mathematics

MDPI

Article

Cascaded Cross-Layer Fusion Network for Pedestrian Detection

Zhifeng Ding [1], Zichen Gu [2,*], Yanpeng Sun [1] and Xinguang Xiang [1]

[1] School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210023, China; dingzhifeng@njust.edu.cn (Z.D.); yanpeng_sun@njust.edu.cn (Y.S.); xgxiang@njust.edu.cn (X.X.)

[2] INMAI Railway Technology Co., Ltd., Beijing 100015, China

* Correspondence: guzichen2018@rails.cn

Abstract: The detection method based on anchor-free not only reduces the training cost of object detection, but also avoids the imbalance problem caused by an excessive number of anchors. However, these methods only pay attention to the impact of the detection head on the detection performance, thus ignoring the impact of feature fusion on the detection performance. In this article, we take pedestrian detection as an example and propose a one-stage network Cascaded Cross-layer Fusion Network (CCFNet) based on anchor-free. It consists of Cascaded Cross-layer Fusion module (CCF) and novel detection head. Among them, CCF fully considers the distribution of high-level information and low-level information of feature maps under different stages in the network. First, the deep network is used to remove a large amount of noise in the shallow features, and finally, the high-level features are reused to obtain a more complete feature representation. Secondly, for the pedestrian detection task, a novel detection head is designed, which uses the global smooth map (GSMap) to provide global information for the center map to obtain a more accurate center map. Finally, we verified the feasibility of CCFNet on the Caltech and CityPersons datasets.

Keywords: pedestrian detection; machine learning; end-to-end; anchor-free; feature reuse

check for updates

Citation: Ding, Z.; Gu, Z.; Sun, Y.; Xiang, X. Cascaded Cross-Layer Fusion Network for Pedestrian Detection. *Mathematics* **2022**, *10*, 139. https://doi.org/10.3390/math10010139

Academic Editors: Radu Tudor Ionescu, Jianping Gou, Weihua Ou, Shaoning Zeng and Lan Du

Received: 25 November 2021
Accepted: 22 December 2021
Published: 4 January 2022

1. Introduction

Pedestrian detection is a crucial but challenging task in computer vision and multimedia, which has been applied in various fields. The goal of pedestrian detection is to find all pedestrians in images and videos. Early detection methods [1–6] show that directly using the features of the backbone output is not conducive to the detection of small objects in the image. Recent detection methods show that obtaining high-resolution and high-quality feature representations is the key to improving detection results. As we all know, the low-level features of the backbone contain accurate small object information, while the high-level features contain accurate large object information. Therefore, how to more effectively integrate the characteristics of different stages has been the focus of research on pedestrian detection in recent years.

According to the feature detection method, we divide the feature fusion methods into FPN-like (Like Feature Pyramid Networks) methods and FCN-like (Like Fully Convolutional Networks) methods. The specific difference is that the FPN-like methods detects features of different scales separately, while the FCN-like methods only detects final feature after the fusion of features of different scales. The basic idea of the FPN-like methods is proposed by Single Shot MultiBox Detector (SSD) [2], and its main process is to detect objects in feature maps at different resolutions. However, SSD ignores the spatial information in the shallow feature map, and thus loses the information of small objects in the shallow feature. To improve the recognition performance of small objects, Feature Pyramid Networks (FPN) [7] combines high-level feature maps with strong semantic information and low-level feature maps with weak semantic information but rich spatial information. Some recent works have proposed some FPN-like methods [8–14]. In order to more effectively integrate features of different scales. However, these methods mainly focus on the features of adjacent stages in the feature fusion process, and the deep features containing rich semantic information gradually weaken during the top-down process. Therefore, high-level semantic information is lost when detecting shallow features, so that small objects in the image can not be effectively detected.

To avoid the shortcomings of FPN-like methods, some methods directly fuse features of different scales, and then only need to detect the fused features. The origin of this type of method comes from Fully Convolutional Networks (FCN) [15], which combines the features of different stages to obtain feature maps containing semantic information of different scales. In this paper, structures similar to FCN are collectively referred to as the FCN-like methods [15–24]. Compared with FPN-like methods, FCN-like methods have lower computational complexity and faster computational speed, while avoiding the situation that small objects can not be detected due to loss of high-level semantic information. These methods have the same weights for feature fusion at different scales in the feature integration process. In this case, the noise in the shallow features will directly affect the accuracy of the final feature. Previous work Semantic Structure Aware Inference (SSA) [25] proved that the information of small objects is not only in the shallow features, but there is also a small amount of small object information in the deep features. However, the noise information in the shallow network is huge, so how to reduce the impact of the noise information in the shallow features on the detection accuracy is a problem that has not been solved by the current FCN-like methods.

Toward this end, this work takes pedestrian detection as an example and proposes a novel Cascaded Cross-layer Fusion Network (CCFNet), which consists of backbone network, Cascaded Cross-layer Fusion module (CCF), and novel detection head. The basic process framework is shown in Figure 1. First, the CCF merges the features in different stages in the backbone to obtain the final feature map and then performs detection on the feature map. Different from the previous method, CCF uses deep features to denoise shallow features and then reuses deep features to increase the semantic information in the final feature map. To improve the running speed of the algorithm, CCFNet adopts the anchor-free method, based on the detection of pedestrian center points, does not generate anchor points and anchor boxes, and does not match multiple key points. In the detection head, we introduced the center map and global smooth map (GSMap) of the object respectively to reduce the impact of complex scenes and object crowding on the detection performance. Traditional anchor-free detection head only rely on scale map to solve the problem of *'where'* and *'how size'* the object is. This approach increases the difficulty of training the detector. Therefore, we first introduce the center map to undertake the task of *'where the object is'*, while the scale map only needs to undertake the task of *'how size the object is'*. The center map is obtained by convolution, so the center map is obtained by local feature inference. The finiteness of local features limits the accuracy of the center map, so we introduce global smooth map to provide global information for the center map. The specific process is shown in the detection head in Figure 1. Extensive experimental are conducted on the Caltech and CityPersons datasets. The superior performance of CFFNet for pedestrian detection is demonstrated in comparison with the state-of-the-art methods.

The main contributions of this work are summarized as follows:

(1) We propose a novel Cascaded Cross-layer Fusion module (CCF) to reduce the noise information in the shallow features through high-level semantic information, and at the same time reuse high-level semantic information to strengthen the high-level semantic information in the final feature map;

(2) The center map provides the confidence of each object center point, but the confidence is obtained from local information. Therefore, this paper proposes global smooth map to provide the center map with global information, thereby improving the accuracy of the center map;

(3) The feasibility of CCFNet is verified on the Caltech and CityPersons Datasets.

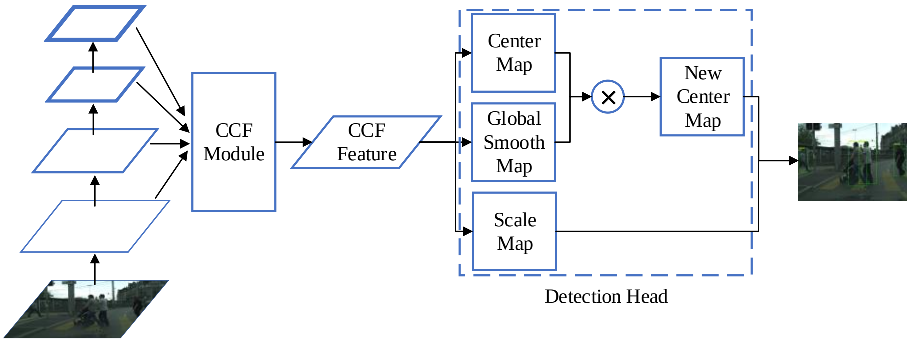


Figure 1. The overall structure of Cascaded Cross-layer Fusion Network (CCFNet). It includes two parts: CCF module and detection head. CCF cascades and reuses features to generate low-level feature maps with contextual semantic information. This feature map generates center map, scale map, and global smooth map through the detection head. And generate the new center map with global information by integrating center map and global smooth map. Finally, locate and mark the objects.

2. Related Work

2.1. Anchor-Base and Anchor-Free

The object detection model can be divided into anchor-based detection network and anchor-free detection network. The anchor-based detection network uses anchor points and anchor boxes to generate high-quality prediction regions, then classifies and regresses the prediction regions, which have high accuracy and can extract richer features. Such as Faster Regions with CNN Features (Faster R-CNN) [1], Cascade Regions with CNN Features (Cascade R-CNN) [26], SSD [2], You Only Look Once Version 2 (YOLOv2) [27], etc. However, anchor-base detection network requires manual intervention due to the number of anchor points and the large aspect ratio of the anchor box, which has disadvantages such as too many parameters and insufficient flexibility.

Therefore, people study methods that do not rely on anchor points and anchor boxes, this method is called the anchor-free detection network. The anchor-free detection network are divided into two types: anchor-free detection network based on key points and anchor-free detection network based on object center. The former generate an object bounding box through a set of predefined or self-learned key points (usually a set of corner points of the bounding box) to locate the object, such as CornerNet-Lite [28] and ExtremeNet [29], etc. The latter locates the object by calculating the distance from the object center to the four sides of the bounding box, such as Center and Scale Prediction (CSP) [23], CenterNet [30], etc. The anchor-free detection network based on object center is similar to the anchor-base detection network, but there is not need to generate a large number of anchor points to predict the bounding box, which improves the detection speed of the algorithm. Recently, Zhang et al. [31] proposed that the definition of positive and negative samples of the dataset is the fundamental difference between their performance. Therefore, CCFNet is also built with an anchor-free structure and has reached or even exceeded the accuracy anchor-base detection network.

2.2. FPN-like Methods

The main idea of FPN [7] is to build a top-down feature pyramid to fuse feature maps at different stages of the backbone, and to detect objects of different sizes on feature maps of different scales. This idea is used in different models, You Only Look Once Version 3 (YOLOv3) [8] obtains multi-scale information through multiple convolutions and repeated fusion of the features of the last three stages of the backbone. Adaptively Spatial Feature Fusion (ASFF) [9] adds attention structure based on YOLOv3, which realizes the selective use of the feature information of different stages by controlling the contribution degree of the features of other stages to the current feature. Bi-Directional Feature Pyramid Network (BiFPN) [11] realize adaptive control of the size of FPN by overlapping effective blocks in FPN multiple times. Recursive Feature Pyramid Network (Recursive-FPN) [12] uses recursive FPN to re-input the mixed multi-scale feature map to the backbone, extract the features again, and finally achieve extremely competitive performance. Multi-level Feature Pyramid Network (MLFPN) [13] proposes three modules, Feature Fusion Module (FFM1), Thinned U-shape Module (TUM), and Scale-wise Feature Aggregation Module (SFAM), to integrate semantic information and detailed information by overlapping feature maps multiple times. However, FPN-like methods not only need to fuse feature maps multiple times but also need to build detection head on feature maps of different output sizes to deal with objects of different sizes. Therefore, FPN-like has shortcomings such as a complex model and slow calculation speed.

2.3. FCN-like Methods

With the attention of anchor-free detection networks, the idea of FCN-like gradually shifted from the segmentation task to the object detection task. Different from the FPN-like methods, the FCN-like methods only outputs a feature map that integrates feature information of different scales to the detection head. FCN [15] uses deconvolution layer to upsample the feature map of the last stage of the backbone to restore it to the same size of the input image, thereby preserving the spatial information in the input image to classify each pixel in the feature map. In contrast, the reference [24] adopts a completely symmetrical structure, uses deconvolution to restore the image size, splices and fuses feature information of different scales according to the dimension of the feature map. However, its parameters are few and it is not suitable for large-scale detection or segmentation tasks. CornerNet [21] and CSP [23] use FCN to generate feature maps adapted to the detection head. FCN-like methods have fast calculation speed, but the feature information contained in feature maps of different scales is different. If two feature layers with a large semantic information gap are mixed through dimensionality reduction, a large amount of feature information will be lost, and small objects in the image will be lost.

The difference from the above is that CCF combines the advantages of FPN-like methods and FCN-like methods, and retains more low-level detailed information and high-level semantic information through feature reorganization. In addition, CCFNet also proposes global smooth map that enhances the global perception of the center map to deal with the problem of object occlusion.

3. Methods

This section will elaborate on the proposed Cascaded Cross-layer Fusion Network (CCFNet) for pedestrian detection by exploring the feature fusion and global dependencies.

3.1. Detection Network

The object detection network is usually divided into backbone network, neck, and detection head. The backbone network is responsible for extracting features from the image. A high-quality feature will significantly improve the ability of object localization. The neck is the hub connecting the backbone and detection head. It integrates the features obtained by the backbone network and then inputs the integrated features into the detection head. A high-quality neck can more fully integrate the high-level and low-level information of the image to improve the representation ability of the model. The detection head is responsible for classification and regression.

Most backbone networks [32–36] can be divided into five stages. With the deepening of the network stage, the resolution of the feature map is reduced at a rate of 2 times. In other words, the size of the feature map obtained in the last stage is 1/32 of the input image, which is not friendly to the small object. Previous work [37,38] proposed that the size of the feature map generated in the fifth stage of backbone should be kept at 1/16 of the input image, which can improve the detailed information in the deep feature map to increase the ability to detect small objects.

The input image $I \in R^{3\times H\times W}$ passes through each stage of the backbone network to obtain a set of feature maps $F = \{F_1, F_2, F_3, F_4, F_5\}$. The low-level feature maps generated in the previous stage have more detailed information, but it has a lot of noise. The high-level feature maps generated in later stages have more semantic information. The neck [13,19,39] will reprocesses the feature map set F of the backbone network to obtain feature map f_{det} suitable for the detection head. The detection head [1,40,41] is used to classify and locate the object on the feature map f_{det} output by the neck. In anchor-free detection network, the detection head is defined as $F_{det} = \{cls(f_{det}), regr(f_{det}))\}$, $cls(\cdot)$ represents the classification branch that classifies the object by key points, $regr(\cdot)$ represents the regression branch that locates the object by scale.

3.2. Cascaded Cross-Layer Fusion Module

We combine the advantages of the FPN-like methods and the FCN-like methods, propose Cascaded Cross-layer Fusion module (CCF) to more effectively extract the feature information of the object. CCF uses deconvolution to change the scale of the deep feature map to fuse with the shallow feature map. CCF transfers the deep features to the shallow features in a top-down method, enriching the shallow features while removing noise. However, in this transfer process, the semantic information contained in the deep feature map will continue to be lost. Therefore, CCF supplements missing semantic information by reusing deep feature maps. In this way, the final feature map can not only retain the detailed information in the shallow feature map, but also have the semantic information in the deep feature map. Following [23,37], the final feature map size of CCF is $[H/4, W/4]$. It is worth noting that this is the same size as the feature map of the second stage. The specific implementation process is as follows:

As shown in Figure 2, CCF uses F_4 and F_5 as the source to deliver deep semantic information and denoise the shallow feature maps, because the feature maps generated in the fourth and fifth stages of the backbone network contain rich semantic information. In addition, to reduce the computational complexity of the network, the dimensions of F_4 and F_5 are reduced by 1×1 convolution to generate F_{c4} and F_{c5}. Finally, F_{c4} and F_{c5} are fused to obtain the feature map F_{s4}. F_{s4} retains the semantic information of F_4 and F_5 and continues to be used for subsequent transmission of semantic information. The fusion generation method of feature map F_{s4} can be expressed as:

$$\mathcal{F}_{s4} = Sum(F_{c4}, F_{c5}) \tag{1}$$

where $Sum(\cdot)$ indicates that the fusion method of F_{c4} and F_{c5} is the element-wise addition between the feature maps F_{c4} and F_{c5}.

The feature map F_{s4} will serve two purposes: (1) Regarding F_{s4} as a new source, it will fuse with the new receiver F_3 and continue to convey semantic information from the deep features map. Only the output features of the last two stages in the backbone have the same size. Therefore, it is necessary to perform deconvolution before fusing the shallow features to make it the same size as the previous layer. Therefore, the new source F_{s4} performs up-sampling through deconvolution to obtain a feature map F_{sd4} of the same size as F_{c3}. The process is as follows:

$$\mathcal{F}_{sd4} = DC(F_{s4}) \tag{2}$$

where $DC(\cdot)$ means 4×4 deconvolution. F_{sd4} will be used as the new source, and F_{c3} after dimensionality reduction of feature map F_3 will be fused to obtain F_{s3} according to Equation (1). F_{s3} will be used to transfer the semantic information and detailed information contained in the feature maps F_3, F_4 and F_5. (2) As mentioned before, in purpose (1), the semantic information of the deep feature map will continue to be lost, so the feature map F_{sd4} needs to be transformed into a feature map F_{d4} of size $[H/4, W/4]$ for feature reuse (Equation (2)). F_{d4} can retain the feature representation in the deep feature map.

To continue to transmit the semantic information from the deep feature map and retain the detailed information in F_3, the feature map F_{s3} is transformed to the same size as F_2 through deconvolution, and the resulting F_{sd3} will be used for subsequent operations (Equation (2)).

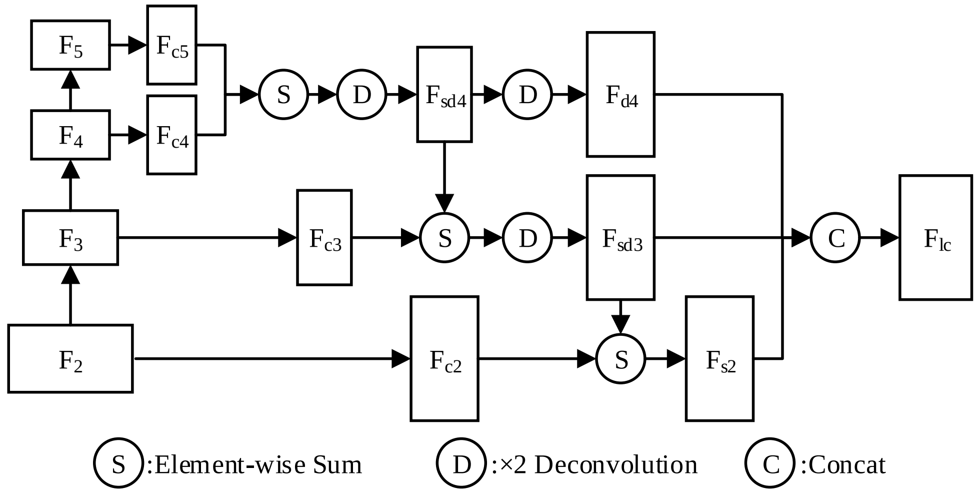


Figure 2. Cascaded Cross-layer Fusion Module (CCF).

The feature map F_3 only contains part of the detailed information, which is not enough to support the network to detect small objects, as shown in the ablation study (Section 4.3). Therefore, CCF refers to the feature map F_2 generated in the second stage, so that the final feature map input to the detection head has more detailed information. However, F_2 contains a lot of noise. CCF uses F_{sd3} containing depth semantics to denoise F_2. In other words, the feature map F_{c2} is obtained by reducing the dimension of F_2 through 1×1 convolution. F_{c2} and F_{sd3} are calculated by Equation (1) to get the feature map F_{s2}. It is worth noting that the size of F_{s2} is $[H/4, W/4]$. There is no need to perform additional processing on F_{s2}.

Finally, CCF merge all feature maps through $Concat(\cdot)$ to obtain a final feature map F_{lc} with rich detailed information and semantic information, F_{lc} can be expressed as:

$$\mathcal{F}_{lc} = Concat(F_{d4}, F_{sd3}, F_{s2}) \tag{3}$$

Following [7], CCF use 3×3 convolution after F_{lc} to reduce the aliasing effect produced in the process of deconvolution and feature fusion.

3.3. Detection Head

Our detection head contains center map, scale map, and global smooth map. Following CSP [23], the center map is equipped with gaussian heat map to locate the object, and scale map is used to determine the size of the object. Although the Gaussian heat map can reduce the weight of negative samples around the object center point, the center map only obtains local perception and lacks global perception. To this end, we add global smooth map, which is fused with the center map, and the generated new center map will have global perception. In addition, considering that the aspect ratio of the pedestrian will change with the change of the pedestrian state, we discarded the scale map that predicts the size of the pedestrian by only predicting the height and fixing the width. The scale map was modified to predict the height and width of pedestrians at the same time.

As shown in Figure 3, the detection head includes center map, global smooth map and scale map. They are all obtained by the feature map F_{lc} generated by CCF through different 1×1 convolutions. Then we use the global smooth map to modify the center map to obtain a more accurate new center map. Finally, the new center map and scale map are used to generate detection results. Optionally, the offset map can be added to the detection head to correct the position of the object.

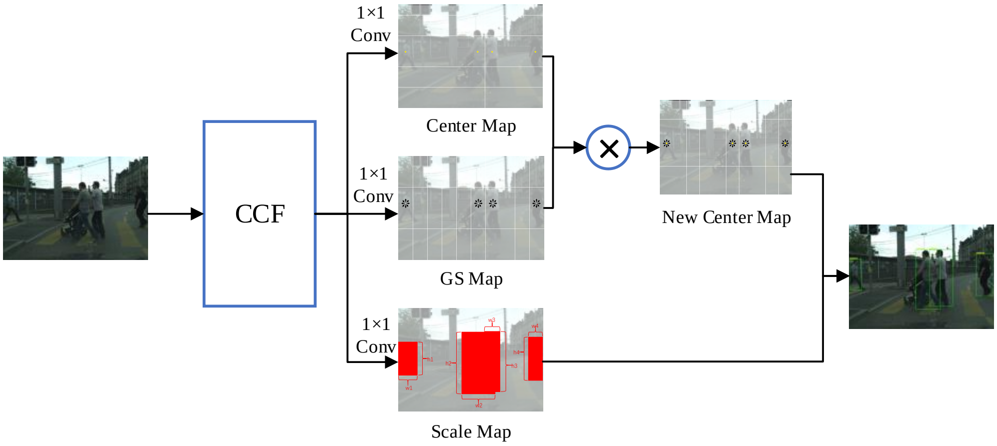


Figure 3. The overall architecture of the detection head mainly includes three map components, namely the center map, the scale map and the global smooth map (GSMap).

3.4. Loss Function

3.4.1. Center Loss

Combined with the global smooth map, the center loss is modified as follows:

$$\mathcal{L}_{center} = -\frac{1}{K}\sum_{i=1}^{W/4}\sum_{j=1}^{H/4}(s_{ij}f_{ij} + (1-s_{ij})b_{ij})log(1-p_{ij}) \tag{4}$$

where

$$\begin{cases} f_{ij} = gs_{ij}(1-p_{ij})^{\gamma} \\ b_{ij} = p_{ij}^{\gamma}(1-M_{ij})^{\beta} \end{cases} \tag{5}$$

from Equations (4) and (5), K is the total number of objects, W and H are the width and height of the input image respectively, s_{ij} represents the true label on the coordinates (i,j), p_{ij} represents the probability of the positive on the coordinates (i,j), gs_{ij} is global smooth confidence, M_{ij} is Gaussian heat map [23], f_{ij} and b_{ij} represent the foreground and background scores in the image, respectively.

3.4.2. Scale Loss

Calculate the scale map by SmoothL1 loss [42] to predict the error between the height and width of the object according to the ground truth. The details of scale loss as follows:

$$\mathcal{L}_{scale} = -\frac{1}{K}(\sum_{k=1}^{K} SmoothL1(h_k, \hat{h}_k) + \sum_{k=1}^{K} SmoothL1(w_k, \hat{w}_k)) \tag{6}$$

where h_k and $\hat{h}_k$ respectively represent the height of the prediction boxes of the network and the height of the ground truth of each positive, w_k and $\hat{w}_k$ respectively represent the width of the prediction boxes of the network and the width of the ground truth of each positive.

3.4.3. Total Loss

Optionally, if the offset map is added to correct the object position, the offset loss is:

$$\mathcal{L}_{offset} = -\frac{1}{K}(\sum_{k=1}^{K} SmoothL1(o_k, \hat{o}_k)) \tag{7}$$

where o_k represents the predicted offset of each positive and $\hat{o}_k$ represents the ground truth of each positive.

Therefore, the complete loss function is:

$$\mathcal{L} = \lambda_c L_{center} + \lambda_s L_{scale} + \lambda_o L_{offset} \tag{8}$$

where λ_c, λ_s, and λ_o are the weights of center loss, scale loss and offset loss, which is set to 0.01, 1 and 0.1 in this experiments. Although on the surface, our loss function is similar to the loss of many methods, from the details we can know that this is different.

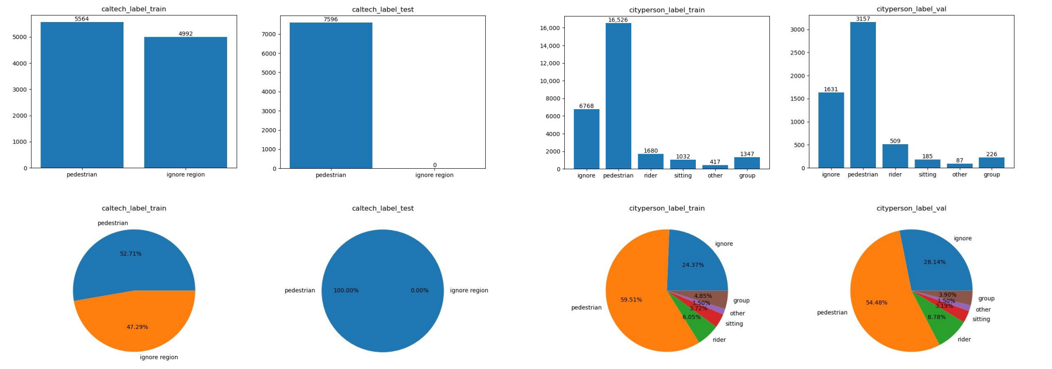
4. Experimental Results

To evaluate the proposed CCFNet, we conducted comparative experiments on Caltech [43,44] and CityPersons [45]. In this section, we introduce the datasets and experimental setting, then verify the effectiveness of the model by the ablation study on the CityPersons dataset, and finally show the compare experimental results with state-of-the-art methods and visualize to verify the superiority of the CCFNet.

The details of each section are as follows: The Section 4.1 introduces the datasets and evaluation indicators of pedestrian detection. The Section 4.2 introduces the experimental setting. The ablation studies on the CityPersons dataset will be analyzed in the Section 4.3. In Section 4.4, the superiority and effectiveness of the model is verified by comparison with other methods on the Caltech and CityPersons datasets. In Section 4.5, visualize the detection results to further illustrate the superiority of CCFNet. Finally, in Section 4.6, we discuss all the experimental results.

4.1. Datasets

The Caltech dataset is about 10 hours of video data, divided into 11 subsets, of which 6 subsets are training sets and 5 subsets are test sets. We divided the video into RGB frames, the training set extracts one image for every 3 frames (total of 42,782 images) and the test set extracts one image for every 30 frames (total of 4024 images). It is observed in Figure 4a,b: the training set contains 5564 pedestrians and 4992 ignored regions, the test set contains 7596 pedestrians and 0 ignored regions.



(a) Caltech_Train **(b)** Caltech_Val **(c)** Cityperson_Train **(d)** Cityperson_Val

Figure 4. The histogram and pie chart represent the distribution statistics of each category in the Caltech and CityPersons datasets. (**a**) represents the label distribution of the training set in the Caltech dataset. (**b**) represents the label distribution of the test set in the Caltech dataset. (**c**) represents the label distribution of the training set in the CityPersons dataset. (**d**) represents the label distribution of the validation set in the CityPersons dataset.

The CityPersons dataset is a subset of the Cityscapes dataset, it has a training set of 2975 images and a validation set of 500 images. From Figure 4c,d, we can clearly known that objects with 59.51% in the training set are marked as pedestrian labels. Objects with 24.37% are marked as ignore labels, including object height pixels less than 20, unclear object status, billboards, etc. Objects with 6.05% are marked as rider labels, Objects with 3.72% are marked as sitting labels. Objects with 1.50% are marked as other labels, including being held of the people. Objects with 4.85% belong to the group. It is worth noting that during the evaluation process, prediction boxes that match rider, sitting, other, ignored areas, etc. It will not be included in the error sample. The label distribution of the validation set is similar to the training set.

Following [44], we using Log-Average Miss Rate (MR^{-2}) as an evaluation indicator. It evaluates the False Positive Per Image (FPPI) of each image between $[0.01, 1]$. The Caltech dataset is evaluated on the Reasonable and Reasonable_Occ=Heavy subsets. The CityPersons dataset is evaluated on the Reasonable, Bare, Partial and Heavy subsets. The definition rules of subsets are shown in Table 1, where inf means infinity.

Table 1. Standards for dividing subsets of the pedestrian datasets.

Subsets	**Height**	**Visibility**
Reasonable	[50, inf]	[0.65, inf]
Bare	[50, inf]	[0.90, inf]
Partial	[50, inf]	[0.65, 0.90]
Heavy	[50, inf]	[0, 0.65]
Reasonable_Occ=Heavy	[50, inf]	[0.2, 0.65]

4.2. Experimental Setting

Unless otherwise specified, The construction of CCFNet follows mmdetection [46] and pedestron [47]. The experiment in this paper is run on a TITAN RTX. On the Caltech dataset, the batch size is set to 16, the initial learning rate is 2×10^{-4}, and the iteration is 20 epoch. On the CityPersons dataset, the batch size is set to 4, the initial learning rate is 2×10^{-4}, and the iteration is 150 epoch. Our experimental setup is based on [48,49].

4.3. Ablation Study

For CCF. To study the effective combination methods of the feature maps, we test the impact of different fusion strategies on model performance. CCF starts with the features of the second stage and keeps the final feature map size as $[H/4, W/4]$, which is consistent with the feature map size of the second stage. As shown in Table 2, s_n represents the feature map generated at the n-th stage of the backbone. It can be easily observed that the last model combines feature maps $\{s_2, s_3, s_4, s_5\}$ obtains the best performance. When s_2 is removed, that is, the combination way $\{s_3, s_4, s_5\}$ gets a poor result, which indicates that the lack of detailed information makes it impossible to accurately locate the object. When s_5 is removed, that is, the combination way $\{s_2, s_3, s_4\}$ also obtains a bad result, which shows that the semantics information contained in the deep features information is crucial. In summary, $\{s_2, s_3, s_4, s_5\}$ is the most suitable combination methods.

Table 2. Ablation study analysis of different combinations of multi-scale feature on the Citypersons dataset.

Feature Maps				**Backbone**	**Reasonable**	**Bare**	**Partial**	**Heavy**
s_2	s_3	s_4	s_5					
✓	✓	-	-	ResNet-50	29.4	22.8	26.9	67.0
-	✓	✓	-	ResNet-50	16.6	12.3	15.4	55.2
-	-	✓	✓	ResNet-50	15.5	10.3	15.4	56.3
✓	✓	✓	-	ResNet-50	16.3	12.4	15.3	54.4
-	✓	✓	✓	ResNet-50	15.4	10.8	14.6	53.7
✓	✓	✓	✓	ResNet-50	10.6	7.1	10.1	48.4

To verify the effectiveness of CCF, we use different neck to connect the backbone network ResNet-50 and the detection head [23], such as FPN [7], Augmented FPN (AugFPN) [50], Attention-guided Context Feature Pyramid Network (ACFPN) [51] and CSP [23]. As shown in the table 3, we can observe that compared with necks of other models, CCF has strong competitiveness in Reasonable, Bare and Partial subsets. In the Heavy subset, CCF is also better than part of the necks. Compared with FPN, CCF reuses the semantic information of deep feature maps to obtain more contextual information in the final feature map. In addition, CCF does not need to output multi-scale feature maps to detect objects. Compared with CSP, CCF removes the noise in the shallow feature map, and retains more detailed information by cascading.

Table 3. Ablation study of different neck module on the Citypersons dataset.

	Reasonable	Bare	Partial	Heavy
ResNet-50 + FPN	11.9	8.1	11.6	48.6
ResNet-50 + AugFPN	11.9	8.5	11.7	50.2
ResNet-50 + ACFPN	11.8	8.2	11.2	50.7
ResNet-50 + CSP	11.2	7.7	10.6	45.7
ResNet-50 + CCF	10.6	7.1	10.1	48.4

For GSMap. Table 4 shows the ablation study on GSMap. The Baseline contains neck and detection head. The neck contains the deconvolution of the fifth stage of ResNet-50 and the detection head contains center map and scale map. Baseline + GSMap means adding GSMap to the detection head. Baseline + GSMap means replacing the neck in the baseline with CCF. Baseline + CCF + GSMap uses CCF to replace the neck in the baseline and adds GSMap to the detection head. As shown in Table 4, we can be observed that adding GSMap separately based on the baseline increases the Reasonable subset by 0.7%, the Bare subset by 0.3%, the Partial subset by 0.8%, and the Heavy subset by 3.7%. If CCF and GSMap work at the same time, compared with baseline + CCF, each subset increases by 0.4%, 0.3%, 0.6% and 5.7%, respectively. This result shows that GSMap enhances the locating ability by making the center map have global feature information. Its performance is enhanced as the effective feature information increases.

Table 4. Ablation study of global smooth map on the Citypersons dataset.

	Backbone	Reasonable	Bare	Partial	Heavy
Baseline	ResNet-50	11.2	7.3	10.8	50.3
Baseline + GSMap	ResNet-50	10.5	7.0	10.0	46.6
Baseline + CCF	ResNet-50	10.6	7.1	10.1	48.4
Baseline + CCF + GSMap	ResNet-50	10.2	6.8	9.5	42.7

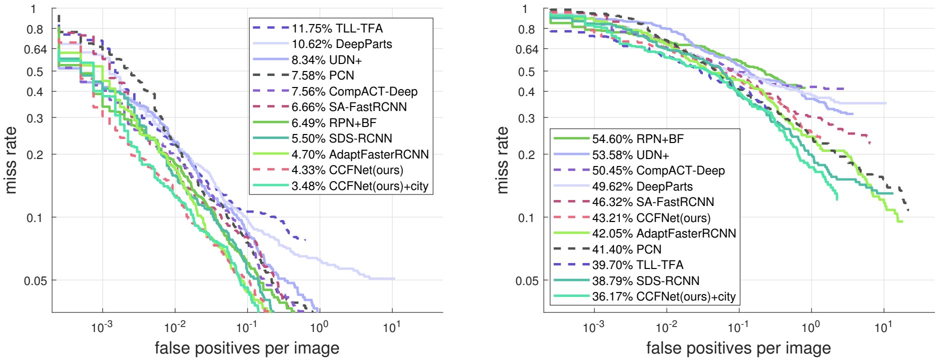
For Scale Prediction. Table 5 shows the impact of scale prediction on CCFNet. Following previous work [23], we set the three scale predictions of height, width and height + width. Compared with the predicted height, height + width increases by 0.6% on the reasonable subset and 4.5% on the heavy subset. Compared with the predicted width, height + width increases by 1.2% on the reasonable subset and 7.2% on the heavy subset. Simultaneously predicting the height and width of the object can further improve the performance of CCFNet. This result is attributed to predicting the height and width of the object at the same time, which can adapt to objects with different aspect ratios, rather than being limited to a certain aspect ratio. In addition, retaining more feature information is conducive to the prediction of object width. From the results of the heavy subsets, it can be concluded that predicting the height and width at the same time helps to deal with dense and overlapping objects.

Table 5. Ablation study of different definitions for scale prediction on the Citypersons dataset.

Scale Prediction	Backbone	Reasonable	Bare	Partial	Heavy
Height	ResNet-50	10.8	7.2	10.7	47.2
Width	ResNet-50	11.4	8.1	11.0	49.9
Height + Width	ResNet-50	10.2	6.8	9.5	42.7

4.4. State-of-the-Art Comparisons

Caltech Dataset: CCFNet compares some excellent methods in reasonable and Reasonable_Occ=Heavy subset. As shown in the Figure 5, CCFNet has 4.33% MR-FPPI on the Reasonable subset, which is 0.37% more advanced than the best method. On the Reasonable_Occ=Heavy subset, CCFNet has 43.21% MR-FPPI, which is also competitive. When the model is initialized on the CityPersons dataset, the performance of CCFNet has increased by 6.04%, surpassing other comparison methods. CCFNet uses feature cascading and reorganization to retain more contextual information, and improves the positioning ability of the center map through global smoothing graph.



(**a**) Reasonable (**b**) Reasonable_Occ=Heavy

Figure 5. The results of various models on the Caltech dataset. (**a**) Compare with existing methods on Reasonable subset. (**b**) Compare with existing methods on the Reasonable_Occ=Heavy subset.

As shown in the Table 6, CCFNet also compares advanced algorithms, such as Repulsion Loss (RepLoss) [38] used to solve the occlusion problem and anchor-free detection network CSP, etc. In the reasonable subset, CCFNet achieved 4.3% MR-FPPI, which is 0.7% and 0.2% lower than that of RepLoss and CSP, respectively. In the Reasonable_Occ=Heavy subset, CCF has reached 43.2% MR-FPPI, which is an increase of 4.7% and 2.6% compared to RepLoss and CSP, respectively. This is an impressive improvement. When the model is initialized on the CityPersons dataset, CCFNet reaches 3.5% on a reasonable subset, and 36.2% on the Reasonable_Occ=Heavy subset. It is proved that CCFNet reuses high-level features in cascaded manner is effective.

Table 6. The results of various models on the Caltech dataset.

	Reasonable	**Reasonable_Occ=Heavy**
ALFNet [52]	6.1	51.0
MGAN [53]	6.8	38.2
HyperLearner [54]	5.5	48.7
RepLoss [38]	5.0	47.9
CSP [23]	4.5	45.8
CCFNet (ours)	4.3	43.2
ALFNet + city [23,52]	4.5	43.4
RepLoss + city [23,38]	4.0	41.8
CSP + city [23]	3.8	36.5
CCFNet + city (ours)	3.5	36.2

CityPersons Dataset: We verify the performance of CCFNet on CityPersons dataset, which contained reasonable, heavy, bare and partial subsets. The comparative experiment results as show in Table 7. MR^{-2} of CCFNet on the reasonable subset is 10.2%, on the bare subset is 6.8%, on the partial subset is 9.5%, and on the heavy subset is 42.7%. In the reasonable subset, CCFNet is 0.4% and 0.3% lower than Attribute-aware Pedestrian Detection (APD) [55] and Mask-Guided Attention Network (MGAN) [53], respectively.

In the heavy subset, CCFNet is increased by 7.1% and 4.5% compared with APD and MGAN, respectively. It can be seen that CCFNet achieved best performance beyond other comparison methods. It reflects the strong competitiveness of CCFNet.

Table 7. The results of various models on the CityPersons dataset.

	Backbone	Reasonable	Bare	Partial	Heavy
TLL [37]	ResNet-50	15.5	10.0	17.2	53.6
TLL + MRF [37]	ResNet-50	14.4	9.2	15.9	52.0
RepLoss [38]	ResNet-50	13.2	7.6	16.8	56.9
OR-CNN [56]	VGG-16	12.8	6.7	15.3	55.7
ALFNet [52]	ResNet-50	12.0	8.4	11.4	51.9
CSP [23]	ResNet-50	11.0	7.3	10.4	49.3
APD [55]	ResNet-50	10.6	7.1	9.5	49.8
MGAN [53]	VGG-16	10.5	-	-	47.2
CCFNet (ours)	ResNet-50	10.2	6.8	9.5	42.7

4.5. Visualization

To further illustrate the superiority of CCFNet, we visualized the detection results on the CityPersons dataset, as shown in Figure 6. The first line (a) represents the original image in the validation set of the CityPersons dataset. The second line (b) represents the ground truth. The third line (c) represents the visualization result of the CSP. And the fourth line (d) represents the visu.alization result of CCFNet. The visualization results of CSP and CCFNet rely on the same confidence.

To show the effectiveness of the CCFNet, we selected three images from different scenes to compared with CSP. The first image belongs to a crowded scene. The second image belongs to a simple scene containing small objects. The third image is a scene with low visibility, low exposure, and small objects. The visualization result as show in Figure 6. It can be seen that in the first image, CSP and CCFNet generate a large number of detection boxes, but CCFNet has fewer false detection boxes. In addition, CCFNet can better solve the problem of multiple detection boxes for one single object. From the second image, CSP and CCFNet have the problem of overlapping detection boxes, but CSP has extremely bad results. In contrast, CCFNet has better visualization. From the third image, CSP can detect small objects in the image, but it also gets a lot of objects that should not be detected. In contrast, CCFNet avoids this problem. Therefore, CCFNet not only has good performance, but its visualization results are also robust.

As shown in Figure 7, the first line (a) represents the original image in the validation set of the CityPersons dataset. The second line (b) represents the heat map of the ACFPN. The third line (c) represents the heat map of the CSP. And the fourth line (d) represents the heat map of CCFNet. We also selected the images of the three scenes for comparison. The three images respectively cover complex environments, crowded scenes, and general scenes. It can be seen that the highlight of ACFPN presents a discrete distribution, the highlight of CSP presents a concentrated distribution, and the highlight of CCFNet is multi-peak. The ACFPN can not distinguish which type of person belongs to, and can not cope with the crowded state of objects, this is related to the fact that ACFPN is a general object detection network. The CSP responds to certain backgrounds, which makes CSP a bad visualization result, even though it has a low error detection rate. The CCFNet will not over-respond to the background and can distinguish the categories of people, it not only has a lower error detection rate, but its visualization results are also more optimistic.

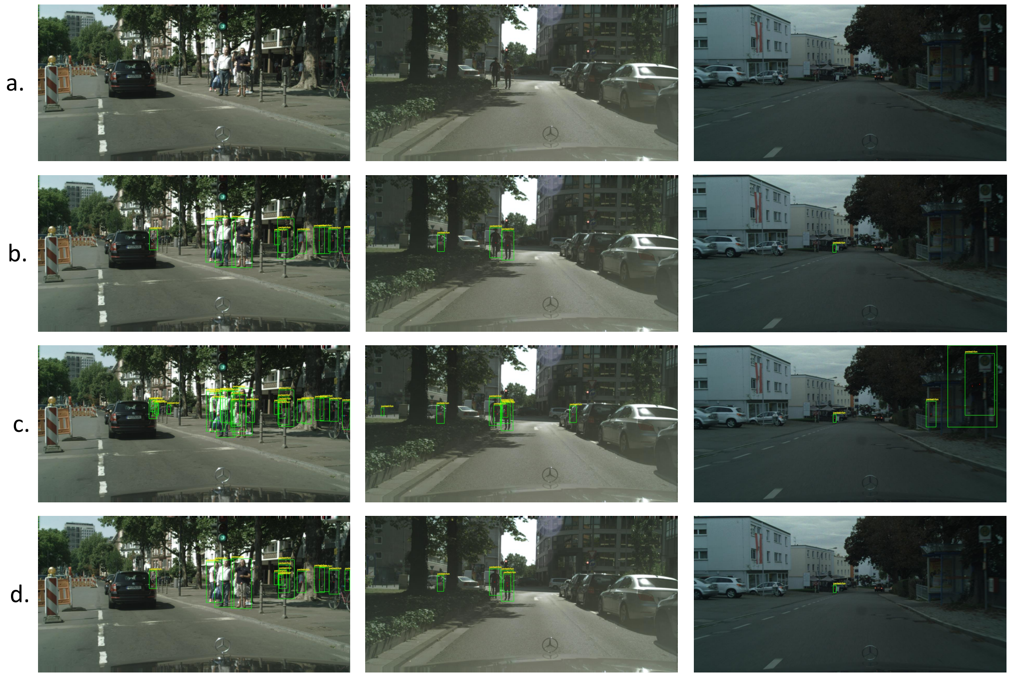


Figure 6. Visualization results of CCFNet and CSP do not limit the visibility of pedestrian objects. (**a**) Input the original image for the CityPersons dataset; (**b**) is the ground truth corresponding to (**a**); (**c**) is the visualization result of CSP; (**d**) is the visualization result of CCFNet.

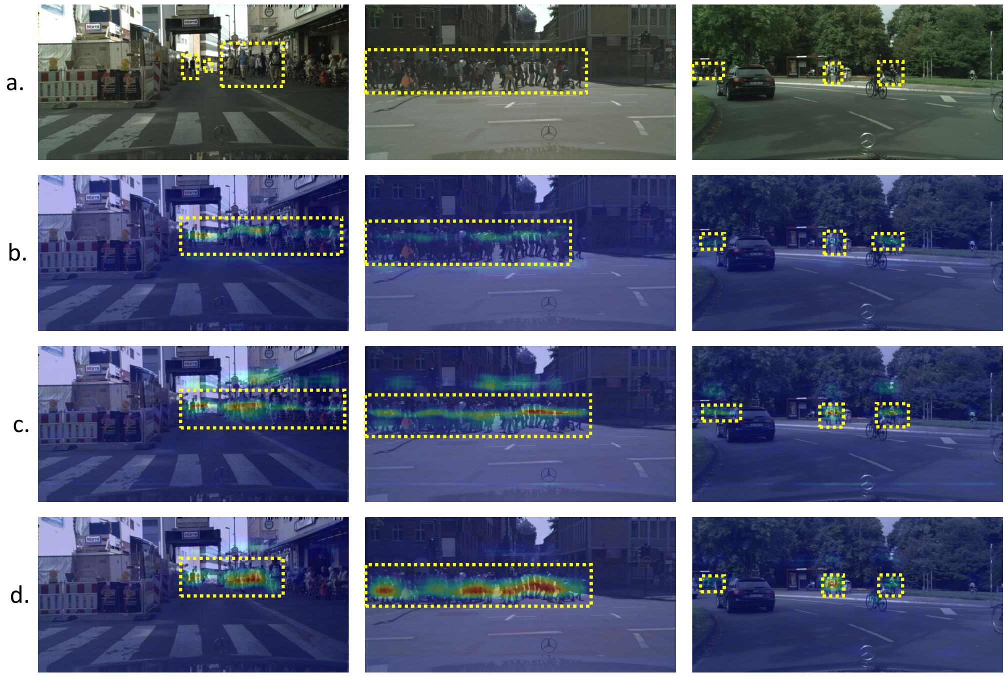


Figure 7. Visualization results of ACFPN, CSP, and CCFNet. (**a**) Input the original image for the CityPersons dataset; (**b**) is the visualization result of ACFPN; (**c**) is the visualization result of CSP; (**d**) is the visualization result of CCFNet.

4.6. Discussion

The proposal of CCFNet is influenced by the anchor-free object detection network. In the anchor-free network, how to make the neck effectively use the feature representation extracted by the backbone network will directly affect the performance of the detection head. Previous work [50,51] has achieved good performance in general object detection, but it can not be generalized to some special tasks, such as pedestrian detection.

Table 2 shows the ablation experiment of multi-scale features in the CCF module. By combining the feature maps of different stages, the optimal feature map combination is discussed. CCF reduces the noise in the shallow feature map by cascading and reusing deep semantic information, while retaining the semantic information lost due to dimensionality reduction operations. The purpose of this is to make the final feature map have more features.

Table 3 shows the comparative experiments between CCF and other necks. The previously proposed FPN-like methods and FCN-like methods achieve the most advanced performance in general object detection, but they are not suitable for pedestrian detection tasks. CCF module shows a very competitive performance.

Table 4 shows the ablation experiment of GSMap. The center map reduces the weight of negative samples through the Gaussian heat map, but does not change the shortcomings of convolution operation that can only obtain partial global information [57–59]. The proposal of GSMap can enable the center map to obtain more global information. In addition, according to the results of the heavy subset. It not only proves that the congestion problem between objects can not be completely solved by enhancing the semantic information in the feature map, but also requires additional modules for assistance, such as GSMap.

Table 5 shows the experiment of object scale prediction. The previous work only determines the size of the object by predicting the height [23,48]. We have proved through experiments that predicting the height and width of objects at the same time is the most suitable for CCFNet. In addition, this can also help cope with dense and overlapping problems.

Figure 5 and Table 6 show the comparative experiments of CCFNet with other advanced algorithms on the Caltech dataset. Table 7 shows the comparative experiments of CCFNet with other advanced algorithms on the Citypersons dataset. Their results prove the effectiveness of CCFNet.

5. Conclusions

In this paper, we proposed Cascaded Cross-layer Fusion module (CCF), which combines deep semantics and shallow details to obtain features, which will obtain more contextual semantic information. In order to cope with the situation of highly congested and severely occluded objects, we designed global smooth map (GSMap) and improved center loss function, which can effectively solve this problem at a small cost. Cascaded Cross-layer Fusion Network (CCFNet) can achieve better performance without relying on anchor points, multiple key points and complex post-processing. Finally, we conducted a large number of experiments on Caltech and CityPersons datasets to verify the superiority of CCFNet. Although the model introduces dimensionality reduction operations in the design process to reduce the computational complexity of the model, the final model still uses a large number of parameters that cannot meet the requirements of the real-time system. Therefore, designing an effective lightweight module is the focus of our next work.

Author Contributions: Formal analysis, Z.G., Y.S. and X.X.; methodology, Z.D. and Y.S.; project administration, Z.D.; validation, Z.G. and X.X.; visualization, Z.G and X.X.; writing–original draft, Z.D.; writing–review and editing, Z.G., Y.S. and X.X. All authors have read and agreed to the published version of the manuscript.

Funding: Not applicable.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: Not applicable.

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

References

1. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 91–99. [CrossRef]
2. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. *European Conference on Computer Vision*; Springer: Amsterdam, The Netherlands, 2016; pp. 21–37.
3. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
4. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
5. Li, Z.; Tang, J.; Zhang, L.; Yang, J. Weakly-supervised Semantic Guided Hashing for Social Image Retrieval. *Int. J. Comput. Vis.* **2020**, *128*, 2265–2278. [CrossRef]
6. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
7. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
8. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
9. Liu, S.; Huang, D.; Wang, Y. Learning spatial fusion for single-shot object detection. *arXiv* **2019**, arXiv:1911.09516.
10. Ghiasi, G.; Lin, T.Y.; Le, Q.V. Nas-fpn: Learning scalable feature pyramid architecture for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7036–7045.
11. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10781–10790.
12. Qiao, S.; Chen, L.C.; Yuille, A. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 10213–10224.
13. Zhao, Q.; Sheng, T.; Wang, Y.; Tang, Z.; Chen, Y.; Cai, L.; Ling, H. M2det: A single-shot object detector based on multi-level feature pyramid network. *Aaai Conf. Artif. Intell.* **2019**, *33*, 9259–9266. [CrossRef]
14. Zhang, D.; Zhang, H.; Tang, J.; Wang, M.; Hua, X.; Sun, Q. Feature pyramid transformer. In *Proceedings of the European Conference on Computer Vision*; Springer: Amsterdam, The Netherlands, 2020; pp. 323–339.
15. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
16. Huang, L.; Yang, Y.; Deng, Y.; Yu, Y. Densebox: Unifying landmark localization with end to end object detection. *arXiv* **2015**, arXiv:1509.04874.
17. Zhu, Z.; Li, Z. Online Video Object Detection via Local and Mid-Range Feature Propagation. In Proceedings of the 1st International Workshop on Human-Centric Multimedia Analysis, Seattle WA, USA, 10–14 October 2020; pp. 73–82.
18. Li, Z.; Tang, J.; Mei, T. Deep collaborative embedding for social image understanding. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 2070–2083. [CrossRef]
19. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef]
20. Zhou, H.; Li, Z.; Ning, C.; Tang, J. Cad: Scale invariant framework for real-time object detection. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Venice, Italy, 22–29 October 2017; pp. 760–768.
21. Law, H.; Deng, J. Cornernet: Detecting objects as paired keypoints. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 734–750.
22. Li, Z.; Sun, Y.; Tang, J. CTNet: Context-based Tandem Network for Semantic Segmentation. *arXiv* **2021**, arXiv:2104.09805.
23. Liu, W.; Liao, S.; Ren, W.; Hu, W.; Yu, Y. High-level semantic feature detection: A new perspective for pedestrian detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5187–5196.
24. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*; Springer: Munich, Germany, 2015; pp. 234–241.
25. Sun, Y.; Li, Z. SSA: Semantic Structure Aware Inference for Weakly Pixel-Wise Dense Predictions without Cost. *arXiv* **2021**, arXiv:2111.03392.
26. Cai, Z.; Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162.
27. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
28. Law, H.; Teng, Y.; Russakovsky, O.; Deng, J. Cornernet-lite: Efficient keypoint based object detection. *arXiv* **2019**, arXiv:1904.08900.
29. Zhou, X.; Zhuo, J.; Krahenbuhl, P. Bottom-up object detection by grouping extreme and center points. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 850–859.
30. Zhou, X.; Wang, D.; Krähenbühl, P. Objects as points. *arXiv* **2019**, arXiv:1904.07850.

31. Zhang, S.; Chi, C.; Yao, Y.; Lei, Z.; Li, S.Z. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9759–9768.
32. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
33. Pouyanfar, S.; Sadiq, S.; Yan, Y.; Tian, H.; Tao, Y.; Reyes, M.P.; Shyu, M.L.; Chen, S.C.; Iyengar, S.S. A survey on deep learning: Algorithms, techniques, and applications. *ACM Comput. Surv. (CSUR)* **2018**, *51*, 1–36. [CrossRef]
34. Adjabi, I.; Ouahabi, A.; Benzaoui, A.; Taleb-Ahmed, A. Past, present, and future of face recognition: A review. *Electronics* **2020**, *9*, 1188. [CrossRef]
35. Dong, S.; Wang, P.; Abbas, K. A survey on deep learning and its applications. *Comput. Sci. Rev.* **2021**, *40*, 100379. [CrossRef]
36. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
37. Song, T.; Sun, L.; Xie, D.; Sun, H.; Pu, S. Small-scale pedestrian detection based on topological line localization and temporal feature aggregation. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 536–551.
38. Wang, X.; Xiao, T.; Jiang, Y.; Shao, S.; Sun, J.; Shen, C. Repulsion loss: Detecting pedestrians in a crowd. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7774–7783.
39. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.
40. Yang, Z.; Liu, S.; Hu, H.; Wang, L.; Lin, S. Reppoints: Point set representation for object detection. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 9657–9666.
41. Tian, Z.; Shen, C.; Chen, H.; He, T. Fcos: Fully convolutional one-stage object detection. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 9627–9636.
42. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
43. Dollár, P.; Wojek, C.; Schiele, B.; Perona, P. Pedestrian detection: A benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 304–311.
44. Dollar, P.; Wojek, C.; Schiele, B.; Perona, P. Pedestrian detection: An evaluation of the state of the art. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *34*, 743–761. [CrossRef]
45. Zhang, S.; Benenson, R.; Schiele, B. Citypersons: A diverse dataset for pedestrian detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3213–3221.
46. Chen, K.; Wang, J.; Pang, J.; Cao, Y.; Xiong, Y.; Li, X.; Sun, S.; Feng, W.; Liu, Z.; Xu, J.; et al. MMDetection: Open MMLab Detection Toolbox and Benchmark. *arXiv* **2019**, arXiv:1906.07155.
47. Hasan, I.; Liao, S.; Li, J.; Akram, S.U.; Shao, L. Generalizable Pedestrian Detection: The Elephant in the Room. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 11328–11337.
48. Wang, W. Adapted Center and Scale Prediction: More Stable and More Accurate. *arXiv* **2020**, arXiv:2002.09053.
49. Krizhevsky, A. One weird trick for parallelizing convolutional neural networks. *arXiv* **2014**, arXiv:1404.5997.
50. Guo, C.; Fan, B.; Zhang, Q.; Xiang, S.; Pan, C. Augfpn: Improving multi-scale feature learning for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 12595–12604.
51. Cao, J.; Chen, Q.; Guo, J.; Shi, R. Attention-guided context feature pyramid network for object detection. *arXiv* **2020**, arXiv:2005.11475.
52. Liu, W.; Liao, S.; Hu, W.; Liang, X.; Chen, X. Learning efficient single-stage pedestrian detectors by asymptotic localization fitting. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 618–634.
53. Pang, Y.; Xie, J.; Khan, M.H.; Anwer, R.M.; Khan, F.S.; Shao, L. Mask-guided attention network for occluded pedestrian detection. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 4967–4975.
54. Mao, J.; Xiao, T.; Jiang, Y.; Cao, Z. What can help pedestrian detection? In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3127–3136.
55. Zhang, J.; Lin, L.; Zhu, J.; Li, Y.; Chen, Y.c.; Hu, Y.; Hoi, C.S. Attribute-aware pedestrian detection in a crowd. *IEEE Trans. Multimed.* **2020**, *23*, 3085–3097. [CrossRef]
56. Zhang, S.; Wen, L.; Bian, X.; Lei, Z.; Li, S.Z. Occlusion-aware R-CNN: Detecting pedestrians in a crowd. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 637–653.
57. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929.
58. Zhang, D.; Zhang, H.; Tang, J.; Hua, X.S.; Sun, Q. Self-Regulation for Semantic Segmentation. In Proceedings of the IEEE International Conference on Computer Vision, Montreal, Canada, 11–17 October 2021; pp. 6953–6963.
59. Srinivas, A.; Lin, T.Y.; Parmar, N.; Shlens, J.; Abbeel, P.; Vaswani, A. Bottleneck transformers for visual recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 16519–16529.