Efficient Single-Stage Pedestrian Detector by Asymptotic Localization Fitting and Multi-Scale Context Encoding

Wei Liu, Shengcai Liao, *Senior Member, IEEE*, and Weidong Hu

***Abstract*— Though Faster R-CNN based two-stage detectors have witnessed significant boost in pedestrian detection accuracy, they are still slow for practical applications. One solution is to simplify this working flow as a single-stage detector. However, current single-stage detectors (e.g. SSD) have not presented competitive accuracy on common pedestrian detection benchmarks. Accordingly, a structurally simple but effective module called *Asymptotic Localization Fitting* (ALF) is proposed, which stacks a series of predictors to directly evolve the default anchor boxes of SSD step by step to improve detection results. Additionally, combining the advantages from residual learning and multi-scale context encoding, a bottleneck block is proposed to enhance the predictors' discriminative power. On top of the above designs, an efficient single-stage detection architecture is designed, resulting in an attractive pedestrian detector in both accuracy and speed. A comprehensive set of experiments on two of the largest pedestrian detection datasets (i.e. CityPersons and Caltech) demonstrate the superiority of the proposed method, comparing to the state of the arts on both the benchmarks.**

***Index Terms*— Pedestrian detection, convolutional neural networks, asymptotic localization fitting.**

I. Introduction

PEDESTRIAN detection is a key problem in a number of real-world applications including auto-driving systems and surveillance systems, and is required to have both high accuracy and real-time speed. Traditionally, scanning an image in a sliding-window paradigm is a common practice for object detection. In this paradigm, designing hand-crafted features [1]–[4] is of critical importance for state-of-the-art performance, which still remains difficult. Beyond early studies focusing on hand-crafted features, R-CNN [5] is among the first to introduce CNN into object detection [6], [7], followed by Fast R-CNN [8] and Faster R-CNN [9]. Specifically, Faster R-CNN proposed Region Proposal Network (RPN) to generate proposals in a unified framework. Beyond its success on generic object detection, numerous Faster R-CNN adapted detectors were proposed and demonstrated better accuracy for pedestrian detection [10], [11].

However, when the processing speed is considered, Faster R-CNN is still unsatisfactory because it requires two-stage processing, namely proposal generation and classification of ROI-pooling features. Alternatively, as a representative one-stage detector, Single Shot MultiBox Detector (SSD) [12] discards the second stage of Faster R-CNN [9] and directly regresses the default anchors into detection boxes. Though faster, SSD [12] lags behind Faster R-CNN in terms of accuracy. Based on the efficiency of SSD, it motivates us to think what the key is in Faster R-CNN and whether this key could be transfered to SSD. Since both SSD and Faster R-CNN have default anchor boxes, we guess that the key is the two-step prediction of the *default anchor boxes*, with RPN one step, and prediction of ROIs another step, but not the ROI-pooling module. A recent work called Cascade R-CNN [13] has proved that Faster R-CNN can be further improved by applying multi-step ROI-pooling and prediction after RPN. Besides, another recent work called RefineDet [14] suggests that ROI-pooling can be replaced by a convolutional transfer connection block after RPN. Based on the above analysis, it seems possible that the default anchors in SSD could be directly processed in multi-steps for an even simpler solution, with neither RPN nor ROI-pooling.

Consequently, in this paper, a simple but effective module called Asymptotic Localization Fitting (ALF) is proposed. It directly starts from the default anchors in SSD, and convolutionally evolves all anchor boxes step by step, pushing more anchor boxes closer to groundtruth boxes. This multi-step processing is inspired from the Cascade R-CNN, but is fully convolutional, without RPN and ROI-pooling.

Additionally, inspired by the impressive success of residual learning by skip connection [15] and multi-context encoding by dilated convolution [16], a bottleneck design is proposed to enhance the discriminant power of the convolutional predictors.

Manuscript received September 30, 2018; revised June 15, 2019; accepted August 26, 2019. Date of publication September 16, 2019; date of current version November 7, 2019. This work was supported by the Chinese National Natural Science Foundation Project 61672521. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Husrev T. Sencar. *(Corresponding author: Shengcai Liao.)*

W. Liu is with the National Key Laboratory of Science and Technology on ATR, College of Electronic Science, National University of Defense Technology, Changsha 410073, China, and also with the Center for Biometrics and Security Research & National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: liuwei16@nudt.edu.cn).

S. Liao is with the Inception Institute of Artificial Intelligence, Abu Dhabi, United Arab Emirates (e-mail: scliao@nlpr.ia.ac.cn).

W. Hu is with the National Key Laboratory of Science and Technology on ATR, College of Electronic Science, National University of Defense Technology, Changsha 410073, China (e-mail: wdhu@nudt.edu.cn).

Digital Object Identifier 10.1109/TIP.2019.2938877

On top of the above designs, a novel pedestrian detection architecture is constructed, which, built upon the single-stage detection framework, significantly improves the pedestrian detection accuracy while maintaining the efficiency of single-stage detectors. Extensive experiments and analysis on two large-scale pedestrian detection datasets demonstrate the effectiveness of the proposed method independent of the backbone network. It is worth noting that when a stricter IoU threshold of 0.75 for evaluation is set, the performance gain of the proposed method over the state of the arts is more significant, indicating that the proposed method is indeed capable of achieving more accurate localization.

To sum up, the main contributions of this work lie in:

- A module called ALF is proposed to overcome the limitations of single-stage detectors;
- Two bottleneck blocks are proposed to enhance the convolutional predictors' discriminative power;
- On top of the above two designs, a simple but effective single-stage pedestrian detection architecture is presented;
- The proposed method achieves new state-of-the-art results on two of the largest pedestrian benchmarks (i.e., CityPerson [11], Caltech [17]).

This work is built upon our preliminary work (ALFNet [18]) recently accepted by the European Conference on Computer Vision (ECCV) 2018. For the convenience of distinction, we called the improved method ALFNetV2. The main improvement is the development of the new detection heads, which reduce the memory burdens while achieves better detection accuracy. We demonstrate that the effectiveness of the ALF design is independent of different detection heads. Main changes contained in this paper are as follows:

- Two more discriminative bottleneck blocks are proposed to further enhance the detection head of ALFNet, which is introduced in Section III-C and experimentally demonstrated in Section IV-B. Specifically, comparisons of different detection heads on the CityPersons validation set are given in Table VII, and extended ablation studies and analyses are included to demonstrate the superiority of the proposed bottleneck blocks over the original convolutional detection head on both memory efficiency and detection accuracy.
- For three variants of the proposed methods, extensive comparisons to the state of the arts in the CityPersons and Caltech benchmark are included in Table IX and Fig.8, respectively.
- The motivations of the proposed method are highlighted in Section I. The relationships to two most related works are pictorially illustrated in Fig.1 and presented in details in the final part of Section II. Besides, some recent related works on pedestrian detection are included in Section II.

II. Related Work

With the emergency of CNN, generic object detection has gained great success, and [5]–[7] are the first trials to introduce CNN in object detection. Following that, various CNN-based detectors are proposed which can be roughly classified into two categories. The first type can be named as two-stage methods, including Fast R-CNN [8], Faster R-CNN [9], R-FCN [19], etc. These two-stage detectors first generate plausible region proposals, then refine proposals by another sub-network for bounding box classification and regression. To generate proposals in a unified framework, Faster R-CNN [9] makes the RPN sharing the base network with the detection sub-network, thus can be trained jointly. However, it still needs region proposal generation and classification, thus its speed is limited by repeated CNN feature extraction and evaluation. Following that, numerous methods have tried to improve the detection performance by focusing on network architecture [19]–[22], training strategy [23], [24], auxiliary context mining [25]–[27], and so on, while the heavy computational burden is still an unavoidable problem.

The second type [12], [28], [29], known as the single-stage method, aims at speeding up detection by removing the region proposal generation stage. This kind of detector directly regresses pre-defined anchors from multiple layers of the base network and thus is more computationally efficient. However, single-stage methods generally yield less satisfactory results than two-stage methods. To improve the detection accuracy of SSD [12], some following methods [30], [31] pay attention to enhance the feature representation of CNN, while some others [32], [33] target at the positive-negative imbalance problem via novel classification strategies. However, less work has been done for pedestrian detection in the single-stage framework.

In terms of pedestrian detection, driven by the success of R-CNN [5], a series of pedestrian detectors are proposed in the two-stage framework. Hosang et al. [34] firstly utilize the SCF detector [2] to generate proposals which are then fed into a network like R-CNN. In TA-CNN [35], the ACF detector [3] is employed for proposal generation, then pedestrian detection is jointly optimized with an auxiliary semantic task. DeepParts [36] uses the LDCF detector [4] to generate proposals and then trains an ensemble of CNN for detecting different parts. Different from the above methods with resort to traditional detectors for proposal generation, RPN+BF [10] adapts the original RPN in Faster R-CNN [9] to generate proposals, then learns boosted forest classifiers on top of these proposals. Towards the multi-scale detection problem, MS-CNN [37] exploits multi-layer featuremaps of a base network to generate proposals, followed by a detection network aided by context reasoning. SA-FastRCNN [38] jointly trains two networks to detect pedestrians of large scales and small scales respectively, based on the proposals generated from ACF detector [3]. Brazil *et al.* [39], Du *et al.* [40] and Mao *et al.* [41] further improve the detection performance by combining semantic information. Xu *et al.* [42] proposes to use the auxiliary feature representation from the infrared images to enhance the detection performance in the RGB images. TLL [43] proposes to detect an object by paired keypoints and the corresponding link edges between them. It achieves significant improvement on Caltech [17], especially for small-scale pedestrian instances. Recently, there are some works focusing on the occlusion problem in pedestrian detection. For example, Wang *et al.* [44] designs a novel regression loss for crowded pedestrian detection based on

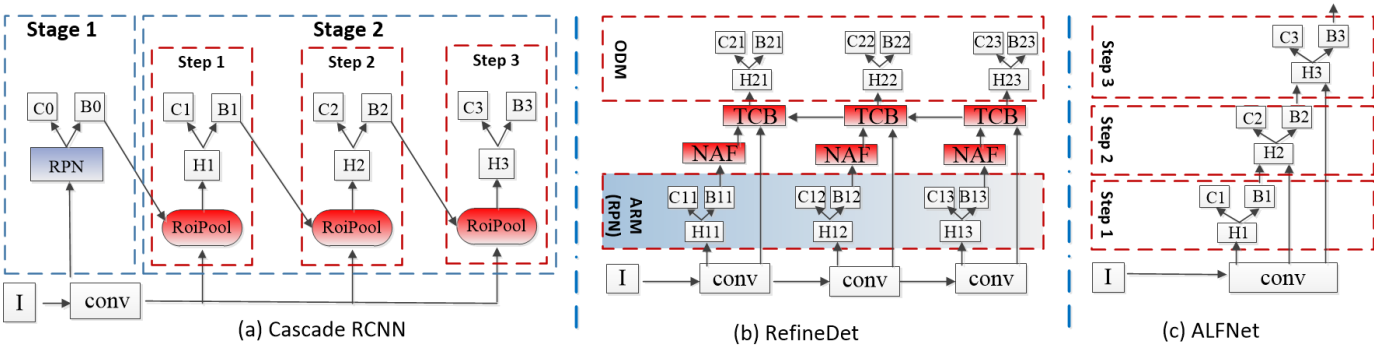


Fig. 1. The architectures of three detection pipelines. (a) Cascade R-CNN [13], a Faster R-CNN [9] based two-stage detector. (b) RefineDet [14], a single-stage detector. (c) The proposed ALFNet, a simple single-stage detector. For better visualization, only the convolutional layers (conv) used for detection are displayed. I: input image. H: detection network head. C: classification scores. B: regressed bounding boxes. ARM: Anchor Refinement Module. ODM: Object Detection Module. NAF: Negative Anchor Filtering. TCB: Transfer Connection Block.

Faster R-CNN [9]. Similarly, OR-CNN [45] also proposes two strategies separately on the two stages of Faster R-CNN for crowded cases. Bi-box [46] proposes an auxiliary sub-network to predict the visible part of a pedestrian instance, and [47] focuses on the discriminative feature learning based on the original SSD architecture. Reference [48], [49] use the attention mechanism to enhance the pedestrian instances in crowded scene. However, compared to accuracy, less attention has been paid to the speed of pedestrian detectors.

Most recently, Cascade R-CNN [13] for generic object detection proposes to train a sequence of detectors with increasing IoU thresholds via the proposals generated by RPN, so that the predicted detections can be improved step by step. The proposed method shares the similar idea of multi-step refinement to the Cascade R-CNN. However, the differences lie in two aspects. Firstly, Cascade R-CNN is towards a better detector based on the Faster R-CNN framework, but we try to answer what the key in Faster R-CNN is and whether this key could be used to enhance SSD for speed and accuracy. The key we get is the multi-step prediction of the default anchors, with RPN one step, and prediction of ROIs another step. Given this finding, the default anchors in SSD could be processed in multi-steps, in fully convolutional way without ROI pooling. Secondly, as shown in Fig. 1 (c), in the proposed method, all default anchors are convolutionally processed in multi-steps, without re-sampling or iterative ROI pooling. In contrast, as shown in Fig. 1 (a), the Cascade R-CNN converts the detector part of the Faster R-CNN into multi-steps, which unavoidably requires RPN, and iteratively applying anchor selection and individual ROI pooling within that framework.

Another close related work to ours is the RefineDet [14] proposed for generic object detection. As shown in Fig. 1 (b), it contains two inter-connected modules, with the former one filtering out negative anchors by objectness scores and the latter one refining the anchors from the first module. A transfer connection block is further designed to transfer the features between these two modules. The proposed method differs from RefineDet [14] mainly in two folds. Firstly, we stack the detection module on the backbone feature maps without the transfer connection block, thus is simpler and faster. Secondly, we do not use objectness scores for anchor box filtering. We consider that scores from the first step are not confident enough for decisions, and the filtered "negative" anchor boxes may contain hard positives that may still have chances to be corrected in latter steps. Therefore, in the proposed method, all default anchors are equally processed in multi-steps without filtering.

Finally, thanks to the breakthrough of the ResNet [15] in deepening the CNN, recent works adapting ResNet have achieved top performance in dense prediction tasks, like object detection [9], [19], [22], [50] and semantic segmentation [51]–[54]. The key idea of the ResNet is the residual learning, which incorporates identity connections from the input to the output to alleviate the gradient degradation problem. Extensive image classification experiments in [15] demonstrate that equipped with the residual learning block, ResNet can achieve better performance than standard convolutional layers based heavier networks, like VGG [55]. On the other hand, dilated convolutions have been extensively explored to encode multi-scale context for better semantic segmentation accuracy in recent works [56]–[58], while whether it is beneficial for pedestrian detection is not aware of. By combining the advantages from the residual learning and multi-scale context encoding, in this paper a bottleneck design is proposed to enhance the discriminative power of the convolutional predictors in single-stage detectors.

III. Approach

A. Preliminary

Our method is built on top of the single-stage detection framework, here we give a brief review of this type of methods.

In single-stage detectors, multiple feature maps with different resolutions are extracted from a backbone network (e.g. VGG [55], ResNet [15]), these multi-scale feature maps can be defined as follows:

$$\Phi_n = f_n(\Phi_{n-1}) = f_n(f_{n-1}(\dots f_1(I))), \tag{1}$$

where I represents the input image, $f_n(.)$ is an existing layer from a base network or an added feature extraction layer, and Φ_n is the generated feature maps from the nth layer. These feature maps decrease in size progressively thus multi-scale

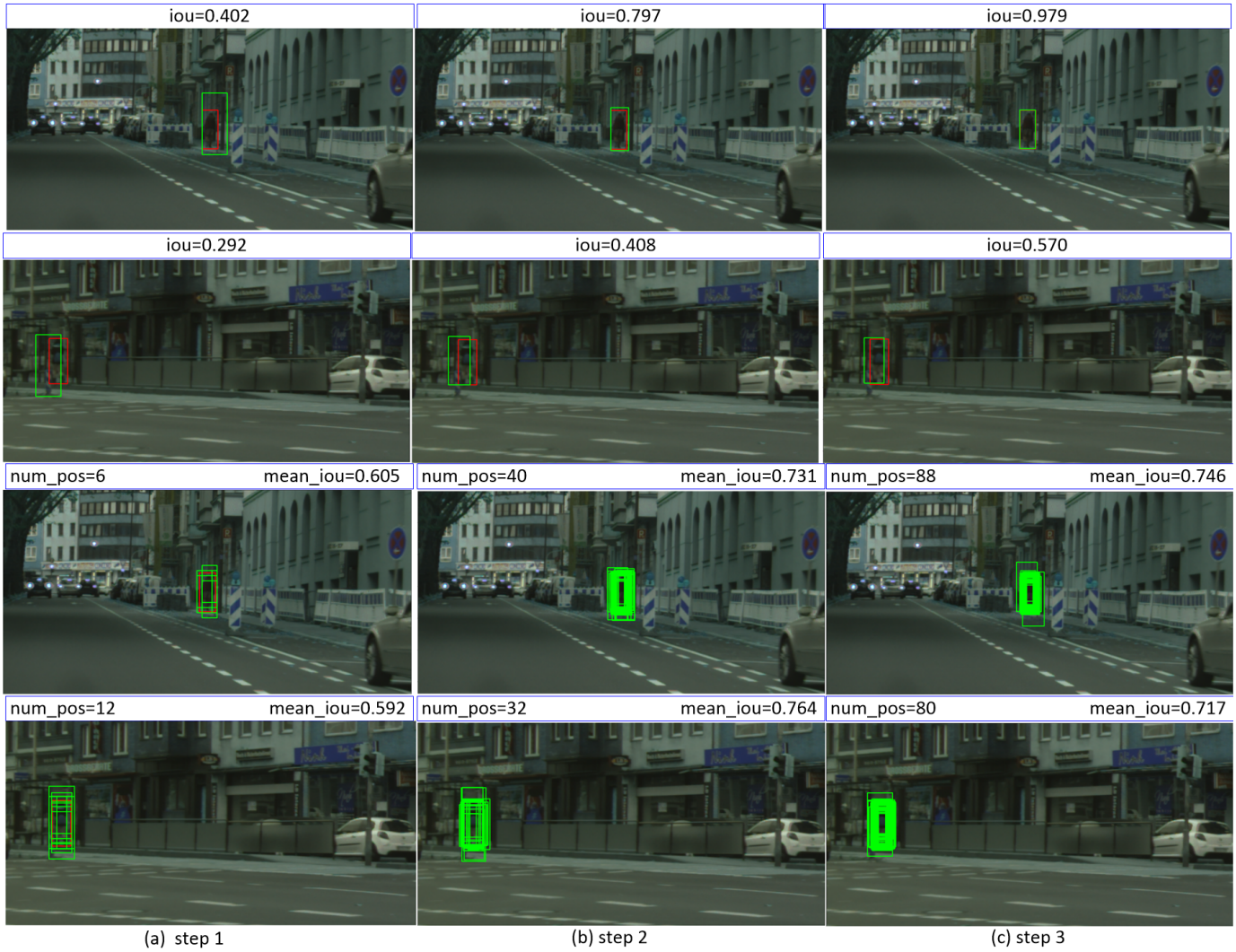


Fig. 2. Two examples from the CityPersons [11] training data. Green and red rectangles are anchor boxes and groundtruth boxes, respectively. (a), (b) and (c) depict the evolution of the default anchor boxes from the first step to the third step. In the first two rows, we randomly choose a single default anchor box overlapped with the groundtruth boxes, the IoU values in the top of the images show that the default anchor boxes are moving towards the groundtruth boxes step by step. In the last two rows, we depict all anchor boxes matched with the groundtruth above the IoU threshold of 0.5. Values on the upper left of the image represent the number of matched anchor boxes, and values on the upper right of the image denote the mean value of overlaps with the groundtruth from all matched anchor boxes, which indicates that more anchor boxes with higher-quality are available for training detectors in latter steps.

object detection is feasible of different resolutions. On top of these multi-scale feature maps, detection can be formulated as:

$$Dets = F(p_n(\Phi_n, \mathcal{B}_n), p_{n-1}(\Phi_{n-1}, \mathcal{B}_{n-1}), \ldots, p_{n-k}(\Phi_{n-k}, \mathcal{B}_{n-k})), \quad n > k > 0, \tag{2}$$

$$p_n(\Phi_n, \mathcal{B}_n) = \{cls_n(\Phi_n, \mathcal{B}_n), regr_n(\Phi_n, \mathcal{B}_n)\}, \tag{3}$$

where $\mathcal{B}_n$ is the anchor boxes pre-defined in the nth layer's feature map cells, $p_n(.)$ is typically a convolutional predictor that translates the nth feature maps Φ_n into detection results. Generally, $p_n(.)$ contains two elements, $cls_n(.)$ which predicts the classification scores, and $regr_n(.)$ which predicts the scaling and offsets of the default anchor boxes associated with the nth layer and finally gets the regressed boxes. $F(.)$ is the function to gather all regressed boxes from all layers and output final detection results. For more details please refer to [12].

We can find that Eq. (2) plays the same role as RPN in Faster R-CNN, except that RPN applies the convolutional predictor $p_n(.)$ on the feature maps of the last layer for anchors of all scales (denoted as $\mathcal{B}$), which can be formulated as:

$$Proposals = p_n(\Phi_n, \mathcal{B}), \quad n > 0. \tag{4}$$

In two-stage methods, the region proposals from Eq. (4) are further processed by the ROI-pooling, and then fed into another detection sub-network for classification and regression, thus is more accurate but less computationally efficient than single-stage methods.

B. Asymptotic Localization Fitting

From the above analysis, it can be seen that the single-stage methods are suboptimal primarily because it is difficult to ask a single predictor $p_n(.)$ to perform perfectly on the

default anchor boxes uniformly paved on the feature maps. A reasonable solution is to stack a series of predictors $p_n^t(.)$ applied on coarse-to-fine anchor boxes $\mathcal{B}_n^t$, where t indicates the t_{th} step. In this case, Eq. 3 can be re-formulated as:

$$p_n(\Phi_n, \mathcal{B}_n^0) = p_n^T(p_n^{T-1}(\ldots(p_n^1(\Phi_n, \mathcal{B}_n^0))), \quad (5)$$

$$\mathcal{B}_n^t = regr_n^t(\Phi_n, \mathcal{B}_n^{t-1}), \quad (6)$$

where T is the number of total steps and $\mathcal{B}_n^0$ denotes the default anchor boxes paved on the n_{th} layer. In each step, the predictor $p_n^t(.)$ is optimized using the regressed anchor boxes $\mathcal{B}_n^{t-1}$ instead of the default anchor boxes. In other words, the progressively refined anchor boxes have higher IoU with the ground-truth boxes as shown in Fig. 4, thus the predictors in latter steps can be trained with a higher IoU threshold, which is helpful to produce more precise localization during inference. Another advantage of this strategy is that multiple classifiers trained with different IoU thresholds in all steps will score each anchor box in a 'multi-expert' manner, and thus if properly fused the score will be more confident than a single classifier. Similar findings are also suggested in the Cascade R-CNN [13], while the proposed method is still in the single-stage detection framework and can be trained in a fully convolutional manner. Given this design, the limitations of current single-stage detectors could be alleviated, resulting in a potential of surpassing the two-stage detectors in both accuracy and efficiency.

Fig. 2 gives two example images to demonstrate the effectiveness of the proposed ALF module. As can be seen from the first two rows in Fig. 2, the increasing IoU values indicate that the default anchor boxes even with a relatively lower overlap with the groundtruth boxes are able to approaching to the targets with progressive steps. In this way, more accurate localization is achieved during inference. In the last two rows in Fig. 2 (a), there are only 6 and 12 default anchor boxes in two example images, respectively, assigned as positive samples under the IoU threshold of 0.5 in the initial step. Later, this number increases progressively with more ALF steps, and the value of mean overlaps with the groundtruth is also going up. It indicates that the former predictor can hand over more anchor boxes with higher IoU to the latter one during training.

C. Convolutional Predictor Block (CPB)

The convolutional predictor defined in Eq. 3 also plays a significant role in final detection results. A toy example of the convolutional predictor block (CPB) adopted in this paper is pictorially illustrated in the left part of Fig. 3. Firstly, a detection head in the red rectangle is utilized to reduce the input feature maps into a fixed dimension (e.g. 512 in original RPN [9]). Secondly, on top of the evolved feature maps, two sibling 1x1 convolutional layers are independently appended for bounding box classification and regression. Finally, the predicted box offsets are applied on the anchor boxes to get the regressed proposals or obtain detection results directly with the predicted confidence scores.

As a starting point, we adopt the same detection head utilized in RPN [9], which is simply a 3x3 convolutional layer with 256 channels. However, the 3x3 convolutional layer

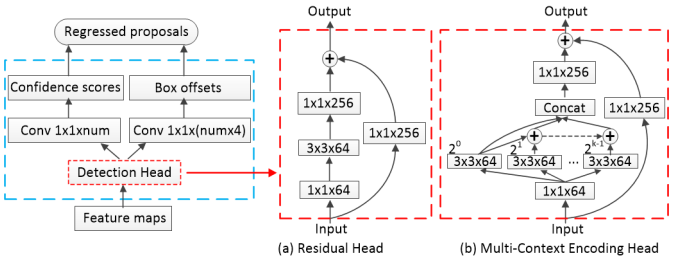


Fig. 3. Detailed architecture of the Convolutional Predictor Block (CPB). Each convolution layer is represented as the kernel size and the number of output channels. 'num' means the number of anchor boxes in each cell predefined in the corresponding feature maps. The 'Detection Head' in the red rectangle is simply a 3x3 convolutional layer in the original ALFNet [18]. (a) and (b) represents the residual block and the proposed multi-context encoding block, respectively. In (b), the intermediate 3x3 convolution layer is applied k times on k parallel paths, with dilation rates of 2^0, 2^1,…and 2^{k-1}, respectively.

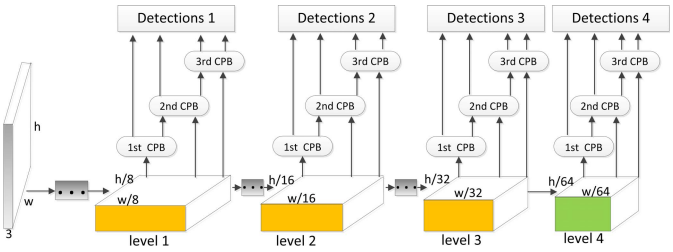


Fig. 4. The detection framework of the proposed ALFNet. It is constructed by four levels of feature maps for detecting objects with different sizes, where the first three blocks in yellow are from the backbone network, and the green one is an added convolutional layer to the end of the truncated backbone network.

contains a large number of parameters, making the CPB a much heavy detection head. To address this problem, inspired by the bottleneck design in ResNet [15], in this paper we propose a novel detection head to achieve a light-weight but more effective CPB. As the core of the ResNet [15], the residual bottleneck block depicted in Fig. 3 (a) helps ResNet [15] achieve better accuracy in image classification than heavier networks with standard convolutional layers, while with less parameters. Based on the above observations, the proposed detection head is on top of residual learning, with the main idea of exploring multi-scale context for better detection accuracy. Fig. 3 (b) gives the details of the proposed multi-context encoding head. Firstly, a 1x1 convolutional layer is applied to project input feature maps onto a low-dimensional space which shares the similar principle to the residual block in Fig. 3 (a). Secondly, a set of dilated convolutions re-samples these low-dimensional feature maps with kernel size 3×3 and dilation rates of 2^0, 2^1,…and 2^{k-1}, respectively. The kernels of different dilation rates result in a large range of effective receptive field, thus encoding multi-scale spatial context in an efficient way. Thirdly, the features maps obtained using kernels of different dilation rates are hierarchically integrated with element-wise summation, followed by concatenation in the channel dimension. Finally, a skip connection layer is utilized to transit the input to the output with the combination of element-wise summation, to improve the gradient flow inside

the network as suggested in [15]. To sum up, the proposed multi-context encoding head not only inherits the advantage of residual learning, but also is capable of encoding multi-scale context, thus is beneficial for more accurate localization which will be demonstrated in our experiments (see Sec. IV-B).

D. Overall Framework

In this section we will present details of the proposed ALFNet pedestrian detection pipeline.

The details of our detection network architecture is pictorially illustrated in Fig. 4. The proposed method is based on a fully-convolutional network that produces a set of bounding boxes and confidence scores indicating pedestrian instances or not. The base network layers are truncated from a standard network used for image classification (e.g. ResNet-50 [15] or MobileNet [59]). Taking ResNet-50 as an example, we firstly emanate branches from feature maps of the last layers of *stage 3, 4 and 5* (denoted as Φ_3, Φ_4 and Φ_5, the yellow blocks in Fig. 4) and attach an additional convolutional layer (denoted as 'conv_p6') at the end to produce Φ_6, generating an auxiliary branch (the green block in Fig. 4). Besides, a buffer convolutional layer (denoted as 'conv_p3') is introduced on the branch from Φ_3 because this branch is close to the lower layers of the backbone network as stated in [37]. For the other two detection heads defined in the right part of Fig. 3, 'conv_p3' and 'conv_p6' are both replaced by a residual block illustrated in Fig. 3 (a). Detection is performed on $\{\Phi_3, \Phi_4, \Phi_5, \Phi_6\}$, with sizes downsampled by 8, 16, 32, 64 w.r.t. the input image, respectively. For proposal generation, anchor boxes with width of $\{(16, 24), (32, 48), (64, 80), (128, 160)\}$ pixels and a single aspect ratio of 0.41, are assigned to each level of feature maps, respectively. Then, we append the Convolutional Predictor Block (CPB) illustrated in Fig. 3 and Sec. III-C with several stacked steps for bounding box classification and regression.

E. Training and Inference

1) Training: Anchor boxes are assigned as positives S_+ if the IoUs with any ground truth are above a threshold u_h, and negatives S_- if the IoUs lower than a threshold u_l. Those anchors with IoU in $[u_l, u_h)$ are ignored during training. We assign different IoU threshold sets $\{u_l, u_h\}$ for progressive steps which will be discussed in our experiments (see Sec. IV-B).

At each step t, the convolutional predictor is optimized by a multi-task loss function combining two objectives:

$$L = l_{cls} + \lambda[y = 1]l_{loc}, \tag{7}$$

where the regression loss l_{loc} is the same smooth L1 loss adopted in Faster R-CNN [9], l_{cls} is cross-entropy loss for binary classification, and λ is a trade-off parameter. Inspired by [33], we also append the focal weight in classification loss l_{cls} to combat the positive-negative imbalance. The l_{cls} is formulated as:

$$l_{cls} = -\alpha \sum_{i \in S_+} (1 - p_i)^\gamma \log(p_i) - (1-\alpha) \sum_{i \in S_-} p_i^\gamma \log(1 - p_i), \tag{8}$$

where p_i is the positive probability of sample i, α and γ are the focusing parameters, experimentally set as $\alpha = 0.25$ and $\gamma = 2$ suggested in [33]. In this way, the loss contribution of easy samples are down-weighted.

To increase the diversity of the training data, each image is augmented by the following options: after random color distortion and horizontal image flip with a probability of 0.5, we firstly crop a patch with the size of [0.3, 1] of the original image, then the patch is resized such that the shorter side has N pixels ($N = 640$ for CityPersons, and $N = 336$ for Caltech), while keeping the aspect ratio of the image.

2) Inference: ALFNet simply involves feeding forward an image through the network. For each level, we get the regressed anchor boxes from the final convolutional predictor and hybrid confidence scores from all convolutional predictors. We firstly filter out boxes with scores lower than 0.01, then all remaining boxes are merged with the Non-Maximum Suppression (NMS) with a threshold of 0.5.

IV. Experiments and Analysis

A. Experiment Settings

1) Datasets: The performance of ALFNetV2 is evaluated on the CityPersons [11] and Caltech [17] benchmarks. The CityPersons dataset is a newly published large-scale pedestrian detection dataset built upon the semantic segmentation dataset (i.e. Cityscapes [60]). It has 2975 images and approximately 20000 annotated pedestrian instances in the training subset. The proposed model is trained on this training subset and evaluated on the validation subset. For Caltech, our model is trained and test with the new annotations provided by [61]. We use the 10x set (42782 images) for training and the standard test subset (4024 images) for evaluation.

The evaluation metric follows the standard Caltech evaluation [17]: log-average Miss Rate over False Positive Per Image (FPPI) range of $[10^{-2}, 10^0]$ (denoted as MR^{-2}). Test are only applied on the original image size without enlarging for speed consideration.

2) Training Details: Our method is implemented in the Keras [62] platform, with 2 GTX 1080Ti GPUs for training. A mini-batch contains 10 images per GPU. The Adam solver is applied. For CityPersons, the backbone network is pretrained on ImageNet [63] and all added layers are randomly initialized with the xavier method. We totally train the network for $240k$ iterations, with the initial learning rate set as 0.0001 and decreased by a factor of 10 after $160k$ iterations. For Caltech, the model is initialized from CityPersons as done in [11] and [44] and totally trained for $140k$ iterations with the learning rate of 0.00001. The backbone network is ResNet-50 [15] unless otherwise stated.

B. Ablation Experiments

In this section, we conduct the following ablation studies to demonstrate the effectiveness of the proposed method. All experimental results are reported on the CityPersons validation dataset unless otherwise stated.

TABLE I

THE STEPWISE LOCALIZATION IMPROVEMENT EVALUATED UNDER IOU THRESHOLD OF 0.5 AND 0.75. C_i REPRESENTS THE CONFIDENCE SCORES FROM STEP i AND B_j MEANS THE BOUNDING BOX LOCATIONS FROM STEP j. MR^{-2} ON THE REASONABLE SUBSET IS REPORTED

IoU	C_1B_1	C_1B_2	C_2B_2	$(C_1+C_2)B_2$	$(C_1*C_2)B_2$	Gain
0.5	13.46	13.17	12.64	12.03	**12.01**	+1.45
0.75	46.83	45.00	**34.90**	36.49	36.49	+11.93

1) ALF Improvement: For clarity, we trained a detector with two steps. Table I summarizes the stepwise localization performance, where C_iB_j represents the detection results obtained by the confidence scores on step i and bounding box locations on step j. As can be seen from Table I, when evaluated with different IoU thresholds (e.g. 0.5, 0.75), the second convolutional predictor consistently performs better than the first one. With the same confidence scores C_1, the improvement from C_1B_2 to C_1B_1 indicates the second regressor is better than the first one. On the other hand, with the same bounding box locations B_2, the improvement from C_2B_2 to C_1B_2 indicates the second classifier is better than the first one.

We also combine the two confidence scores by summation or multiplication, which is denoted as (C_1+C_2) and (C_1*C_2). For the IoU threshold of 0.5, this kind of score fusion is considerably better than both C_1 and C_2. Yet interestingly, under a stricter IoU threshold of 0.75, both the two hybrid confidence scores underperform the second confidence score C_2, which reasonably indicates that the second classifier is more discriminative between groundtruth and many "close but not accurate" false positives. It is worth noting that when we increase the IoU threshold from 0.5 to a stricter 0.75, the largest improvement increases by a large margin (from 1.45% to 11.93%), demonstrating the high-quality localization performance of the proposed ALFNet.

To further demonstrate the effectiveness of the proposed method, Fig. 5 depicts the distribution of anchor boxes over the IoU range of [0.5, 1]. The total number of matched anchor boxes increases by a large margin (from 16351 up to 100571). Meanwhile, the percentage of matched anchor boxes in higher IoU intervals is increasing stably. In other words, anchor boxes with different IoU values are relatively well-distributed with the progressive steps.

2) IoU Threshold for Training: As shown in Fig. 5, the number of matched anchor boxes increases drastically with latter steps, and the gap among different IoU thresholds is narrowing down. A similar finding is also observed in the Cascade R-CNN [13], with a single threshold instead of dual thresholds here. This inspires us to also study how the IoU threshold for training affects the final detection performance. Experimentally, the $\{u_l, u_h\}$ for the first step should not be higher than that for the second step, because more anchors with higher quality are assigned as positives after the first step (shown in Fig. 5). Results in Table II shows that training predictors of two steps with the 'increasing' IoU thresholds is better than that with the same IoU thresholds, which indicates that optimizing the later predictor more strictly with higher-quality positive anchors is vitally important for better performance. We choose {0.3, 0.5} and {0.5, 0.7} for two steps in the following experiments, which achieves the lowest MR^{-2} in both of the two evaluated settings (IoU=0.5, 0.75).

TABLE II

COMPARISON OF TRAINING THE TWO-STEP ALFNET WITH DIFFERENT IOU THRESHOLD SETS. $\{u_l, u_h\}$ REPRESENTS THE IOU THRESHOLD TO ASSIGN POSITIVES AND NEGATIVES DEFINED IN SECTION. III-D. BOLD AND ITALIC INDICATE THE BEST AND SECOND BEST RESULTS

Training IoU thresholds		$MR^{-2}(\%)$	
step 1	step 2	IoU=0.5	IoU=0.75
{0.3, 0.5}	{0.3, 0.5}	13.75	44.27
	{0.4, 0.6}	13.31	39.30
	{0.5, 0.7}	**12.01**	*36.49*
{0.4, 0.6}	{0.4, 0.6}	13.60	42.31
	{0.5, 0.7}	*12.80*	**36.43**
{0.5, 0.7}	{0.5, 0.7}	13.72	38.20

TABLE III

COMPARISON OF ALFNET WITH VARIOUS STEPS EVALUATED IN TERMS OF MR^{-2}. TEST TIME IS EVALUATED ON THE ORIGINAL IMAGE SIZE (1024X2048 ON CITYPERSONS)

Method	# Steps	Test step	Test time	$MR^{-2}(\%)$	
				IoU=0.5	IoU=0.75
ALFNet-1s	1	1	0.26s/img	16.01	48.95
ALFNet-2s	2	1	0.26s/img	13.17	45.00
	2	2	0.27s/img	**12.01**	**36.49**
ALFNet-3s	3	1	0.26s/img	14.53	46.70
	3	2	0.27s/img	12.67	37.75
	3	3	0.28s/img	12.88	39.31

TABLE IV

COMPARISON OF ALFNET WITH VARIOUS STEPS EVALUATED WITH F-MEASURE. # TP AND # FP DENOTE THE NUMBER OF TRUE POSITIVES AND FALSE POSITIVES

Method	Test step	mIoU	IoU=0.5			IoU=0.75		
			# TP	# FP	F-mea.	# TP	# FP	F-mea.
ALFNet-1s	1	0.49	**2404**	13396	0.263	1786	14014	0.195
ALFNet-2s	1	0.55	2393	9638	0.330	**1816**	10215	0.250
	2	0.76	2198	1447	0.717	1747	1898	0.570
ALFNet-3s	1	0.57	2361	7760	0.375	1791	8330	0.284
	2	0.76	2180	1352	0.725	1734	1798	0.576
	3	0.80	2079	**768**	**0.780**	1694	**1153**	**0.635**

3) Number of Stacked Steps: The proposed ALF module is helpful to achieve better detection performance, but we have not yet studied how many stacked steps are enough to obtain a speed-accuracy trade-off. We train our ALFNet up to three steps when the accuracy is saturated. Table III compares the three variants of our ALFNet with 1, 2 and 3 steps, denoted as ALFNet-1s, ALFNet-2s and ALFNet-3s. Experimentally, the ALFNet-3s is trained with IoU thresholds {0.3, 0.5}, {0.4, 0.65} and {0.5, 0.75}). By adding a second step, ALFNet-2s significantly surpasses ALFNet-1s by a large margin (12.01% VS. 16.01%). It is worth noting that the results from the first step of ALFNet-2s and ALFNet-3s are substantially better than ALFNet-1s with the same computational burden, which indicates that multi-step training is also beneficial for optimizing the former step.

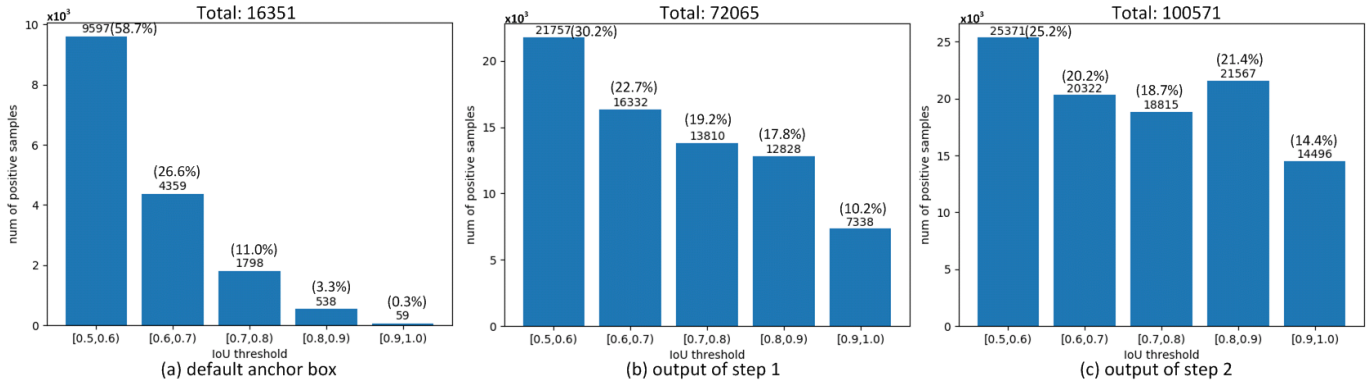


Fig. 5. The distribution of anchor boxes over the IoU range of [0.5, 1]. It depicts the number of anchor boxes matched with the ground-truth boxes w.r.t. different IoU thresholds. (a), (b) and (c) represent the distribution of default anchor boxes, refined anchor boxes after the first and second step, respectively. The total number of boxes with IoU above 0.5 is presented in the heads of the three sub-figures. The numbers and percentages of each IoU threshold range are annotated on the head of the corresponding bar.

TABLE V

COMPARISON OF DIFFERENT BACKBONE NETWORK WITH OUR ALF DESIGN

Backbone	ALF	# Parameters	Test time	$MR^{-2}(\%)$	
				IoU=0.5	IoU=0.75
ResNet-50		39.5M	0.26s/img	16.01	48.94
	✓	48.4M	0.27s/img	12.01	36.49
MobileNetV1		12.1M	0.17s/img	18.88	56.26
	✓	17.4M	0.18s/img	15.45	47.42

TABLE VI

COMPARISON BETWEEN SSD AND THE PROPOSED ALFNET ON BOTH THE CALTECH AND THE CITYPERSONS DATASETS. THE MISS RATES ON THE REASONABLE SETTING ARE REPORTED

Method	# Parameters	$MR^{-2}(\%)$		Improvement(%)	
		CityPersons	Caltech	CityPersons	Caltech
SSD*	39.5M	16.01	10.46	4.00	4.36
ALFNet	48.4M	12.01	6.10		

* indicates that this variant of SSD has the same backbone with the proposed ALFNet.

TABLE VII

COMPARISONS OF DIFFERENT DETECTION HEADS DISCUSSED IN SEC. III-C. THE BACKBONE NETWORK IS RESNET-50

Det. Head	ALF	# Parameters	Test time	$MR^{-2}(\%)$	
				IoU=0.5	IoU=0.75
Std. Conv.		39.5M	0.26s/img	16.01	48.94
	✓	48.4M	0.27s/img	12.01	36.49
Res. Block		26.3M	0.30s/img	13.93	42.88
	✓	27.8M	0.31s/img	**11.46**	35.67
MCE Block		26.7M	0.31s/img	15.18	42.63
	✓	28.6M	0.32s/img	11.78	**32.14**

TABLE VIII

COMPARISON ON DIFFERENT COMBINATIONS OF DILATION RATES IN THE MCE BLOCK. THE MISS RATES ON THE REASONABLE SETTING ARE REPORTED

Det. Head	Dilation Rates				$MR^{-2}(\%)$
	2^0	2^1	2^2	2^3	
MCE Block	✓	✓			12.11
	✓	✓	✓		**11.78**
	✓	✓	✓	✓	11.80
		✓	✓	✓	11.95

From the results shown in Table III, it appears that the addition of the 3rd step can not provide performance gain in terms of MR^{-2}. Yet when taking a deep look at the detection results of this three variants of ALFNet, the detection performance based on the metric of F-measure is further evaluated, as shown in Table IV. In this case, ALFNet-3s tested on the 3rd step performs the best under the IoU threshold of both 0.5 and 0.75. It substantially outperforms ALFNet-1s and achieves a 6.3% performance gain from ALFNet-2s under the IoU of 0.5, and 6.5% with IoU=0.75. It can also be observed that the number of false positives decreases progressively with increasing steps, which is pictorially illustrated in Fig. 6. Besides, as shown in Table IV, the average mean IoU of the detection results matched with the groundtruth is increasing, further demonstrating the improved detection quality. However, the improvement of step 3 over step 2 is saturating, compared to the large gap of step 2 over step 1. Therefore, considering the speed-accuracy trade-off, we choose ALFNet-2s in the following experiments.

4) Different Backbone Network: Large backbone network like ResNet-50 is *strong* in feature representation. To further demonstrate the improvement from the ALF module, a light-weight network like MobileNetV1 [59] is chosen as the backbone and the results are shown in Table V. Notably, the *weaker* MobileNetV1 equipped with the proposed ALF module is able to beat the *strong* ResNet-50 without ALF (15.45% VS. 16.01% in MR^{-2} under IoU of 0.5). Note that the proposed method without ALF is essentially a SSD detector, thus Table V also gives a direct comparison between SSD and the proposed ALFNet built on different backbone networks. It can seen that compared to SSD, the proposed method has merely 5.3M and 8.9M parameters overhead based on the backbone of ResNet-50 and MobileNetV1, respectively, and the additional time consumption is 0.01s per image of 1024x2048 pixels in the CityPersons dataset. Furthermore, we also conduct experiments on the Caltech dataset and the direct comparisons between SSD and the proposed ALFNet

Fig. 6. Examples of detection results of ALFNet-3s. Red and green rectangles represent groundtruth and detection bounding boxes, respectively. It can be seen that the number of false positives decreases progressively with increasing steps, which indicates that more steps are beneficial for higher detection accuracy.

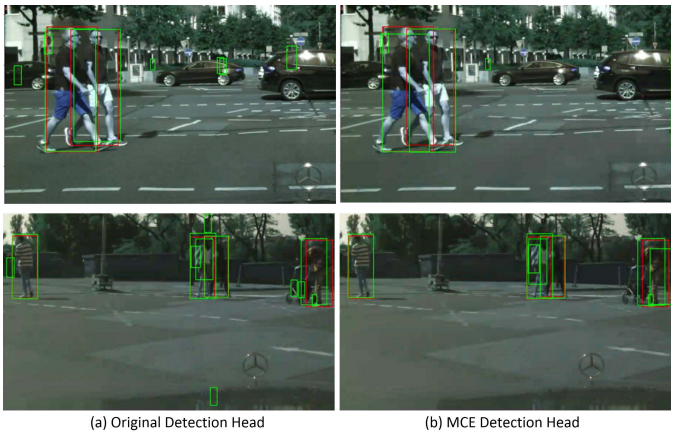


Fig. 7. Examples of detection results of ALFNet with (a) the Original Detection Head and (b) the MCE Detection Head. Red and green rectangles represent groundtruth and detection bounding boxes, respectively. These detections with confidence scores larger than 0.01 are plotted. It can be seen that the number of false positives in (b) is less than that in (a).

are reported in Table VI. It can be seen that the proposed ALFNet consistently outperforms SSD by 4.00% and 4.36% on CityPersons and Caltech, respectively.

5) Different Detection Head: As shown in Table V, when the ALF module is utilized, the detection accuracy is significantly boosted but with the cost of much more memory burdens. With the standard convolutional layer used in the detection head illustrated in Fig. 3, the ALF module brings additional 8.9M and 5.3M parameters based on the ResNet-50 and MobileNet backbone, respectively, resulting in a relatively expensive network. Hence, we implemented another two variants of the detection heads illustrated in Fig. 3, which are denoted as **Res. Block** and **MCE Block**. Comparisons of different detection heads are reported in Table VII. Note that **Std. Conv** denotes the original ALFNet and **MCE Block** denotes the proposed ALFNetV2. Experimentally, k in the **MCE Block** (Fig.3 (b)) is set as 3. The parameter counts of the **Res. Block** and the **MCE Block** are approximately 1/5 of that of a standard 3x3 convolutional layer here. With the same backbone ResNet-50 containing 23.6M parameters, due to the utilization of the **Res. Block** and **MCE Block** in the detection head, it can be seen from Table VII that the parameter counts of the non-ALF variants decrease from 39.5M to 26.3M and 26.7M, respectively. Impressively, the residual block significantly reduce the number of parameters of the standard convolution counterpart while still achieves a better detection accuracy no matter whether the ALF module is utilized (13.93% and 11.46% VS. 16.01% and 12.01% in MR^{-2} under the IoU threshold of 0.5). In this case, the ALF module only introduces 1.5M parameters but still brings a substantial performance gain, which indicates that 1) the improvement is from the capability of the ALF module, other than increasing model parameters; 2) the residual learning is also beneficial for better detection accuracy. It is worth noting that when the multi-scale context is utilized, the **MCE Block** achieves the best performance under a stricter IoU threshold of 0.75, and is comparative to the **Res. Block** under the IoU threshold of 0.5, which indicates that the multi-scale context encoding is helpful for better localization accuracy. As the core of multi-scale context encoding, the value of k plays a significant role in final detection performance. To further analyse the performance using different combinations of dilation rates in the MCE Block, we conduct an ablative experiment and the results are given in Table VIII. It can be seen from Table VIII that the combination of $\{2^0, 2^1\}$ $(k = 2)$ performs poorly which indicates that insufficient context are encoded, and the combination of $\{2^0, 2^1, 2^2, 2^3\}$ $(k = 4)$ performs on par with the combination of $\{2^0, 2^1, 2^2\}$ $(k = 3)$, which indicates that $k = 3$ is suitable to encode sufficient context information. To answer whether larger dilation rate is helpful, we also experiment with the combination of $\{2^1, 2^2, 2^3\}$, which performs relatively poorly than $\{2^0, 2^1, 2^2\}$ as shown in the last row of Table VIII. It indicates that proper dilation rates are critical for better detection performance. To further

TABLE IX

COMPARISON WITH THE STATE-OF-THE-ART ON THE CITYPERSONS [11]. THE ORIGINAL IMAGE SIZE IS 1024X2048 PIXELS. 'SCALE' INDICATES THE UP-SAMPLE RATE DURING TEST. RED AND GREED INDICATE THE BEST AND SECOND BEST PERFORMANCE

Method	Backbone	+RepGT	+RepBox	+Seg.	Scale	Reasonable	Heavy	Partial	Bare
Faster R-CNN[11]	VGG16				x1	15.4	-	-	-
				✓	x1	14.8	-	-	-
					x1.3	12.8	-	-	-
RepLoss[44]	ResNet-50				x1	14.6	60.6	18.6	7.9
		✓			x1	13.7	57.5	17.3	7.2
			✓		x1	13.7	59.1	17.2	7.8
		✓	✓		x1	13.2	56.9	16.8	7.6
		✓	✓		x1.3	11.6	55.3	14.8	7.0
OR-CNN[45]	VGG16				x1	12.8	55.7	15.3	6.7
					x1.3	11.0	51.3	13.7	5.9
Bi-box[46]	VGG16				x1.3	11.2	-	-	-
ALFNet	ResNet-50				x1	12.0	51.9	11.4	8.4
ALFNet_Res	ResNet-50				x1	11.5	50.0	11.4	7.6
ALFNetV2	ResNet-50				x1	11.8	48.9	10.8	8.1

TABLE X

COMPARISONS OF RUNNING TIME ON CALTECH. THE TIME OF LDCF, CCF, COMPACT-DEEP AND RPN+BF ARE REPORTED IN [10], AND THAT OF SA-FASTRCNN AND F-DNN ARE REPORTED IN [40]. MR^{-2} IS BASED ON THE NEW ANNOTATIONS [61]. THE ORIGINAL IMAGE SIZE ON CALTECH IS 480X640. 'SCALE' INDICATES THE UP-SAMPLE RATE DURING TEST. RED AND GREED INDICATE THE BEST AND SECOND BEST PERFORMANCE

Method	Hardware	Scale	Test time	$MR^{-2}(\%)$	
				IoU=0.5	IoU=0.75
LDCF [4]	CPU	x1	0.6 s/img	23.6	72.2
CCF [64]	Titan Z GPU	x1	13 s/img	23.8	97.4
CompACT-Deep [65]	Tesla K40 GPU	x1	0.5 s/img	9.2	59.0
RPN+BF [10]	Tesla K40 GPU	x1.5	0.5 s/img	7.3	57.8
SA-FastRCNN [38]	Titan X GPU	x1.7	0.59 s/img	7.4	55.5
F-DNN [40]	Titan X GPU	x1	0.16 s/img	6.9	59.8
ALFNet	GTX 1080Ti GPU	x1	0.05 s/img	6.1	22.5
ALFNet+City	GTX 1080Ti GPU	x1	0.05 s/img	4.5	18.6
ALFNet_Res+City	GTX 1080Ti GPU	x1	0.06 s/img	4.4	17.6
ALFNetV2+City	GTX 1080Ti GPU	x1	0.06 s/img	4.3	15.7

demonstrate the effectiveness of the MCE Block, we also give two examples of detection results of ALFNet with the original detection head and the MCE detection head in Fig. 7, where red and green rectangles represent groundtruth and detection bounding boxes, respectively. It can be seen that several false positives are suppressed in Fig. 7 (b), since the utilization of the multi-scale context encoding is helpful to reduce the false positives in the background.

To sum up, we also achieved several similar observations with Cascade R-CNN [13]. Firstly, progressively training can significantly improve the detection accuracy especially under a stricter IoU threshold, which is demonstrated in Table I Secondly, training with increasing IoU threshold is beneficial for more accurate localization, which is demonstrated in Table II. Thirdly, the total number and the percentage of matched anchor boxes with higher overlaps are increasing with progressively steps, which is demonstrated in Fig. 5. Finally, limited steps (2 in ours and 3 in Cascade R-CNN) are enough to achieve saturated detection accuracy, which is demonstrated in Table III.

C. Comparison to the State of the Arts

In this section, we report experimental results compared with the state of the arts on the CityPersons [11] and Caltech [17] benchmark. For the convenience of

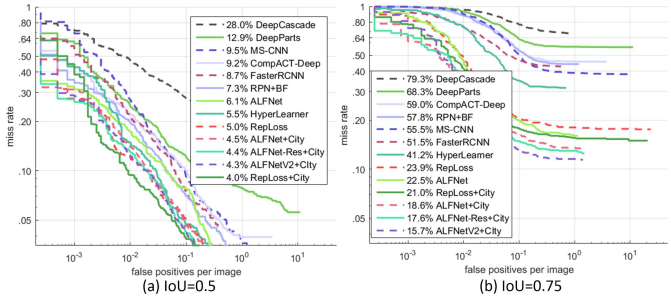


Fig. 8. Comparisons of the state-of-the-arts on Caltech (reasonable subset) under the IoU threshold of (a) 0.5 and (b) 0.75.

distinction, we denoted the ALFNet with the residual head as **ALFNet_Res** and the one with the multi-context-encoding head as **ALFNetV2**.

1) CityPersons [11]: Table IX shows the comparisons to the previous state of the arts on CityPersons. Note that it is a common practice to up-sample the image to achieve a better detection accuracy as demonstrated in Table IX, but with the cost of more computational expense. We only test on the original image size as pedestrian detection is more critical on both accuracy and efficiency. For completeness, Table IX also gives the results of other methods up-sampling

the image size by 1.3 during test. Besides the reasonable subset, following [44], we also test the proposed method on three subsets with different occlusion levels. On the Reasonable subset, without any additional supervision like semantic labels (as done in [11]) or auxiliary regression loss (as done in [44] and [45]), the proposed **ALFNet** already provides an improvement of 1.2% MR^{-2} from the closest competitor RepLoss [44] test on the original image without up-sampling. Note that RepLoss [44] and OR-CNN [45] are specifically designed for the occlusion problem, however, the proposed **ALFNet** achieves comparable or even better performance in terms of different levels of occlusions, demonstrating the self-contained ability of the proposed method to handle occlusion issues in crowded scenes. This is probably because in the latter ALF steps, more positive samples are recalled for training, including occluded samples. On the other hand, harder negatives are mined in the latter steps, resulting in a more discriminant predictor. When the residual block is utilized in the detection head, the **ALFNet_Res** achieves the best performance (11.5% MR^{-2}) among the state-of-the-art results without up-sampling on the reasonable set and outperforms **ALFNet** on all occlusion levels. Further, the multi-context encoding helps the proposed **ALFNetV2** achieve the best performance on the 'Heavy' and 'Partial' sets even without up-sampling, beating RepLoss [44] specifically targeted at occlusions with a large margin (8.0% and 6.0% MR^{-2} on the 'Heavy' and 'Partial' sets, respectively), which demonstrates the effectiveness of the proposed method to encode multi-scale context for occluded pedestrian detection in crowded scenes. Note that OR-CNN [45] and Bi-box [46] are also specifically targeted at the occlusion problem in pedestrian detection, the proposed **ALFNetV2** performs substantially better on 'Heavy' and 'Partial' occlusions.

2) Caltech [17]: We also test the proposed ALFNetV2 on Caltech and the comparisons with the state of the arts on this benchmark are shown in Fig. 8 and Table X, where '+City' means the weights of models are initialized from CityPersons before training. The proposed **ALFNetV2** achieves MR^{-2} of 4.3% under the IoU threshold of 0.5, which is comparable to the best competitor (4.0% of RepLoss [44]). However, in the case of a stricter IoU threshold of 0.75, the proposed method is the first one to achieve the MR^{-2} below 20.0%, outperforming all previous state-of-the-arts. Without multi-context encoding, the **ALFNet+City** has already performed better than RepLoss+City with an improvement of 2.4% MR^{-2}. It indicates that the proposed method has a substantially better localization accuracy. Finally, when the multi-scale context is utilized, the **ALFNetV2+City** achieves the best performance of 15.7% MR^{-2} under the IoU threshold of 0.75, outperforming the **ALFNet+City** by approximately 3.0% MR^{-2}, which further demonstrates that the multi-scale context encoding is beneficial when precise localization is required.

Table X reports the running time on Caletch, ALFNet significantly outperforms the competitors on both speed and accuracy. The proposed ALFNet cost merely one third of the running time of the fastest competitor, F-DNN [40], but achieves a substantial performance gain without the input upsampling or mutli-scale testing. Thanks to the ALF module, our method avoids the time-consuming proposal-wise feature extraction (ROIpooling), instead, it refines the default anchors step by step, thus achieves a better speed-accuracy trade-off.

V. Conclusions

In this paper, we present a simple but effective single-stage pedestrian detector, achieving competitive accuracy while performing faster than the state-of-the-art methods. On top of a backbone network, an asymptotic localization fitting module is proposed to refine anchor boxes step by step into final detection results. This novel design is flexible and independent of any backbone network, without being limited by the single-stage detection framework. Therefore, it is also interesting to incorporate the proposed ALF module with other single-stage detectors like YOLO [28], [29] and FPN [22], [33], which will be studied in future.

References

[1] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," in *Proc. Brit. Mach. Vis. Conf.*, 2009, pp. 1–11.

[2] R. Benenson, M. Omran, J. Hosang, and B. Schiele, "Ten years of pedestrian detection, what have we learned?" in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 613–627.

[3] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1532–1545, Jan. 2014.

[4] W. Nam, P. Dollár, and J. H. Han, "Local decorrelation for improved pedestrian detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 424–432.

[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[6] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2553–2561.

[7] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. Lecun, "OverFeat: Integrated recognition, localization and detection using convolutional networks," in *Proc. Int. Conf. Learn. Represent.*, 2014, pp. 1–16.

[8] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1440–1448.

[9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[10] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster R-CNN doing well for pedestrian detection?" in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 443–457.

[11] S. Zhang, R. Benenson, and B. Schiele, "CityPersons: A diverse dataset for pedestrian detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 3213–3221.

[12] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 21–37.

[13] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6154–6162.

[14] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Single-shot refinement neural network for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4203–4212.

[15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[16] F. Yu, V. Koltun, and T. Funkhouser, "Dilated residual networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, vol. 1, Jul. 2017, pp. 472–480.

[17] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, Apr. 2012.

[18] W. Liu, S. Liao, W. Hu, X. Liang, and X. Chen, "Learning efficient single-stage pedestrian detectors by asymptotic localization fitting," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 618–634.

[19] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 379–387.

[20] T. Kong, A. Yao, Y. Chen, and F. Sun, "HyperNet: Towards accurate region proposal generation and joint object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 845–853.

[21] H. Lee, S. Eum, and H. Kwon, "ME R-CNN: Multi-expert region-based CNN for object detection," 2017, *arXiv:1704.01069*. [Online]. Available: https://arxiv.org/abs/1704.01069

[22] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2117–2125.

[23] A. Shrivastava, A. Gupta, and R. Girshick, "Training region-based object detectors with online hard example mining," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 761–769.

[24] X. Wang, A. Shrivastava, and A. Gupta, "A-fast-RCNN: Hard positive generation via adversary for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2606–2615.

[25] S. Bell, C. Lawrence Zitnick, K. Bala, and R. Girshick, "Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2874–2883.

[26] S. Gidaris and N. Komodakis, "Object detection via a multi-region and semantic segmentation-aware CNN model," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1134–1142.

[27] A. Shrivastava and A. Gupta, "Contextual priming and feedback for faster R-CNN," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 330–348.

[28] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.

[29] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 7263–7271.

[30] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, "DSSD: Deconvolutional single shot detector," 2017, *arXiv:1701.06659*. [Online]. Available: https://arxiv.org/abs/1701.06659

[31] Z. Shen, Z. Liu, J. Li, Y.-G. Jiang, Y. Chen, and X. Xue, "DSOD: Learning deeply supervised object detectors from scratch," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Jul. 2017, vol. 3, no. 6, pp. 1–7.

[32] T. Kong, F. Sun, A. Yao, H. Liu, M. Lu, and Y. Chen, "RON: Reverse connection with objectness prior networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 5936–5944.

[33] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2980–2988.

[34] J. Hosang, M. Omran, R. Benenson, and B. Schiele, "Taking a deeper look at pedestrians," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 4073–4082.

[35] Y. Tian, P. Luo, X. Wang, and X. Tang, "Pedestrian detection aided by deep learning semantic tasks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5079–5087.

[36] Y. Tian, P. Luo, X. Wang, and X. Tang, "Deep learning strong parts for pedestrian detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1904–1912.

[37] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 354–370.

[38] J. Li, X. Liang, S. Shen, T. Xu, J. Feng, and S. Yan, "Scale-aware fast R-CNN for pedestrian detection," *IEEE Trans. Multimedia*, vol. 20, no. 4, pp. 985–996, Oct. 2017.

[39] G. Brazil, X. Yin, and X. Liu, "Illuminating pedestrians via simultaneous detection & segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Jun. 2017, pp. 4950–4959.

[40] X. Du, M. El-Khamy, J. Lee, and L. Davis, "Fused DNN: A deep neural network fusion approach to fast and robust pedestrian detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2017, pp. 953–961.

[41] J. Mao, T. Xiao, Y. Jiang, and Z. Cao, "What can help pedestrian detection?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 3127–3136.

[42] D. Xu, W. Ouyang, E. Ricci, X. Wang, and N. Sebe, "Learning cross-modal deep representations for robust pedestrian detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4236–4244.

[43] T. Song, L. Sun, D. Xie, H. Sun, and S. Pu, "Small-scale pedestrian detection based on topological line localization and temporal feature aggregation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 536–551.

[44] X. Wang, T. Xiao, Y. Jiang, S. Shao, J. Sun, and C. Shen, "Repulsion loss: Detecting pedestrians in a crowd," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7774–7783.

[45] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Occlusion-aware R-CNN: Detecting pedestrians in a crowd," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 637–653.

[46] C. Zhou and J. Yuan, "Bi-box regression for pedestrian detection and occlusion estimation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 135–151.

[47] C. Lin, J. Lu, G. Wang, and J. Zhou, "Graininess-aware deep feature learning for pedestrian detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 745–761.

[48] S. Zhang, J. Yang, and B. Schiele, "Occluded pedestrian detection through guided attention in CNNs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6995–7003.

[49] J. Noh, S. Lee, B. Kim, and G. Kim, "Improving occlusion and hard negative handling for single-stage pedestrian detectors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 966–974.

[50] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Jul. 2017, pp. 2980–2988.

[51] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1925–1934.

[52] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2881–2890.

[53] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8759–8768.

[54] H. Zhang *et al.*, "Context encoding for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7151–7160.

[55] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–14.

[56] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. Int. Conf. Learn. Represent.*, 2016, pp. 1–13.

[57] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2017.

[58] S. Mehta, M. Rastegari, A. Caspi, L. Shapiro, and H. Hajishirzi, "ESPNet: Efficient spatial pyramid of dilated convolutions for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 552–568.

[59] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: https://arxiv.org/abs/1704.04861

[60] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 3213–3223.

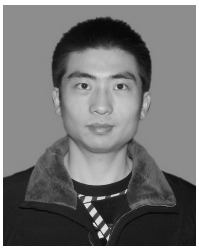
[61] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele, "How far are we from solving pedestrian detection?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1259–1267.

[62] F. Chollet. (2015). *Keras*. [Online]. Available: https://github.com/fchollet/keras

[63] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and F.-F. Li, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[64] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Convolutional channel features," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 82–90.

[65] Z. Cai, M. Saberian, and N. Vasconcelos, "Learning complexity-aware cascades for deep pedestrian detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 3361–3369.



Wei Liu received the B.S. and M.S. degrees from the Aviation University of Air Force, Changchun, China, in 2013 and 2015, respectively. He is currently pursuing the Ph.D. degree with the College of Electronic Science, National University of Defense Technology, Changsha, China, under the supervision of Prof. Hu. Since 2016, he has been an Intern with the Center for Biometrics and Security Research & National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, under the supervision of Prof. Liao. His research interests include object detection, visual object tracking, video surveillance, and generative adversarial networks.

Shengcai Liao (SM'16) received the B.S. degree in mathematics and applied mathematics from Sun Yat-sen University in 2005 and the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences (CASIA), in 2010. He was a Postdoctoral Fellow with the Department of Computer Science and Engineering, Michigan State University, from 2010 to 2012. Previously, he was an Associate Professor with CASIA. He is currently a Lead Scientist with the Inception Institute of Artificial Intelligence (IIAI), Abu Dhabi, UAE. He has published over 100 articles, with over 9000 citations according to Google Scholar. His research interests include computer vision and pattern recognition, with a focus on image and video analysis, particularly face recognition, object detection, person re-identification, and video surveillance. He was a recipient of the Best Student Paper Award in ICB 2006, ICB 2015, and CCBR 2016, and the Best Paper Award in ICB 2007. He was also a recipient of the Best Reviewer Award in IJCB 2014 and CVPR 2019 Outstanding Reviewers. He served as an Area Chair for ICPR 2016, ICB 2016, and ICB 2018, and as a PC member for ICCV, CVPR, and ECCV.

Weidong Hu received the B.S. degree in microwave technology and the M.S. and Ph.D. degrees in communication and electronic system from the National University of Defense Technology, Changsha, China, in 1990, 1994, and 1997, respectively. He is currently a Professor with the College of Electronic Science, National University of Defense Technology. His research interests include radar signal and data processing.