we build a graph for each proposal to model occlusion patterns, where each body part acts as one node and edges linked to the node are defined by the affinity between the node and the full body. The basic assumption is that the full body has a stronger affinity to visible parts than to occluded parts. In this way, the occlusion pattern is modeled via an intra-proposal graph. Given this occlusion-aware graph, those features from the visible parts are expected to contribute more than those from the invisible parts during feature aggregation. On the other hand, to alleviate weak visual cues of occluded pedestrians, we create the inter-proposal graph, which views each proposal as one node and defines each edge weight based on the spatial relations between two proposals. We consider, for one occluded pedestrian proposal, its neighboring proposals are complementary, as they might cover some visible parts of the target person. Given this spatial-aware graph, those neighboring proposals, which have larger overlaps with the target proposal, would contribute more to the final aggregated features.

Our proposed intra- and inter-proposal graphs can be added on top of any proposal-based detector with small computation overhead.

Our contributions are summarized as follows:

- We propose an intra-proposal graph to model occlusion patterns. Through the message passing in the graph, meaningful information is highlighted, meanwhile, noisy features can be suppressed.
- We build an inter-proposal graph to provide complementary visual cues for occluded pedestrian proposals. It takes spatial relationships between proposals into consideration and alleviates the lack of information caused by occlusion.
- To validate the effectiveness of our method, we conduct extensive experiments on CityPersons and Caltech datasets, and build the new state of the art on both of them.

2 RELATED WORK

In this work, we address the problem of occluded pedestrian detection using graph neural networks. Therefore in this section, we discuss the three lines of works related to this paper: pedestrian detection with CNN, occluded pedestrian detection, and graph neural networks.

2.1 Pedestrian Detection with CNN

In recent years, pedestrian detection is dominated by CNN-based methods [34, 35]. These CNN-based pedestrian detectors can be divided into two categories: one-stage and two-stage detectors. One-stage detectors [11, 15, 16, 20, 23] aim to achieve a trade-off between speed and accuracy. Among them, ALFNet [15] and GDFL [11] follow the framework of SSD [14] and directly predict the object category and anchor box offsets, based on multi-level features. ALFNet [15] uses multi-step refinement and a strict classification criterion to provide accurate detection. And GDFL [11] proposes a scale-aware pedestrian attention module to enhance features. Though one-stage detectors have faster inference speed, they rely on data augmentation and take more training time to converge.

In contrast, more works [2, 6–8, 31, 37, 38] are based on two-stage detectors. Faster R-CNN [24] is a typical two-stage detector, which achieves state-of-the-art performance for object detection. It first generates proposals by region proposal network (RPN), and then crops features of each proposal to conduct classification and regression again. Many recent pedestrian detectors are built on top of it. For example, adapted Faster R-CNN [36] improves Faster R-CNN to better handle the canonical problem of pedestrian detection; AR-Ped [2] introduces an encoder-decoder module for the RPN stage and improves precision progressively. In this paper, we also use Faster R-CNN as the baseline and improve it via employing graph neural networks.

2.2 Occluded Pedestrian Detection

Occlusion is a challenging problem in pedestrian detection, and it has been widely studied in the last few years. Zhang *et al.* [37] and Wang *et al.* [31] handle the occlusion by designing the novel loss formulation. Specifically, aggregation loss is proposed in OR-CNN [37], enforcing proposals to locate compactly around the corresponding person; RepLoss [31] designs a repulsion loss to prevent proposals from shifting to surrounding persons. Since occlusion happens less frequently at the head region, some works learn the relation between the head and the full body. JointDet [7] designs a head-body relationship discriminating module to recall some suppressed pedestrians. PedHunter [6] inserts a parallel branch in R-CNN stage to conduct head segmentation. Besides, some other methods [21, 38] exploit attention mechanisms to enhance the features from visible regions. Zhang *et al.* [38] introduce the channel-wise attention to increase the weights of features from visible body parts. And Xie *et al.* [21] propose the spatial-wise attention to suppress noisy features from the background. In contrast, in this work we propose to build intra- and inter-proposal graphs to enhance features. A

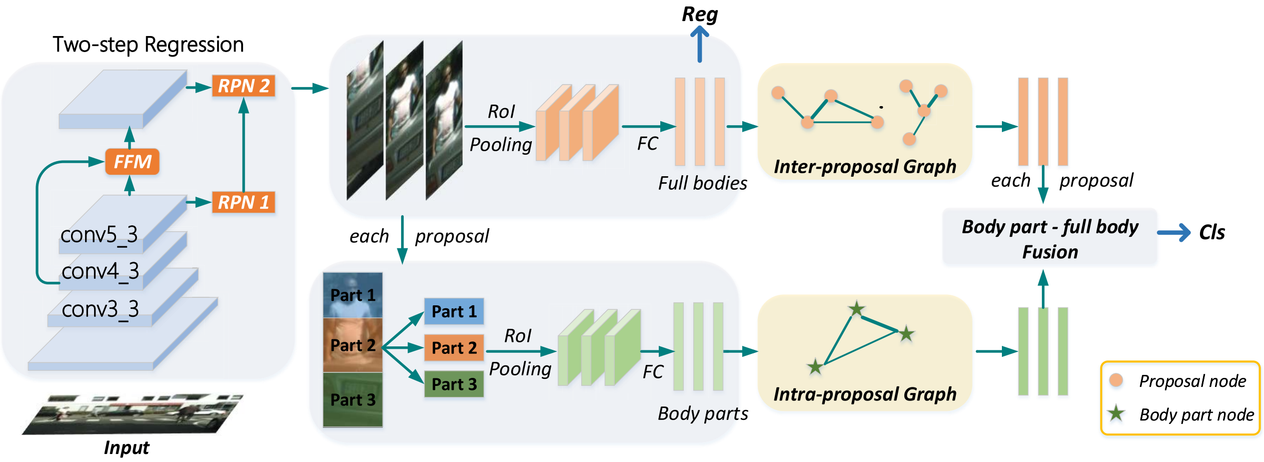

Figure 2: Structure of hierarchical Graph Pedestrian Detector. In the RPN stage, two-step regression is used to provide high-quality proposals, and Feature Fusion Module (FFM) aggregates features from conv4_3 and conv5_3 as input features for second-stage regression (RPN2). In the R-CNN stage, we feed features from holistic proposals into Inter-proposal Graph, meanwhile, we divide each proposal into several body parts and send features from local parts into Intra-proposal Graph. Finally, we fuse output features from intra- and inter-proposal graphs to perform classification. And holistic features before the graph are selected to perform the regression task.

similar attempt is found in [32], where graph is also used to address the occlusion issue. However, our method is substantially different. [32] only creates a graph within the proposal and aims to learn spatial co-occurrence among body parts. While our method considers not only semantic relations among body parts but also spatial relations among proposals. Moreover, we use graph neural networks to enhance the features by focusing more on visible regions and aggregating complementary features from neighboring proposals.

2.3 Graph Neural Networks (GNN)

GNN was proposed in [25], then it is extended to CNN. Bruna *et al.* [4] propose two constructions of graph convolutional networks: the one is based on spatial convolution, the other is based on the Laplician spectrum transformation. As GNN can effectively capture the relation of data and make information propagation between nodes explainable, it has been widely applied to various tasks. In person re-id task, Shen *et al.* [26] construct a graph to represent the pairwise relationships between probe-gallery pairs, and it not only focuses on the target probe-gallery pair but also takes others pairs into consideration. In person search task, Yan *et al.* [33] introduce the context graph, which takes instance pairs as nodes and places the target pair at the center of the graph, so context information can be passed to the target pair through the graph linkage. In the action recognition task, Shi *et al.* [27] propose to learn the topology of the graph in a data-driven method, and introduce hierarchical GCNs to model the first-order and second-order simultaneously.

3 METHODS

In this section, we first describe the overall framework of our proposed Hierarchical Graph Pedestrian Detector (HGPD) in Sec. 3.1, then detail the sub-module in Sec. 3.2, 3.3, and 3.4.

3.1 Overview

In this work, we adopt Faster R-CNN [24] as our baseline, and as in [12, 21], we choose VGG-16 [28] as the backbone. The overall network structure of HGPD is shown in Figure 2. Based on vanilla Faster R-CNN, we propose the hierarchical graph (intra- and inter-proposal graph) and two-step regression. We place the hierarchical graph at the R-CNN stage, right after the RoI pooling layer, and our framework is two-stream. For the inter-proposal stream, we take features of proposals as input, and enhance each proposal by aggregating features from its neighboring ones. For the intra-proposal stream, we feed features from each body part into the graph, then the graph outputs enhanced local features. Finally, we combine features enhanced by each graph for classification. Considering aggregating neighboring features might damage position information of original features, we use features before the graph to perform bounding box regression. And two-step regression is used in the RPN stage to provide high-quality [1] proposals for the R-CNN.

3.2 Intra-proposal Graph

The part-based method for pedestrian detection has been explored in OR-CNN [37], which divides the full body into five body parts based on prior structure information of the human body. However, we consider pedestrians with small scale often occur in real-world scenes. If dividing the small scale pedestrian into five parts, each part only covers limited pixels and can not provide accurate features. So the number of human parts (noted as N) is set to be three in this work. Then we conduct the RoI pooling operation on the holistic body and every body part. Followed by fully connected layers, pooled features maps ($512 \times 7 \times 7$) are converted into a feature vector (f), whose dimension is 1024. To model occlusion

[1] Specifically, high-quality represents proposals with accurate localization.

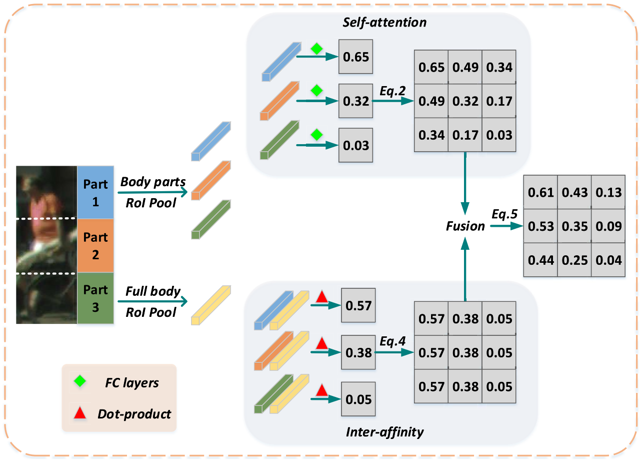

Figure 3: Example of the intra-proposal graph. The region proposal is from the CityPersons validation set. The graph takes features from the full body and body parts as input. There are two parallel modules: Self-attention and Inter-affinity. Self-attention predicts the edge weight in Eq. 2; Inter-affinity calculates the edge weight in Eq. 4. Finally, outputs of two methods are fused to generate the adjacent matrix A_{intra}, as in Eq. 5.

patterns, we propose the intra-proposal graph, where the adjacent matrix can reveal occlusion level of each body part. Formally, the intra-proposal graph is noted as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ represents vertices corresponding to N body parts, and $\mathcal{E}$ refers to a set of edges . The adjacent matrix is noted as A_{intra}. We calculate the output of GNN as:

$$\widetilde{X}_b = \hat{A}_{intra} X_b W, \tag{1}$$

where $X_b \in R^{N \times d}$ refers to input feature vectors of N body parts, $\hat{A}_{intra} \in R^{N \times N}$ is the normalized adjacent matrix and W refers to learning parameters. The adjacent matrix is normalized by: $\hat{A}_{intra} = D^{-1} A_{intra}$, where D denotes the degree matrix. Then we describe how to create the adjacent matrix based on semantic relations. First, we predict the visibility score for each human part in a self-attention manner. Specifically, a series of two fully connected layers followed by a sigmoid function, is used to generate the visibility score (p_i) for the i-th body part. The higher score indicates the larger visibility ratio. The edge weight ($\Omega_{i,j}$) between the i-th and j-th part is calculated by:

$$\Omega_{i,j} = \frac{p_i + p_j}{2}. \tag{2}$$

In this way, the edge between two visible parts is assigned with a high weight, meanwhile, edges linked to the occluded part are assigned lower weights. Without supervision on visible regions, the accuracy of visibility score prediction is not guaranteed, so we introduce the additional guidance as complementary. The basic assumption is that features from full body have a stronger affinity with that from visible parts, than that from occluded parts. Based on this assumption, we use the affinity between local-part features and holistic features to represent the occlusion level of the part. Specifically, we use "Scaled Dot-Product Attention" [30] to measure affinity:

$$q_i = dot(W_K \mathrm{f}_h, W_Q \mathrm{f}_i), \tag{3}$$

where W_K and W_Q project original features into a subspace to measure how well they match. The feature dimension after projection is d_k, which is set to 64. The resulting value (q_i) reflects the affinity between the i-th part (f_i) and the holistic body (f_h). Using inter-affinity guidance, we define the edge weight $\phi_{i,j}$ as:

$$\phi_{i,j} = q_j. \tag{4}$$

Combining the above two methods: self-attention and inter-affinity, the occlusion patterns can be modeled more accurately. The combined adjacent matrix $A_{intra}(i, j)$ is defined as:

$$A_{intra}(i, j) = \sqrt{\Omega_{i,j} * \phi_{i,j}}, \tag{5}$$

and an example is shown in Figure 3.

We also analyze the advantage of using the graph to model body parts rather than directly concatenating features of them. We assume the biggest advantage is that GNN can interweave all parts and provide rich supervision for feature learning. As elements of the graph are connected with each other, we describe the procedure of information propagation for concatenation in Eq. 6 and for GNN in Eq. 7.

$$\mathrm{f}_k^{'} = p_k * \mathrm{f}_k, \tag{6}$$

$$\mathrm{f}_k^{'} = A(k,1)*\mathrm{f}_1 + A(k,2)*\mathrm{f}_2 + \cdots + A(k,k)*\mathrm{f}_k + \cdots + A(k,N)*\mathrm{f}_N, \tag{7}$$

where f_k represents the feature vector of the k-th body part, p_k is the predicted visibility score, $A(k, 1)$ refers to element in the k-th row and $1st$ column of the adjacent matrix, and $\mathrm{f}_k^{'}$ represents enhanced features. We calculate the gradients of error $loss$, w.r.t concatenation and GNN, respectively. For concatenation, it can be written as follows:

$$\frac{\partial loss}{\partial \mathrm{f}_k^{'}} = [0, 0, \cdots, \frac{1}{p_k} * \frac{\partial loss}{\partial \mathrm{f}_k}, \cdots, 0], \tag{8}$$

while for graph neural networks, it is:

$$\begin{aligned} \frac{\partial loss}{\partial \mathrm{f}_k^{'}} = [&\frac{1}{A(k,1)} * \frac{\partial loss}{\partial \mathrm{f}_k}, \frac{1}{A(k,2)} * \frac{\partial loss}{\partial \mathrm{f}_k}, \cdots, \\ &\frac{1}{A(k,k)} * \frac{\partial loss}{\partial \mathrm{f}_k}, \cdots, \frac{1}{A(k,N)} * \frac{\partial loss}{\partial \mathrm{f}_k}]. \end{aligned} \tag{9}$$

As Eq. 8 and 9 demonstrate, with GNN, gradients of different parts are highly correlated, so that features are updated in a more coherent way, rather than separately. The weight assigned to each part not only depends on local part features, but also takes semantic relations between body parts into consideration. Without the help of visibility annotations, semantic relation information provides a potential cue for modeling occlusion patterns. We also compare performance with and without GNN in ablation study (Table 2).

3.3 Inter-proposal Graph

Because occluded pedestrians only provide weak visual cues, insufficient information makes them more difficult to be detected. Specifically, for some detection proposals of occluded pedestrians, though they are highly-overlapped with the ground truth, they derive low classification scores, as the majority of the overlapped regions may be occluded regions and provide meaningless information. However, we observe that some neighboring proposals

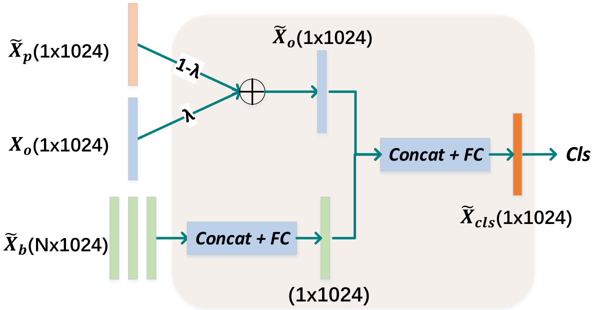
$\widetilde{X}_p$(1x1024)
1-λ
λ
$\widetilde{X}_o$(1x1024)
X_o(1x1024)
$\widetilde{X}_b$(Nx1024)
Concat + FC
Concat + FC
Cls
$\widetilde{X}_{cls}$(1x1024)
(1x1024)

Figure 4: Illustration of aggregating output features from the intra- and inter-proposal graphs. The resulting features ($\widetilde{X}_{cls}$) are used to perform classification.

may cover other visible regions of the ground truth, as shown in Figure 1. Given this, we construct the inter-proposal graph to create connections between neighbouring proposals. In the graph, we take region proposals as vertices, and define the edge weight based on spatial relationships. It is intuitive that spatially closer proposals contain more related features and should be assigned higher edge weights. We select Intersection over Union (IoU) as the metric to measure spatial relations. The propagation of the inter-proposal graph is:

$$\widetilde{X_p} = \hat{A}_{inter} X_p W, \tag{10}$$

where X_p refers to input features from proposals, W refers to learning parameters and $\hat{A}_{inter}$ represents the normalized adjacent matrix. And the derivation of the adjacent matrix is as follows:

$$A_{inter}(i,j) = \begin{cases} IoU(m_i, m_j) & if\ i \neq j \\ 0 & otherwise, \end{cases} \tag{11}$$

where m_i, m_j refers to the i-th,j-th proposal, and $IoU(m_i, m_j) = \frac{m_i \cap m_j}{m_i \cup m_j}$. We set $A_{inter}(k,k)$ to 0 to avoid self-enhancing. Finally, we combine original features X_o with enhanced neighboring features $\widetilde{X_p}$ as:

$$\widetilde{X_o} = \lambda X_o + (1-\lambda)\widetilde{X_p}, \tag{12}$$

where λ represents the weighting parameter that balances neighbouring features and original features, and λ is set to 0.9 empirically. Finally, we fuse output features from intra- and inter-proposal graph to perform classification, and the procedure of the feature fusion is shown in Figure 4. Original proposal features (X_o) are enhanced by their neighboring features ($\widetilde{X_p}$), and resulting features are noted as $\widetilde{X_o}$. Then $\widetilde{X_o}$ is combined with enhanced local features $\widetilde{X_b}$.

3.4 Two-step Regression

In this work, we adopt Faster R-CNN [24] as our baseline. Because of the huge variance of pedestrian scale, we observe that some ground truth boxes are assigned low-quality proposals, some of which only cover the partial person body and contain too many background clutters. Inaccurate localization would make our part-based graph meaningless, so we introduce two-step regression to provide high-quality proposals for the hierarchical graphs. Moreover, high-quality proposals can also reduce false detections of adjacent overlapping pedestrians.

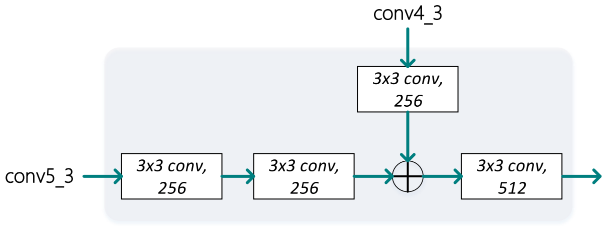

Figure 5: Structure of feature fusion module.

We perform two-step regression in the RPN stage. As in [36], we remove the fourth max-pooling layer from VGG-16, so the stride of conv5 is reduced to 8, which provides higher resolution feature maps for RPN and helps to detect small objects. The first regression is performed on the conv5 layer and outputs regressed anchors for the following step. With these well-initialized anchors as inputs, the second regression can generate more high-quality proposals. To make use of precise position information in shallow layers [13], we combine features from conv4 and conv5 via the feature fusion module (FFM), and the structure of FFM is shown in Figure 5. Finally, we perform the classification and second regression on fused features to derive more accurate localization and object scores.

The whole network can be trained end-to-end by minimizing the following loss function:

$$L = L^{rpn}_{loc1} + L^{rpn}_{loc2} + \alpha L^{rpn}_{cls} + L^{rcnn}_{loc} + L^{rcnn}_{cls}, \tag{13}$$

where $L^{rpn}_{loc1}, L^{rpn}_{loc2}$ are the first and second regression losses in the RPN stage. And other loss terms are the same with vanilla Faster R-CNN [24]. We set $\alpha = 2$ to balance the loss terms.

4 EXPERIMENTS

In this section, we will give a brief description of the datasets used for experiments, followed by an introduction to the evaluation metrics. Then some implementation details are presented. After that, extensive ablation studies and comparisons with the state-of-the-art methods are presented.

4.1 Datasets

CityPersons. The CityPersons dataset [36] is built upon Cityscapes and provides accurate pedestrian annotations. It has 2975, 500, and 1525 images for train, validation, and test subsets, respectively. The images of CityPersons are collected from 27 cities, so it covers diverse environments. And the ratio of fully visible pedestrians in CityPersons is less than 30%, which makes CityPersons a challenging dataset and suitable for researching on occlusion cases.

Caltech. Caltech [9] is another widely used pedestrian dataset, which is approximately 10 hours of 30 Hz video, taken from a vehicle driving in an urban environment. It consists of 11 sets of videos, the first 6 sets of which are training sets and the last 5 sets are testing sets. Zhang *et al.* [34] provide refined annotations for both training and test sets, which correct several errors in original annotations and improve localization accuracy of bounding boxes. In our work, we conduct all experiments related to Caltech on the new annotations.

Table 1: Ablation study on CityPersons validation set. Numbers are log-average miss rates (lower number indicates better performance).

Baseline	Two-step Regression	Intra-proposal Graph	Inter-proposal Graph	R	HO
✓				13.46	56.98
	✓			12.50	55.58
		✓		12.33	53.19
			✓	12.58	53.59
	✓	✓		11.82	52.87
	✓	✓	✓	**11.27**	**51.74**
Overall Improvement				**+2.19**	**+5.24**

4.2 Evaluation metric

Following [9], the log-average miss rate (noted as MR) is used in our work. It is calculated by averaging miss rates over 9 points uniformly sampled from $[10^{-2}, 10^{0}]$ false positive per image (FPPI). On the CityPersons validation set and Caltech test set, we report results across two different occlusion subsets: Reasonable (**R**) and Heavy Occlusion (**HO**). The visibility ratio in **R** is larger than 65%, and the visibility ratio in **HO** ranges from 20% to 65%. In **R** and **HO**, the height of pedestrians is at least 50 pixels. Better results on **HO** are considered as stronger evidence of better occlusion handling. On the CityPersons test set, besides **R** and **HO**, we also report results on the **All** subset. **All** contains pedestrians, whose visibility ratio is larger than 20% and height is at least 20 pixels. Besides, to evaluate the quality of region proposals, we introduce the average recall (noted as AR), which is calculated by averaging the recalls across IoU thresholds from 0.5 to 0.95 with a step of 0.05.

4.3 Implementation

We implement our method with Pytorch [22] and mmdetection [5]. No data augmentation is used except standard horizontal image flipping. SGD is selected as the back-propagation algorithm.

CityPersons. The model is trained on one GTX 2080Ti GPU with a batch size of 2, for 14 epochs. The learning rate is set to 0.02 and reduced to 0.002 after 10 epochs.

Caltech. As in [15, 16, 31, 37], we start with the model pretrained on CityPersons, then finetune the model on the Caltech dataset for another 6 epochs. For the first 4 epochs, the learning rate is set to 0.02 and then reduced to 0.002 for the last two epochs.

4.4 Ablation Study

To better understand our model, we conduct ablation experiments on the CityPersons validation set. The **R** set is used as the training set for these experiments.

Component-wise Analysis. To demonstrate the effectiveness of our hierarchical graph pedestrian detector, a comprehensive component-wise analysis is performed in which different components are added on top of a strong baseline method step by step. The results are reported in Table 1. As Table 1 shows, both of intra- and inter-proposal graphs dramatically reduce the error on the **HO** set. Specifically, intra-proposal graph brings an absolute gain of 3.79*pp* (*pp* represents percentage points), and inter-proposal graph also outperforms the baseline with 3.39*pp*, demonstrating the proposed hierarchical graph is useful for addressing occlusion. With the help of two-step regression, more accurate region proposals are sent into the hierarchical graph. Finally, HGPD achieves log-average miss rates of 11.27% on the **R** set and 51.74% on the **HO** set, achieving a total improvement to the baseline of 2.19% and 5.24% respectively. These results demonstrate the proposed method effectively handles different levels of occlusion.

Table 2: The effects of intra-proposal graph.

w/o graph	intra-proposal graph		R	HO
	self-attention	inter-affinity		
			13.46	56.98
✓			13.98	55.15
	✓		12.36	54.47
	✓	✓	**12.33**	**53.19**

Table 3: Ablation study on the number of human parts. 2 or 3 parts mean uniformly cropping the full body into 2 or 3 parts vertically. 5 parts mean cropping the full body into 3 parts vertically and 2 parts horizontally.

# number	R	HO	R+HO
2	**11.96**	53.59	**30.92**
3	12.33	**53.19**	31.05
5	12.49	55.67	31.65

The effects of intra-proposal graph. To model accurate occlusion patterns, we propose two options to define the edge weights, namely self-attention and inter-affinity. From Table 2, when only self-attention is used as edge weights, intra-proposal graph brings gains of 1.10*pp* on the **R** set and 2.51*pp* on the **HO** set; when we involve inter-affinity as an additional term for edge weights, we achieve larger improvements, i.e. 1.13*pp* and 3.79*pp* on the **R** and **HO** sets respectively. These results validate that inter-affinity guidance can help to better model occlusion patterns. We also conduct experiments to verify the effect of using the intra-proposal graph. For comparison, we introduce another reference method without graph, for which features from body parts are multiplied with corresponding visibility scores, and simply concatenated for classification. The results in Table 2 indicate that the concatenation operation without graph brings negligible improvements on the **HO** set and it even drops by 0.52*pp* on the **R** set. Compared to the simple concatenation operation, our intra-proposal graph integrates the features from different body parts in a more effective way.

Number of body parts. Table 3 shows the performance on different number of body parts. When we divide the full body into 2 parts (top and bottom half), it achieves the best performance on the **R** set, as the occlusion ratio on the **R** set is lower than 35%, so 2 parts can handle all occlusion patterns on the **R** set. But the

Table 4: The effects of two-step regression.

RPN	R-CNN	AR_{100}	AR_{300}	AR_{1000}
		64.8	69.7	71.2
✓		69.6	72.8	**73.7**
	✓	**72.2**	**73.0**	73.2

Table 5: The effects of decoupling two tasks.

sibling	separate	**R**	**HO**
✓		12.44	53.36
	✓	**11.27**	**51.74**

best performance on the **HO** set is obtained under the number of 3, where more diverse occlusion patterns can be modeled. The result also indicates more body parts, e.g. 5, can not bring inconsistent improvements. We divide the full body into 3 parts in the following experiments.

The effects of two-step regression. Table 4 shows the quality of proposals when we place two-step regression at different locations of Faster R-CNN. AR_{100}, AR_{300} and AR_{1000} refer to average recalls for top 100, 300 and 1000 proposals in each image. Two-step regression can be placed at either the RPN or R-CNN stage, and we conduct experiments to compare these two choices. As Table 4 indicates, no matter where to place, two-step regression can bring significant improvements on AR. In practice, we usually use a large number of region proposals for Faster R-CNN (i.e. 1000), so AR_{1000} is a better indicator. Performing two-step regression in the RPN stage achieves the highest AR_{1000} of 73.7%, and outperforms the baseline and placing it at R-CNN by 2.5*pp*, 0.5*pp* respectively.

Effects of decoupling classification and regression tasks. We assume that combining local features and neighboring features would be harmful to the localization task, since the localization is sensitive to the boundary features [29]. So we propose to select features before the hierarchical graph to perform bounding box regression, which is noted as the separate head. And performing both classification and regression on the output features from hierarchical graphs is noted as sibling head. The comparison in Table 5 demonstrates decoupling two tasks works better, and it outperforms the counterpart by 1.17*pp*/1.61*pp* on the **R**/**HO** set.

4.5 Comparison on CityPersons

We compare our method on the CityPersons validation with state-of-the-art methods in Table 6. It is noted that existing pedestrian detectors employ different subsets of training samples, which differ in occlusion level, and input scales, which highly affect the performance. Considering fairness, we make comparisons at different settings in terms of training subset and input scale. Based on whether visible box annotations (VBB) are used at training time, pedestrian detectors are divided into two groups, namely VBB-free and VBB-based methods.

First, we compare our method with VBB-free methods, including ATT+part [38], RepLoss [31], adaptive-NMS [12], CSP [16], and ALFNet [15]. From Table 6(a), we have the following observations:

Table 6: Comparisons of different methods on the CityPersons validation set. Numbers are log-average miss rates (lower is better). The scale column indicates the upsampling factor of input images. Bold indicates the best results. We separate VBB-free and VBB-based methods in two subtables.

(a) Comparison of VBB-free methods

Methods	Visibility	Scale	**R**	**HO**
ATT+part [38][CVPR18]	≥ 65%	1x	16.0	56.7
RepLoss [31][CVPR18]	≥ 65%	1x	13.2	56.9
AdapNMS [12][CVPR19]	≥ 65%	1x	11.9	55.2
HGPD(Ours)	≥ 65%	1x	**11.3**	**51.7**
ALFNet [15] [ECCV18]	≥ 0%	1x	12.0	52.0
CSP [16] [CVPR19]	≥ 0%	1x	**11.0**	49.3
HGPD(Ours)	≥ 0%	1x	11.5	**41.3**
RepLoss [31] [CVPR18]	≥ 65%	1.3x	11.5	55.3
AdapNMS [12] [CVPR19]	≥ 65%	1.3x	10.8	54.0
HGPD(Ours)	≥ 65%	1.3x	**10.7**	**50.9**

(b) Comparison of VBB-based methods

Methods	Visibility	Scale	**R**	**HO**
MGAN [21] [ICCV19]	≥ 65%	1x	11.5	51.7
HGPD(Ours)	≥ 65%	1x	**11.3**	**51.7**
MGAN [21] [ICCV19]	≥ 0%	1x	**11.3**	42.0
HGPD(Ours)	≥ 0%	1x	11.5	**41.3**
OR-CNN [37] [ECCV18]	≥ 50%	1x	12.8	55.7
MGAN [21] [ICCV19]	≥ 50%	1x	**10.8**	46.7
HGPD(Ours)	≥ 50%	1x	11.5	**45.9**
Bi-box [41] [ECCV18]	≥ 30%	1.3x	11.2	44.2
A+DT [40] [ICCV19]	≥ 30%	1.3x	11.1	44.3
HGPD(Ours)	≥ 30%	1.3x	**10.9**	**40.9**

(1) Our method outperforms top competitors by a large margin (more than 3*pp*) on the **HO** set and achieves comparable performance to the second best one (CSP) on the **R** set. (2) Our HGPD also consistently outperforms those methods, which are designed for occlusion handling, including RepLoss and Adaptive-NMS, on both **R** and **HO**. With a 1x input scale and **R** training set, our method achieves log-average miss rates of 11.3% and 51.7% on the **R** and **HO** sets, surpassing RepLoss and Adaptive-NMS. (3) Besides, compared with one-stage detectors (ALFNet and CSP), our method outperforms both of them on the **HO** set by a large margin (∼ 8*pp*).

Furthermore, we compare our HGPD with VBB-based methods in Table 6(b). Though our method uses less supervision information, it still achieves the best performance on the **HO** set. MGAN [21] is a strong competitor, which uses visible regions to generate spatial-wise attention. We conduct comprehensive comparisons with MGAN, specifically we use three training sets, including visibility ratio ≥ 65%, ≥ 50% and ≥ 0%. With each training set, our

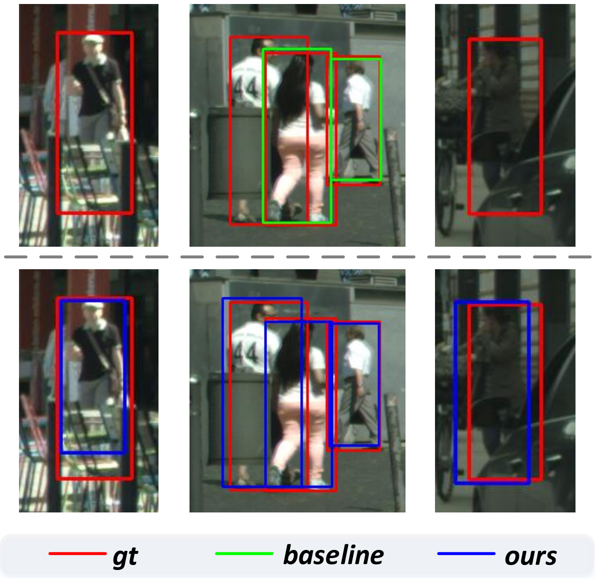

Figure 6: Visualization of detection results from baseline (top row) and our method (bottom row). The results are collected at *FPPI*=0.1. Our method shows fewer false negative detections for occlusion.

HGPD outperforms MGAN on **HO**; while using training data with visibility ≥ 65%, our HGPD surpasses MGAN by 0.23*pp* on the **R** set; when we use a 1.3x input scale and more occluded training data (visibility ≥ 30%), our HGPD achieves the best performance on the **HO** set: a log-average miss rate of 40.9%, outperforming Bi-box [41] and FRCN+A+DT [40].

To summarize, our HGPD builds a new state of the art on the **HO** set (40.9% MR), and also yields competitive results on the **R** set (10.7% MR). Qualitative results on the CityPersons validation set are shown in Figure 6. As the detection examples for occluded pedestrians indicate, our method robustly handles occlusion and yields higher recall. Finally, we send our detection predictions on the Citypersons test set to the evaluation server [36], and the results are shown in Table 7.

Table 7: State-of-the-art comparison on CityPersons test set.

Methods	Reference	**R**	**HO**	**All**
Adaptive-FRCN [36]	CVPR17	12.97	50.47	43.86
RepLoss [31]	CVPR18	11.48	52.59	39.17
OR-CNN [37]	ECCV18	11.32	51.43	40.19
Adaptive-NMS [12]	CVPR19	11.40	46.99	38.89
MGAN [21]	ICCV19	**9.29**	40.97	38.86
HGPD(Ours)	-	10.17	**38.65**	**38.24**

Table 8: Comparison with state-of-the-art methods on the Caltech dataset. All results are evaluated on the new annotations provided by [34].

Methods	Reference	**R**	**HO**
SDS-RCNN [3]	ICCV17	6.44	42.56
HyperLearner [17]	CVPR17	5.5	48.7
ALFNet [15]	ECCV18	4.50	51.0
OR-CNN [37]	ECCV18	4.10	45.0
RepLoss [37]	CVPR18	4.00	41.8
AR-Ped [2]	CVPR19	4.36	48.80
CSP [16]	CVPR19	3.80	36.5
HGPD(Ours)	-	4.83	36.33
HGPD*(Ours)	-	**3.78**	**32.26**

4.6 Comparison on Caltech

Here, we compare our HGPD with state-of-the-art methods on the Caltech dataset. Firstly, we train the HGPD with original image scale (640x480), and only use standard horizontal image flipping as data augmentation. As shown in Table 8, our HGPD achieves the best performance of 32.26% MR on the **HO** set, and it outperforms the second-best method CSP [16] with 0.17*pp*. It is worth mentioning, OR-CNN [37] and RepLoss [31] use the 2x image scale as input, and ALFNet, CSP also utilize random crop as extra data augmentation. To reduce the impact of input size for a fair comparison, we use multi-scale test strategy (noted as HGPD*), and obtain the state-of-the-art miss rates of 3.78% on the **R** set and 32.26% on the **HO** set. AR-Ped [2] is a strong competitor, and our HGPD outperforms it with absolute gains of 0.58*pp* and 16.54*pp* on **R** and **HO** sets. Our method also runs at a speed of 12 FPS on Caltech and achieves a real-time pedestrian detection.

5 CONCLUSION

In this work, we propose a hierarchical graph pedestrian detector to handle the occlusion issue, which contains intra- and inter-proposal graphs. We employ semantic relationships to construct intra-proposal graph, and it can effectively model occlusion patterns and highlight features from visible regions. And the inter-proposal graph can provide meaningful visual cues for proposals of occluded pedestrians. Without extra visible box annotations, the proposed framework achieves state-of-the-art performance on two widely adopted pedestrian datasets, CityPersons and Caltech.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (Grants No. 61702262, U1713208), Funds for International Cooperation and Exchange of the National Natural Science Foundation of China (Grant No. 61861136011), Natural Science Foundation of Jiangsu Province, China (Grant No. BK20181299), Young Elite Scientists Sponsorship Program by CAST (2018QNRC001), “the Fundamental Research Funds for the Central Universities” (No. 30920032201), and Science and Technology on Parallel and Distributed Processing Laboratory Open Fund (WDZC20195500106).

REFERENCES

[1] Markus Braun, Sebastian Krebs, Fabian Flohr, and Dariu M Gavrila. 2018. The eurocity persons dataset: A novel benchmark for object detection. *arXiv preprint arXiv:1805.07193* (2018).

[2] Garrick Brazil and Xiaoming Liu. 2019. Pedestrian detection with autoregressive network phases. In *CVPR*. 7231–7240.

[3] Garrick Brazil, Xi Yin, and Xiaoming Liu. 2017. Illuminating pedestrians via simultaneous detection & segmentation. In *ICCV*. 4950–4959.

[4] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. 2013. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203* (2013).

[5] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. 2019. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155* (2019).

[6] Cheng Chi, Shifeng Zhang, Junliang Xing, Zhen Lei, Stan Z Li, and Xudong Zou. 2019. PedHunter: Occlusion Robust Pedestrian Detector in Crowded Scenes. *arXiv preprint arXiv:1909.06826* (2019).

[7] Cheng Chi, Shifeng Zhang, Junliang Xing, Zhen Lei, Stan Z Li, and Xudong Zou. 2019. Relational Learning for Joint Head and Human Detection. *arXiv preprint arXiv:1909.10674* (2019).

[8] Xuangeng Chu, Anlin Zheng, Xiangyu Zhang, and Jian Sun. 2020. Detection in Crowded Scenes: One Proposal, Multiple Predictions. *arXiv preprint arXiv:2003.09163* (2020).

[9] Piotr Dollar, Christian Wojek, Bernt Schiele, and Pietro Perona. 2011. Pedestrian detection: An evaluation of the state of the art. *IEEE transactions on pattern analysis and machine intelligence* 34, 4 (2011), 743–761.

[10] Jiefeng Li, Can Wang, Hao Zhu, Yihuan Mao, Hao-Shu Fang, and Cewu Lu. 2019. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 10863–10872.

[11] Chunze Lin, Jiwen Lu, Gang Wang, and Jie Zhou. 2018. Graininess-aware deep feature learning for pedestrian detection. In *ECCV*. 732–747.

[12] Songtao Liu, Di Huang, and Yunhong Wang. 2019. Adaptive NMS: Refining Pedestrian Detection in a Crowd. In *CVPR*. 6459–6468.

[13] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. 2018. Path aggregation network for instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 8759–8768.

[14] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. 2016. Ssd: Single shot multibox detector. In *European conference on computer vision*. Springer, 21–37.

[15] Wei Liu, Shengcai Liao, Weidong Hu, Xuezhi Liang, and Xiao Chen. 2018. Learning efficient single-stage pedestrian detectors by asymptotic localization fitting. In *ECCV*. 618–634.

[16] Wei Liu, Shengcai Liao, Weiqiang Ren, Weidong Hu, and Yinan Yu. 2019. High-level Semantic Feature Detection: A New Perspective for Pedestrian Detection. In *CVPR*. 5187–5196.

[17] Jiayuan Mao, Tete Xiao, Yuning Jiang, and Zhimin Cao. 2017. What can help pedestrian detection?. In *CVPR*. 3127–3136.

[18] Lukáš Neumann, Michelle Karg, Shanshan Zhang, Christian Scharfenberger, Eric Piegert, Sarah Mistr, Olga Prokofyeva, Robert Thiel, Andrea Vedaldi, Andrew Zisserman, et al. 2018. NightOwls: A pedestrians at night dataset. In *ACCV*. Springer, 691–705.

[19] Alejandro Newell, Zhiao Huang, and Jia Deng. 2017. Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in neural information processing systems*. 2277–2287.

[20] Junhyug Noh, Soochan Lee, Beomsu Kim, and Gunhee Kim. 2018. Improving occlusion and hard negative handling for single-stage pedestrian detectors. In *CVPR*. 966–974.

[21] Yanwei Pang, Jin Xie, Muhammad Haris Khan, Rao Muhammad Anwer, Fahad Shahbaz Khan, and Ling Shao. 2019. Mask-guided attention network for occluded pedestrian detection. In *ICCV*. 4967–4975.

[22] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch. (2017).

[23] Jimmy Ren, Xiaohao Chen, Jianbo Liu, Wenxiu Sun, Jiahao Pang, Qiong Yan, Yu-Wing Tai, and Li Xu. 2017. Accurate single stage detector using recurrent rolling convolution. In *CVPR*. 5420–5428.

[24] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*. 91–99.

[25] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. 2008. The graph neural network model. *IEEE Transactions on Neural Networks* 20, 1 (2008), 61–80.

[26] Yantao Shen, Hongsheng Li, Shuai Yi, Dapeng Chen, and Xiaogang Wang. 2018. Person re-identification with deep similarity-guided graph neural network. In *Proceedings of the European conference on computer vision (ECCV)*. 486–504.

[27] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. 2019. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 12026–12035.

[28] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[29] Guanglu Song, Yu Liu, and Xiaogang Wang. 2020. Revisiting the Sibling Head in Object Detector. *arXiv preprint arXiv:2003.07540* (2020).

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[31] Xinlong Wang, Tete Xiao, Yuning Jiang, Shuai Shao, Jian Sun, and Chunhua Shen. 2018. Repulsion loss: Detecting pedestrians in a crowd. In *CVPR*. 7774–7783.

[32] Jin Xie, Yanwei Pang, Hisham Cholakkal, Rao Muhammad Anwer, Fahad Shahbaz Khan, and Ling Shao. 2020. PSC-Net: Learning Part Spatial Co-occurence for Occluded Pedestrian Detection. *arXiv preprint arXiv:2001.09252* (2020).

[33] Yichao Yan, Qiang Zhang, Bingbing Ni, Wendong Zhang, Minghao Xu, and Xiaokang Yang. 2019. Learning context graph for person search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2158–2167.

[34] Shanshan Zhang, Rodrigo Benenson, Mohamed Omran, Jan Hosang, and Bernt Schiele. 2016. How far are we from solving pedestrian detection?. In *CVPR*. 1259–1267.

[35] Shanshan Zhang, Rodrigo Benenson, Mohamed Omran, Jan Hosang, and Bernt Schiele. 2017. Towards reaching human performance in pedestrian detection. *IEEE transactions on pattern analysis and machine intelligence* 40, 4 (2017), 973–986.

[36] Shanshan Zhang, Rodrigo Benenson, and Bernt Schiele. 2017. Citypersons: A diverse dataset for pedestrian detection. In *CVPR*. 3213–3221.

[37] Shifeng Zhang, Longyin Wen, Xiao Bian, Zhen Lei, and Stan Z Li. 2018. Occlusion-aware R-CNN: detecting pedestrians in a crowd. In *ECCV*. 637–653.

[38] Shanshan Zhang, Jian Yang, and Bernt Schiele. 2018. Occluded pedestrian detection through guided attention in CNNs. In *CVPR*. 6995–7003.

[39] Feng Zheng, Cheng Deng, Xing Sun, Xinyang Jiang, Xiaowei Guo, Zongqiao Yu, Feiyue Huang, and Rongrong Ji. 2019. Pyramidal person re-identification via multi-loss dynamic training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 8514–8522.

[40] Chunluan Zhou, Ming Yang, and Junsong Yuan. 2019. Discriminative Feature Transformation for Occluded Pedestrian Detection. In *ICCV*. 9557–9566.

[41] Chunluan Zhou and Junsong Yuan. 2018. Bi-box regression for pedestrian detection and occlusion estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 135–151.