A Unified Multi-Task Learning Architecture for Fast and Accurate Pedestrian Detection

Chengju Zhou, Meiqing Wu, *Member, IEEE*, and Siew-Kei Lam, *Senior Member, IEEE*

***Abstract*— We present a unified multi-task learning architecture for fast and accurate pedestrian detection. Different from existing methods which often focus on either a new loss function or architecture, we propose an improved multi-task convolutional neural network learning architecture to effectively and efficiently interfuse the task of pedestrian detection and semantic segmentation. To achieve this, we integrate a lightweight semantic segmentation branch to Faster R-CNN detection framework that enables end-to-end hard parameter sharing in order to boost the detection performance and maintain computational efficiency as follows. Firstly, a Semantic Segmentation to Feature Module (SS2FM) refines the convolutional features in RPN stage by integrating the features generated from the semantic segmentation branch. Secondly, a Semantic Segmentation to Confidence Module (SS2CM) refines the classification confidence in RPN stage by fusing it with the semantic segmentation confidence. We also introduce an effective anchor matching point transform to alleviate the problem of feature misalignment for heavily occluded pedestrians. The proposed unified multi-task learning architecture lends itself well to more robust pedestrian detection in diverse scenarios with negligible computation overhead. In addition, the proposed architecture can achieve high detection performance with low resolution input images, which significantly reduces the computational complexity. Experiment results on CityPersons and Caltech datasets show that our method is the fastest among all state-of-the-art pedestrian detection methods while exhibiting competitive detection performance.**

***Index Terms*— Multi-task learning, pedestrian detection, semantic segmentation, feature aggregation.**

I. Introduction

PEDESTRIAN detection plays a key role in many computer vision applications such as pedestrian identification, autonomous driving, robotic navigation and video surveillance [1], [2]. Many research efforts in pedestrian detection have been undertaken in recent years [2]–[11], however they still perform poorly in challenging cases such as heavy occlusion, highly cluttered background, low resolution, etc. [12]. In addition to high robustness, real-world applications often necessitate that the algorithms run at high-speed on limited computational resources (e.g., embedded systems employed in autonomous vehicle and robotics) [6], [13], [14]. This imposes a low computational complexity requirement on pedestrian detection algorithms in many applications.

Pedestrian detection falls under a broader problem of object recognition, where the existing works are categorized into the following tasks: object detection and semantic segmentation. Object detection classifies and localizes a region of a specific object instance, while semantic segmentation assigns each pixel with the corresponding object class label. Recent methods rely on deep convolution neural networks to learn semantic features for representing objects effectively [15], [16] [17], [18] [6], [19] [14]. However, both tasks learn semantic features from different types of inputs, and hence they are characterized by different advantages and disadvantages. Object detection works well for localizing distinct objects but often includes unnecessary backgrounds. Semantic segmentation can provide object pixel-wise boundary but fails to distinguish objects within same class, especially in inter-occlusion cases. Existing works in pedestrian detection typically rely on either an object detection framework [6], [7] [5] or a semantic segmentation framework [18], [20] to locate pedestrians in an image. As such, these works suffer from the inherent limitations mentioned above.

Recent studies demonstrated that learning multiple tasks simultaneously while exploiting commonalities and differences across tasks, i.e. multi-task learning, can lead to improved prediction accuracy and learning efficiency [17], [21]. In multi-task learning, tasks share a common low-dimensional representation, which can be jointly learnt with task specific parameters to improve the performance of each task. Some recent works have included semantic segmentation results as a prior to boost the performance of pedestrian detection. In [11], a semantic segmentation mask is introduced as an additional semantic channel to the RGB channels to improve the detection of small pedestrians. In [22], [23], the semantic segmentation mask is utilized to compute the scaling factors to re-score the estimation results. However, all of these works require a separate network (e.g., FCN [20]) to generate the semantic segmentation mask. Moreover, pixel-wise annotations, which are costly to obtained, are required to provide the supervision signals for training (e.g., Cityscapes [16]). In [24], an additional semantic segmentation branch is used with the detection head in a weakly-supervised manner. The full potential of fusing semantic segmentation with pedestrian detection is not fully exploited in [24], and we will later show that the learned features are still not distinctive enough in some challenging cases (e.g., heavily occluded pedestrians).

State-of-the-art pedestrian detection methods also often induce high computational complexity [6], [14], [22], [23]. For

Manuscript received December 26, 2019; revised April 25, 2020 and June 28, 2020; accepted August 13, 2020. Date of publication September 4, 2020; date of current version February 2, 2022. This work was supported in part by the National Research Foundation Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) Program with the Technical University of Munich at TUMCREATE. The Associate Editor for this article was Z. Duric. *(Corresponding author: Chengju Zhou.)*

The authors are with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798 (e-mail: zhou0271@e.ntu.edu.sg).

Digital Object Identifier 10.1109/TITS.2020.3019390

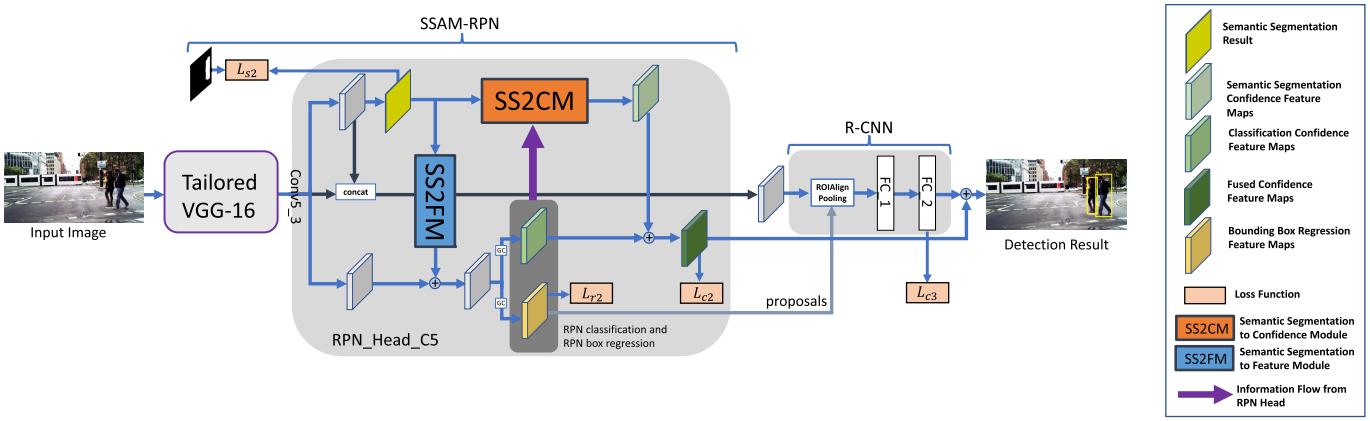


Fig. 1. Illustration of proposed learning architecture. The proposed architecture consists of two stages: RPN with Semantic Segmentation Aggregation Module (referred as SSAM-RPN) and R-CNN. In SSAM-RPN, there are totally two detection heads attached to Conv4_3 and Conv5_3 named as RPN_Head_C4 and RPN_Head_C5 respectively. Here we only show the RPN head attached to Conv5_3 feature maps (referred as RPN_Head_C5) for simplicity. The Semantic Segmentation to Feature Module (referred as SS2FM, highlighted with blue rectangle) is exploited for building robust proposal convolutional features. The Semantic Segmentation to Confidence Module (referred as SS2CM, highlighted with orange rectangle) is used for generating confidence from semantic segmentation result. The R-CNN of Faster R-CNN is adopted to refine the pedestrian proposals from SSAM-RPN. We use rectangle with orange background to represent corresponding losses. In the tailored VGG-16, we only keep conv1-5 layers (except for the pool4 layer). GC stands for Group Convolution.

example, F-DNN2+SS [23] requires 2.48 seconds to process one image from Caltech dataset [25] using NVIDIA TITAN X GPU. The high computational complexity of F-DNN2+SS comes from its combination of predictions from several backbone networks (including GoogleNet [26] and ResNet-50 [27]) and a separate mask generation network. In order to reduce overall computational complexity, CSID [14] uses a much lighter convolutional neural network as backbone network (i.e., DLA-34 [28]). CSID obtains a new state-of-the-art detection performance on recently released CityPersons dataset [11] with a test time of 0.16 seconds per image on NVIDIA GTX 1080Ti GPU, which is about 2 times faster than its baseline CSP [6] that takes ResNet-50 [27] as backbone network. Even though CSID achieves a much better inference efficiency than CSP, it is still too high for many real-world applications.

Our work addresses the high computational complexity of pedestrian detection while maintaining competitive detection performance. We proposed an end-to-end multi-task learning neural network architecture that effectively interfuses pedestrian detection and semantic segmentation tasks. The proposed architecture simultaneously learns the pedestrian detection and semantic segmentation tasks using only bounding box annotations. Specifically, we integrate a lightweight semantic segmentation head to the Faster R-CNN framework in order to extract semantic segmentation features in a weakly-supervised manner. The semantic segmentation features are then infused with the proposal convolutional features to obtain more robust and distinctive features for RPN proposal generation (SS2FM in Fig. 1 and Fig. 2a). Another lightweight module is introduced to compute the semantic segmentation confidence (SS2CM in Fig. 1 and Fig. 2b), which is then fused with the classification confidence to derive the final pedestrian proposal confidence. Finally, the segmentation convolutional feature maps are concatenated with top convolutional feature maps from the backbone network to serve as features for the R-CNN step. By aggregating semantic segmentation features from the two modules (SS2FM and SS2CM), the proposed method detects pedestrians more robustly. It is noteworthy

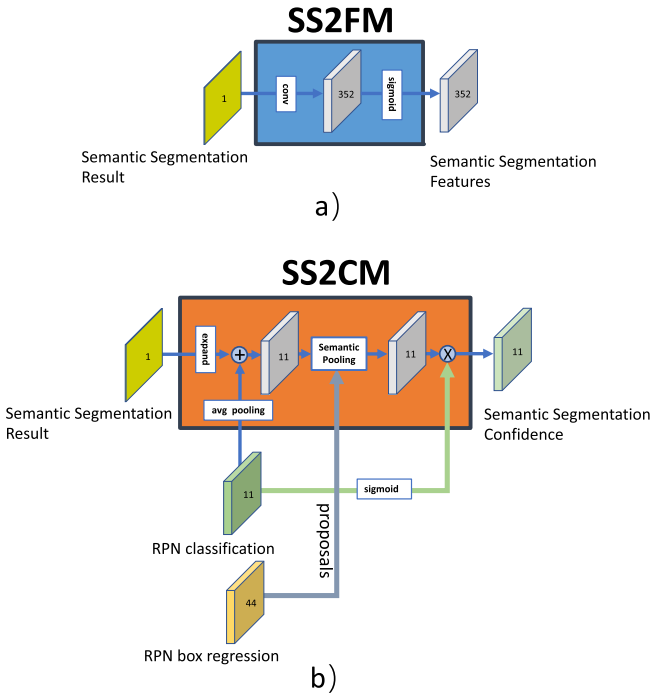


Fig. 2. Details of proposed a) SS2FM, and b) SS2CM. The SS2FM takes semantic segmentation result as input. The SS2CM takes semantic segmentation result, the RPN classification and RPN box regression as input. The digits attached to feature maps are the number of channels.

that the semantic segmentation branch is introduced in a hard parameter sharing manner, hence it incurs negligable computation overhead. In addition, group convolution operations are introduced to reduce the computational complexity in RPN stage. We show that the proposed unified multi-task learning architecture can achieve high detection performance with low resolution input images, which significantly reduces the computational complexity. The utilization of low resolution input images can also lead to higher power efficiency [29], which is desirable in many applications that are battery-powered. Experiment results on well-known datasets show that

our method is the fastest among all state-of-the-art pedestrian detection methods while exhibiting competitive detection performance.

A. Main Contributions

Our contributions are summarized as follows:

1) We propose a unified multi-task end-to-end neural network architecture for pedestrian detection that simultaneously achieves low computational complexity and robust detection in challenging scenarios (e.g., heavily occluded scenes).
2) This is the first work to demonstrate that robust pedestrian detection can be achieved using low resolution input images.
3) We introduce a simple and effective anchor matching point transform to alleviate feature misalignment for heavily occluded pedestrians.
4) Finally, experiment results on well-known datasets show that our method is the fastest among all state-of-the-art pedestrian detection methods while exhibiting competitive detection performance.

II. Related Work

1) Multi-Task Learning: Multi-task learning is defined as follows [30]: Given m learning tasks $\{T_i\}_{i=1}^{m}$ where all the tasks or a subset of them are related, multi-task learning aims to improve the learning of a model for T_i by using the knowledge contained in all or some of the m tasks.

Multi-task learning has been widely used in machine learning [30]–[34]. Multi-task learning can also work as regularization by providing inductive bias to each task [31]. By exploiting commonalities and differences across tasks, multi-task learning can achieve better generalizations for each task with less annotations than single task learning [35]. Most multi-task learning in deep neural network can be divided into two groups based on how the parameters of hidden layers are shared. The first category is called soft parameter sharing, wherein each task has its own model and parameters. The parameters in each model are encouraged to be similar by regularizing their distance [36]–[41]. The second category is hard parameters sharing, which generally shares several hidden layers between all tasks and keeps several task-specific branch head for each learning task [17], [42]–[45]. Compared with hard parameter sharing, soft parameter sharing typically requires a carefully designed knowledge sharing mechanism and has higher overall computational complexity as each task is processed by a separate model.

2) Pedestrian Detection: Pedestrian detection has achieved notable progress in recent years with the prevalence of deep convolutional neural network (DCNN). Many works were proposed to improve pedestrian detection by either adding new layers to the existing Faster R-CNN object detection network or designing better loss functions to learn more robust convolutional classification features. By integrating the RPN of Faster R-CNN and a boosted forest as downstream classifier, RPN+BF [46] obtained improved performance on Caltech dataset. In Multi-Scale CNN (MS-CNN) [47], a unified DCNN is proposed to perform detection at various intermediate network layers such that the receptive fields match objects at different scales. F-DNN (Fused Deep Neural Network) [22] proposes to combine more classification networks (including GoogleNet [26] and ResNet-50 [27]) as downstream classifier. AR-Ped [5] proposes an autoregressive pedestrian detection framework that utilizes a stackable de-encoder module with convolutional re-sampling layers, which can autoregressively produce and refine both features and classification predictions. Inspired by the "Squeeze-and-Excitation" (SE) block in [48], FasterRCNN+ATT [49] proposes to employ channel-wise attention to handle occlusions for pedestrian detection. This is motivated by the findings that many channel features are localizable and often correspond to different body parts. An attention vector is learned from attention network to re-weight the top convolution channels as attention guidance and notable performance improvement is achieved for occluded cases. GDFL [9] and its extension CA-GDFL [50] propose to exploit scale-aware pedestrian attention masks and a zoom-in-zoom-out module to improve the capability of the feature maps to identify small and occluded pedestrians. CSP [6] reformulates pedestrian detection as a problem of center and scale predictions. In order to improve the detection accuracy in the crowd, OR-CNN [3] designs a new aggregation loss to enforce proposals to be close and located compactly to the corresponding objects. RepLoss [4] proposes to exploit the repulsion loss for pedestrian detection in crowd scenes, which is motivated by the fact that the attraction-by-target loss alone may not be sufficient for training an optimal detector, and repulsion-by-surrounding can be beneficial. Based on CSP, CSID [14] proposes an ID-Map to encode both identity and density pedestrian information for each predicted box, which is used in the post-processing step of NMS (Non-Maximum Suppression) and achieves improved detection performance. The above methods often obtain inferior predictions for pedestrians that are heavily occluded (e.g., only head is visible). This is because these methods rely on features from pedestrian center to classify pedestrian and in heavily occluded scenes, the features are dominated by other objects (e.g., vehicles) or other pedestrian's part as shown in Fig. 3. In addition, CSP and CSID assume a fixed pedestrian aspect ratio (i.e., pedestrian width/pedestrian height), and they only predict the pedestrian height. The pedestrian width is computed based on the fixed aspect ratio. However, such strict assumption would lead to inaccurate predictions.

A few works have attempted to infuse information obtained from semantic segmentation result into the object detection pipeline to improve detection performance. In Faster R-CNN + Seg [11], a semantic segmentation mask is introduced as an additional semantic channel to the RGB channels to improve the detection of small pedestrians. In F-DNN+SS [22] and F-DNN2+SS [23], the semantic segmentation mask is utilized to compute the scaling factors to re-score the estimation results. However, all of these works require a separate network (e.g., FCN [20]) to generate the semantic segmentation mask, hence the pedestrian detection and semantic segmentation tasks are not learnt simultaneously. The separate training of each task cannot exploit the advantages of

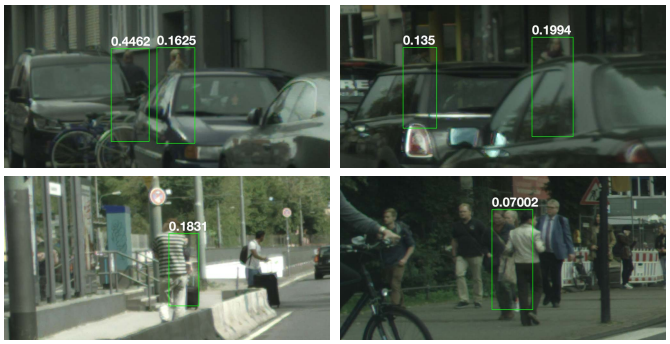


Fig. 3. Visualization of heavily occluded pedestrians from CityPersons dataset. The digits indicate the visibility of each pedestrian.

multi-task learning and reduces the robustness for semantic segmentation as small segmentation errors may be amplified in the process of re-scaling pedestrian predictions. In addition, these methods require pixel-wise annotations, which are costly to obtained, to provide the supervision signals for training (e.g., Cityscapes [16]).

Recent studies demonstrating the utilization of segmentation masks for generic object detection [21], [51], [52] have motivated the authors of SDS-RCNN [24] to jointly learn the pedestrian detection and semantic segmentation tasks by adding a semantic segmentation branch to the top network layer. This led to improved performance on Caltech dataset. The added semantic segmentation branch is only utilized during training, and hence does not incur additional computation overhead in the inference stage. Analysis by the authors shows that the performance gain comes mostly from improved robustness in detecting atypical pedestrians (e.g., partially occluded pedestrians or pedestrians with unusual pose). It is observed in FasterRCNN+ATT [49] that many channels from top convolutional feature maps show highly localizable activation patterns that relate to specific pedestrian body regions or body part. Hence, the success of SDS-RCNN can be attributed by the channel boosting in top convolutional feature maps that are related to pedestrian body. However, this could also increase the risk of inaccurate prediction for heavily occluded pedestrians. For example, when only head and shoulder of pedestrian are visible (as shown in top right of Fig. 3), SDS-RCNN uses pedestrian center to conduct prediction in which features from pedestrian center is heavily influenced by the vehicle. This increase the risk of misclassification. In order to alleviate this problem, we propose a simple and effective method that exploits pedestrian upper center rather than pedestrian center to conduct prediction. This can be achieved by simply transferring the matching point between feature map and anchors from anchor center to anchor upper center as shown in Fig. 8.

3) Computational Complexity: Very few works on pedestrian detection focus on reducing the computational complexity, despite this being a key factor in practical scenarios, e.g., autonomous driving and robot navigation. The computation platforms of such applications necessitate low complexity pedestrian detector, as they often have tight computational resources and employ battery as their main power source. The computational complexity of DCNN based pedestrian detector [11], [22] [4], [24] [5], [6] [14], is mainly contributed by: the computation of backbone networks, and the combination of feature maps for final pedestrian prediction. For instance, VGG-16 [53], ResNet-50 [27] and DLA-34 [28] need about 15.5 billion FLOPs, 3.8 billion FLOPs and 3.0 billion FLOPs for an input image with resolution of 224×224.[1] As shown in Table VIII (i.e., CSP), the feature maps combination occupies most of the execution time as higher resolution feature maps are required to achieve robust pedestrian detection. The challenge in reducing computational complexity arise from the fact that existing mechanisms for improving the detection performance often incur significant computational overheads. In order to learn more robust features for small pedestrian detection, existing works often fuse convolutional feature maps from several intermediate backbone network layers [6], [14] or combine features from several backbone networks [22]. This increases the overall computational complexity of pedestrian detector. F-DNN2+SS [22] combines several backbone networks and requires 2.48 seconds to process one image from Caltech dataset using NVIDIA TITAN X GPU, which is un-acceptable in real-world applications. Existing works also often use larger input image resolution than the original image resolution for inference [3], [4] [24] in order to achieve better detection performance, especially for small pedestrians. This is because lower image resolution would reduce the discriminative power of pedestrians from the background. However, the computational complexity increases by a power-of-two factor with the input image resolution. As such, achieving high detection performance and reducing computation complexity are often viewed as orthogonal goals in the existing works.

4) Semantic Segmentation: The semantic segmentation task assigns semantic labels to each pixel [20], [54] [55], [56] [57]. Recently, real-time semantic segmentation algorithms are proposed to meet the demands for fast response in practical applications. E-Net [58] is a neural network architecture that is designed from scratch for semantic segmentation and achieves low computational complexity. The authors in [55] designed SegNet, a deep convolutional network architecture which includes small network structures and skip connections. SegNet achieves high efficiency both in terms of computational complexity and memory consumption. ERFNet [56] exploits a novel layer that uses residual connections and factorized convolutions. ERFNet achieves top precision on CityScapes dataset, while running orders of magnitude faster than state-of-the-art methods. ESPNet [59], [60] was introduced as a light-weight and power-efficient network which is based on a new convolutional module named efficient spatial pyramid (ESP). DFANet [57] employs a single lightweight backbone and aggregates discriminative features through sub-network and sub-stage cascade. DFANet achieves 160 FPS (Frame Per Second) with input image resolution 512×1024 on Cityscapes dataset [16], while obtaining comparable performance with state-of-the-art methods.

The lack of common datasets for both pedestrian detection and semantic segmentation tasks, however, makes it challenging to directly exploit semantic segmentation result for

[1] https://github.com/osmr/imgclsmob/blob/master/chainer_/README.md

accurate pedestrian detection. The difficulty in obtaining such datasets arises from the fact that the pixel-wise annotations for semantic segmentation is much more labour intensive than box-wise annotations for pedestrian detection. Currently, only CityPersons dataset, which is widely-used for pedestrian detection, has pixel-wise annotations for each image [16]. Several popular pedestrian detection benchmarks, including INRIA, Caltech and recently published CrowdHuman [61] and Wide Pedestrian [62] datasets, do not have pixel-wise annotations for semantic segmentation. In addition, effectively and efficiently exploring semantic segmentation for pedestrian detection is still an unsolved problem. The methods adopted by Faster R-CNN + Seg [11], F-DNN+SS [22] and F-DNN2+SS [23] to exploit semantic segmentation results do not rely on end-to-end training, which limits their performance in pedestrian detection. In this work, we show that the proposed learning architecture can automatically extract coarse semantic segmentation results to achieve competitive pedestrian detection performance and also lower runtime when compared with state-of-the-art methods. As such, the proposed learning architecture can directly exploit semantic segmentation results for accurate pedestrian detection in all pedestrian detection datasets, including those without pixel-wise annotations.

III. Proposed Method

Our work is motivated by the fact that the pedestrian regions are highlighted in semantic segmentation results while non-pedestrian regions (i.e., backgrounds) are suppressed. The highlighted regions provide a new perspective to locate pedestrian from background, i.e., candidate with highlighted region support in semantic segmentation result is more likely to be a real pedestrian. Our proposed method extends the Faster R-CNN detection framework to multi-task learning that simultaneously learns the task of pedestrian detection and semantic segmentation, and it consists of two stages: RPN with Semantic Segmentation Aggregation Module (referred as SSAM-RPN) to generate pedestrian proposals with semantic segmentation confidence, and R-CNN to refine pedestrian proposals from SSAM-RPN. In R-CNN stage, we tailor R-CNN from Faster R-CNN detection framework and combine confidence from SSAM-RPN and R-CNN as final pedestrian confidence. Fig. 1 illustrates the proposed learning architecture.

A. RPN With Semantic Segmentation Aggregation Module (SSAM-RPN)

The RPN in Faster R-CNN aims to obtain a set of bounding box proposals with certain confidence levels for pedestrians. In this section, we introduce a RPN with Semantic Segmentation Aggregation Modules (SSAM-RPN) to obtain better pedestrian proposals. A tailored VGG-16 [53] is used as the backbone network and we only keep conv1-5 layers (except for the pool4 layer) in the proposed SSAM-RPN.

In order to take advantage of multi-task learning, we extend the vanilla RPN to simultaneously learn the task of pedestrian detection and semantic segmentation. The task of semantic segmentation aims to predict class label for each pixel, i.e., pedestrian and non-pedestrian, as shown in Fig. 5b. It can be observed that the pedestrian regions are highlighted in the semantic segmentation results. Intuitively, the semantic segmentation results can potentially improve the pedestrian detector's performance, as demonstrated in Faster R-CNN+Seg [11] and F-DNN2+SS [23]. However, Faster R-CNN+Seg and F-DNN2+SS employ independent models to obtain semantic segmentation results which prevent them from simultaneously learning the pedestrian detection and semantic segmentation tasks. In addition, using a separate model induces inferior performance and higher computational complexity as discussed earlier. In order to overcome these drawbacks, we propose a multi-task learning architecture to obtain semantic segmentation result based on vanilla RPN. Specifically, we attach semantic segmentation branch to the RPN backbone network designed for pedestrian detection, which can integrate semantic segmentation features into backbone network and share computation with detection task. As shown in Fig. 1, the connections from tailored VGG-16 network to L_{s2} is a semantic segmentation branch for RPN detection head attached to Conv5_3 layer. Compared with semantic segmentation branch used in SDS-RCNN, our proposed approach adds a segmentation convolutional layer before generating semantic segmentation result. This enables the multi-task learning architecture to learn more compact semantic segmentation features and provide additional feature maps for R-CNN step as described in the next sub-section. We use weakly bounding box annotations designed for pedestrian detection as semantic segmentation annotations, in which pedestrian regions are labelled as foreground and others are labelled as background. The box-wise annotations have minor differences from pixel-wise annotations when the image is downsampled significantly across the network layers as shown in Fig. 4. The box-wise annotations are demonstrated to work well for highlighted pedestrian region and are sufficient for detection task that focuses on predicting pedestrian bounding box rather than pixel labels for pedestrians. The semantic segmentation result from branch corresponding to L_{s2} is shown in Fig. 5b. We can observe that slightly larger regions than box-wise annotations are classified as pedestrian.

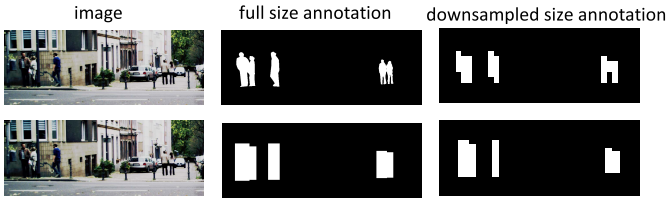


Fig. 4. Visualization of pixel-wise annotations (upper row) and weakly box-wise annotations (lower row) from CityPersons dataset. It can be observed that the differences between pixel-wise annotation and box-wise annotation become smaller as the image is downsampled.

In the proposed architecture, the following two modules are introduced to exploit the semantic segmentation results to boost pedestrian detection: Semantic Segmentation to Feature Module (SS2FM) and Semantic Segmentation to Confidence Module (SS2CM).

1) Semantic Segmentation to Feature Module (SS2FM): SS2FM (highlighted with blue rectangle in Fig. 2a) aims to

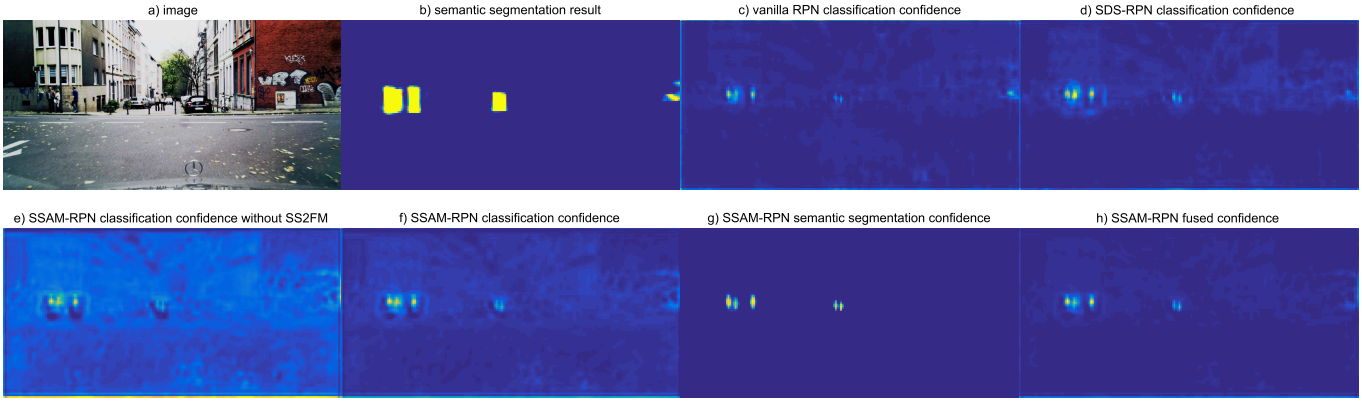


Fig. 5. Visualization of semantic segmentation result and various confidence feature maps from RPN stage. Title represents the name of feature map. It can be observed that the background region in fused feature map (i.e., h)) of proposed method are suppressed significantly than vanilla RPN (i.e., c)) and SDS-RPN (i.e., d), RPN stage of SDS-RCNN [24] as standalone detector). Even though there are some errors at the right boundary of semantic segmentation result (i.e., b)), the learned semantic segmentation confidence of the proposed method eliminates these segmentation errors and provides accurate evaluation for pedestrians. The brighter pixels indicates larger digit value. (Best viewed in color).

incorporate semantic segmentation result into proposal convolutional features. More concretely, the semantic segmentation result is first applied with convolution and sigmoid operations, which can reduce the problem of scale mismatch with features from proposal convolutional feature maps [9]. The semantic segmentation result is then added with proposal convolutional features to provide more discriminative features for pedestrian proposal generation (e.g., RPN classification and bounding box regression highlighted in gray rectangle in Fig. 1).

Compared with GDFL [9] and its extension CA-GDFL [50] that perform multiplication operation over feature maps from backbone network, our proposed SS2FM infuses semantic segmentation features into proposal convolutional features more smoothly. This is due to the fact that semantic segmentation is supervised by weakly box-wise annotations which would incur some inaccurate semantic segmentation predictions as shown in the right boundary of Fig. 5b. These inaccurate predictions would be significantly amplified by the multiplication operation, thereby increasing the risk of mis-classification. Another reason for adopting addition instead of multiplication operation is that our learned proposal convolutional features include negative values, and multiplication operation will lead to the problem of scale inconsistency. The classification confidence without and with proposed SS2FM are shown in Fig. 5e and f respectively. It is evident that the background confidence are suppressed when SS2FM is exploited and the classification confidence becomes more distinct in pedestrian regions (i.e., Fig. 5f).

2) Semantic Segmentation to Confidence Module (SS2CM): In contrast to SS2FM which helps to build more discriminative convolutional features, SS2CM focuses on obtaining more accurate pedestrian confidence by exploring semantic segmentation result. The motivation for SS2CM stems from the fact that a pedestrian candidate is more likely to be a true pedestrian if its associated region in semantic segmentation result is highlighted. This is due to the fact that non-pedestrian regions are suppressed in semantic segmentation results as shown in Fig. 5b.

As shown in the orange rectangle in Fig. 2b, SS2CM takes several feature maps as input, including semantic segmentation result, classification confidence from classification branch of RPN head and proposals from bounding box regression branch of RPN head. Firstly, the semantic segmentation feature map is expanded and added with classification confidence after average pooling to obtain new feature maps. Then semantic pooling operation is applied on the new feature maps with proposals from bounding box regression branch of RPN head (highlighted in blue-gray connection in Fig. 2b). It is worth noting that the semantic pooling is conducted with every proposal and no NMS (Non-Maximum Suppression) is used, which implies that every proposal has its own semantic segmentation confidence. Finally, semantic segmentation confidence is obtained after scaling with classification confidence using sigmoid operation, which enables the extracted semantic segmentation confidence to maintain consistency with classification confidence. The obtained semantic segmentation confidence feature map is shown in Fig. 5g. It can be observed that only pixels around pedestrian upper center are highlighted and background pixels are suppressed, which implies that the semantic segmentation confidence provides a reasonable prediction for pedestrian proposal. It can also be observed that there is a small brighter region near the left boundary of semantic segmentation result and this is eliminated in semantic segmentation confidence feature map. This implies that the proposed SS2CM is robust to small semantic segmentation errors which is common due to the usage of weakly box-wise annotation to supervise semantic segmentation.

Compared with F-DNN2+SS that employs binary mask (e.g., 1 for pedestrian and 0 for background) as semantic segmentation prior, the proposed SS2CM have the following advantages: Firstly, our semantic segmentation result is learned from a unified multi-task learning framework and is not truncated into binary mask which will lose some semantic segmentation information. Secondly, our SS2CM works as a part of a unified multi-task learning framework and can be trained in an end-to-end way which can learn more reasonable and robust semantic segmentation features.

After obtaining semantic segmentation confidence from SS2CM, we combine it with classification confidence to serve as final pedestrian proposal confidence (refered as fused

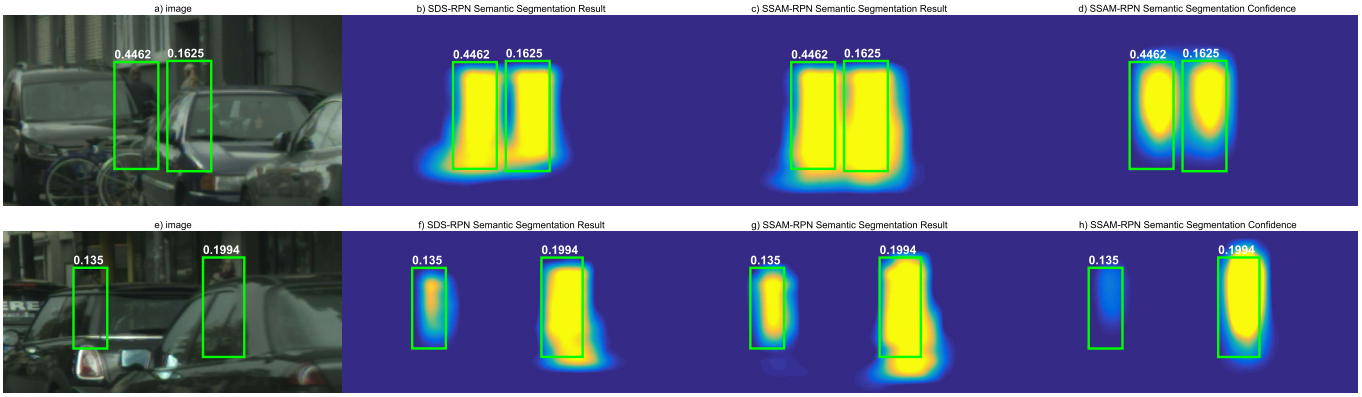


Fig. 6. Visualization of semantic segmentation result of SDS-RPN [24] and proposed SSAM-RPN, and semantic segmentation confidence of proposed SSAM-RPN for heavily occluded pedestrians. It can be observed that both SDS-RPN and proposed SSAM-RPN can highlight pedestrian regions in semantic segmentation result (i.e., b), c) and f), g)). Our SSAM-RPN can obtain reasonable pedestrian evaluation with proposed SS2CM (i.e., d) and h)). The brighter pixels indicates larger digit value. The number on each bounding box indicates the visibility of pedestrian. (Best viewed in color).

confidence) as shown in Fig. 5h. Benefiting from SS2CM, the fused confidence can effectively suppress the non-pedestrian regions, especially on the regions at the right boundary. This implies that the semantic segmentation confidence works well as complementary information for obtaining better pedestrian proposal prediction. Compared with classification confidence feature maps for vanilla RPN (i.e., Fig. 5c) and SDS-RPN[2] (i.e., Fig. 5d, RPN stage of SDS-RCNN as standalone detector), the background and pixels around pedestrians in fused confidence feature map are suppressed significantly. This demonstrates the effectiveness of proposed SS2CM and the necessity to exploit semantic segmentation result for more accurate pedestrian proposal prediction.

3) Prediction Location: As described in the Related Work section, one limitation of existing pedestrian detection methods (including both anchor-based methods and anchor-free methods), is that they utilize the pedestrian center for classification and regression. This works well when the pedestrian center is visible. However, when the pedestrian is heavily occluded, for example, only head or shoulder is visible, the features around the pedestrian center are often dominated by other pedestrians (bottom row of Fig. 3) or objects (top row of Fig. 3), which can lead to mispredictions. Fig. 7 shows the statistics of occlusion patterns from CityPersons dataset [11], wherein most occlusions reside in lower pedestrian parts. In order to alleviate the misalignment between features and prediction location, we propose to move the anchor matching point between feature map and anchors from anchor center to anchor upper center (see Fig. 8). As such, the pedestrian prediction location also changes from pedestrian center to pedestrian upper center. This enables the features that are utilized for pedestrian prediction to be applicable even in heavily occluded cases (as shown in Fig. 9), hence increasing the robustness of pedestrian prediction.

4) Analysis of Occlusion Robustness for Proposed SS2FM and SS2CM: Our proposed method achieves robustness in detecting occluded pedestrian from both the feature level and confidence level. The semantic segmentation result of

[2]Since SDS-RCNN did not release its result and model on CityPersons dataset, we train SDS-RPN using our setting.

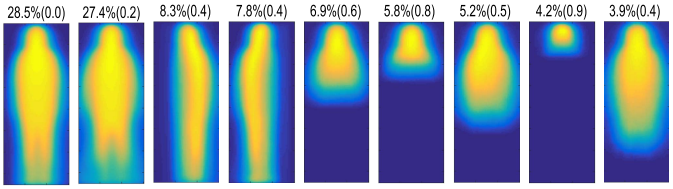


Fig. 7. Top occlusion patterns of pedestrians on CityPersons dataset. Two numbers on top indicate percentage and average occlusion ratio of samples clustered into each pattern. Image is extracted from [11].

SDS-RPN and our proposed method for occluded pedestrians are shown in Fig. 6. It can be observed that the pedestrian regions are highlighted even though most of pedestrian parts are occluded, which is partly attributed to the ability to simultaneously learn the task of pedestrian detection and semantic segmentation. This semantic segmentation result implies that reasonable features for RPN head can be constructed with proposed SS2FM as semantic segmentation features from highlighted pedestrian regions are infused into the proposal convolutional features. The semantic segmentation confidence for occluded pedestrians are shown in the right of Fig. 6. Although the pedestrians are heavily occluded, the proposed SS2CM can also obtain reasonable predictions from semantic segmentation result. This extracted semantic segmentation confidence works as complementary information with RPN classification confidence, and accurate confidence can be produced even when most of the pedestrian body is occluded as shown in Fig. 10. Even though SDS-RPN can also highlight occluded pedestrian parts in semantic segmentation result, SDS-RPN does not use this prior to boost detection and hence achieves inferior performance for heavily occluded pedestrians.

5) Low Resolution Image for Inference: As mentioned in the Related Work section, existing pedestrian works often use original or higher resolution input image during inference in order to achieve better detection performance. This increases the computational complexity which grows by a power-of-two factor with image resolution. In the proposed method, the box annotations designed for pedestrian detection is used for the task of semantic segmentation, which implies that our learnt semantic segmentation result is not pixel-wise accurate.

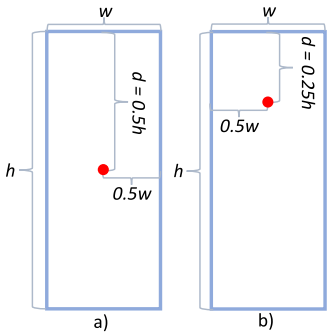


Fig. 8. Visualization of anchor center as matching point a) and proposed anchor upper center as matching point b). h and w represent the height and width of anchor respectively. d represents the distance between anchor upper boundary and matching point.

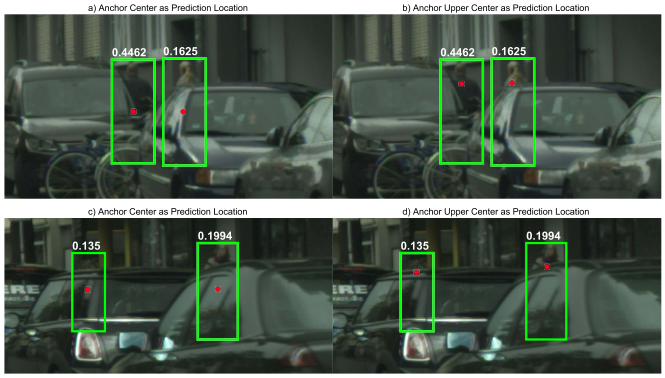


Fig. 9. Visualization (i.e., red dot) of anchor center as prediction location a) and c), and proposed anchor upper center as prediction location b) and d). With proposed matching point transform, the pedestrian prediction location changes from pedestrian center to pedestrian upper center. The number on each bounding box indicates the visibility of each pedestrian.

When the input image resolution increases, the differences between box-wise annotations and pixel-wise annotation will also increase. Hence, we propose to use input image with smaller resolution during inference. Our experiments show that the proposed method can produce more reasonable semantic segmentation result with lower input image resolution, which helps to significantly reduce the overall computational complexity.

In addition to the detection head that is attached to the top backbone network layer (i.e., RPN_Head_C5 in Fig. 1), we add another detection head to Conv4_3 layer called RPN_Head_C4. Only SS2FM is employed in RPN_Head_C4 detection head to improve the training stability. RPN_Head_C4 is only used in training process and hence has no effect on the inference execution time. Besides, group convolution (GC in Fig. 1) is utilized in the proposed SSAM-RPN detection head. As discussed in the experimental section, GC can lower the computation complexity without compromising on the detection performance.

6) Loss Function: Our proposed SSAM-RPN is trained by mining the following loss function:

$$\begin{aligned} L_{rpn} = {} & w_{cls\_1}L_{c1} + w_{bb\_reg\_1}L_{r1} + w_{seg\_1}L_{s1} \\ & + w_{cls\_2}L_{c2} + w_{bb\_reg\_2}L_{r2} + w_{seg\_2}L_{s2} \end{aligned} \quad (1)$$

where L_{c*} and L_{s*} are cross-entropy loss for classification and semantic segmentation. The w_* are weights for corresponding loss function and the subscripts of *1* and *2* correspond to losses in RPN_Head_C4 and RPN_Head_C5 respectively. The classification loss L_{c*} and segmentation loss L_{s*} are designed for binary classification problem (i.e., pedestrian vs. non-pedestrian). The L_{r*} is a modified smooth-L_1 loss as follows:

$$L_r(x) = \begin{cases} 0.5x^2\beta, & \text{if } |x| < \beta \\ |x| - 0.5\beta, & \text{otherwise} \end{cases} \quad (2)$$

where β is parameter to control where the frontier between the L1 and the L2 losses are switched, and x is the difference between predicted value and ground truth at bounding box regression of Faster R-CNN [63].

In SSAM-RPN, we use a stricter labelling policy than Faster R-CNN [63]. In particular, a proposal is labelled as positive (i.e., pedestrian) if the IoU (Intersection over Union) with groundtruth box is larger than 0.55 for RPN_Head_C5 and 0.6 for RPN_Head_C4. Otherwise the proposal is labelled as negative (i.e., non-pedestrian). This labelling policy aims to learn more compact features from RPN_Head_C4 and alleviate detection difficulty for RPN_Head_C5. The detection result of SSAM-RPN is obtained after applying NMS with threshold of 0.5 on the proposals. In semantic pooling process, we use ROIAlign pooling operation [17] to extract a 7×7 feature map, and its average is used as semantic segmentation confidence for each proposal.

B. R-CNN Binary Classifier

We tailor R-CNN from Faster R-CNN as binary classifier (i.e., pedestrian vs. non-pedestrian) in the proposed method, wherein only classification branch is kept. The pedestrian regions have been highlighted in semantic segmentation results as shown in Fig. 5b which implies that the convolutional features in semantic segmentation branch can provide some cues to recognize pedestrian. Therefore, we concatenate the Conv5_3 feature maps with semantic segmentation convolutional feature maps as R-CNN input feature maps as shown in Fig. 1.

We train R-CNN by mining a cross-entropy loss for pedestrian classification. The ROI pooling operation is replaced with ROIAlign pooling operation in order to alleviate the problem of pooling bin collapse [46] if ROI's input resolution is smaller than the output (i.e., 7×7 which is 56×56 in input image with our tailored VGG-16) as the latter will induce the extracted features flat and less discriminative. The dimension of fully connected layer is set to 2048 and one dropout layer with 0.5 rate is used to reduce the effect of overfitting in R-CNN. The proposals with IoU larger than 0.55 are labelled as positive (i.e., pedestrian), otherwise they are labelled as negative (i.e., non-pedestrian).

IV. Experiments

In this section, we first introduce the pedestrian detection datasets and evaluation metrics used in our experiments. Then we show experiment results to compare the detection performance and test time of the proposed method with state-of-the-art methods. Finally, we will report the results of our ablation studies for the proposed method on CityPersons dataset.

TABLE I
DETECTION PERFORMANCE AND TEST TIME COMPARISON WITH STATE-OF-THE-ARTS ON CITYPERSONS DATASET

Method	Backbone	Reasonable	Heavy	Partial	Bare	Small	Medium	Large	All	GPU	Test Image Size	Test Time
Faster R-CNN	VGG-16	15.4	-	-	-	25.6	7.2	7.9	-	-	2048 × 1024	-
	VGG-16	12.8	-	-	-	-	-	-	-	-	2662 × 1331	-
Faster R-CNN+Seg	VGG-16	14.8	-	-	-	22.6	6.7	8.0	-	-	2048 × 1024	-
OR-CNN	VGG-16	12.8	55.7	15.3	6.7	-	-	-	-	-	2048 × 1024	-
	VGG-16	11.0	51.3	13.7	5.9	-	-	-	-	-	2662 × 1311	-
RepLoss	ResNet-50	13.2	56.9	16.8	7.6	-	-	-	-	-	2048 × 1024	-
	ResNet-50	11.6	55.3	14.8	7.0	-	-	-	-	-	2662 × 1331	-
	ResNet-50	10.9	52.9	13.4	6.3	-	-	-	-	-	3072 × 1536	-
GDFL	VGG-16	14.8	44.2	-	-	-	-	-	-	-	2048 × 1024	-
CA-GDFL	VGG-16	13.6	43.2	-	-	-	-	-	-	1080Ti	2048 × 1024	466ms/img
SDS-RCNN_1	VGG-16	15.1	60.0	15.3	11.0	19.1	6.7	9.3	39.2	1080Ti	1600 × 800	273ms/img
SDS-RCNN_2	VGG-16	16.5	52.9	16.6	12.6	20.3	6.8	9.9	39.6	1080Ti	2048 × 1024	307ms/img
TLL	ResNet-50	15.5	53.6	17.2	10.0	-	-	-	-	-	2048 × 1024	-
TLL+MRF	ResNet-50	14.4	52.0	15.9	9.2	-	-	-	-	-	2048 × 1024	-
ALFNet	ResNet-50	12.0	51.9	11.4	8.4	19.0	5.7	6.6	-	1080Ti	2048 × 1024	270ms/img
CSP	ResNet-50	14.5	53.6	14.8	9.1	24.2	5.7	7.4	43.3	1080Ti	1600 × 800	220ms/img
	ResNet-50	11.0	49.3	10.4	8.1	16.0	3.7	6.5	36.5	1080Ti	2048 × 1024	330ms/img
CSID	DLA-34	8.8	46.6	8.3	5.8	-	-	-	-	1080Ti	2048 × 1024	160ms/img
Ours-SSAM-RPN	VGG-16	11.3	47.6	11.1	7.7	17.4	6.3	6.3	36.4	1080Ti	1600 × 800	90ms/img
Ours	VGG-16	14.5	50.6	13.9	10.6	17.0	4.8	10.2	37.1	1080Ti	2048 × 1024	161ms/img
	VGG-16	10.9	47.5	10.8	7.4	17.6	6.2	6.2	36.2	1080Ti	1600 × 800	110ms/img

A. Datasets and Evaluation Metrics

We conduct experiments on two public datasets: CityPersons [11] and Caltech [25]. The CityPersons dataset is built upon the Cityscapes dataset [16] in which data is collected from multiple cities and countries across Europe. There are a large number of occluded pedestrians in CityPersons dataset that makes it ideal for evaluating the occlusion robustness of the detection approaches. The original image size is 2048×1024, which we resized to 1400×700 for training and to 1600×800 for testing respectively. We conduct experiments on original training and validation subset which include 2975 and 500 images respectively. For the Caltech dataset, we adopt the approach in [64] to extract 42782 images with resolution of 640×480. The Caltech test set has 4024 images which includes 1014 positive images. We use new annotations from [12] for training and testing. The images are resized to 512×384 for training and 544×408 for testing respectively.

For CityPersons and Caltech datasets, we employ commonly-used standard log-average miss rate (MR) between 10^{-2} and 10^{0} of false positive per image (FPPI) to evaluate the detection performance. A detection result is considered as positive (i.e., pedestrian) when its IoU with groundtruth is larger than 0.5. We perform a rigorous evaluation of the detection performance of the proposed method by varying pedestrian height and pedestrian occlusion levels. The details of the evaluation setups are as follows:

1) Reasonable: height $\in [50, \infty]$, visibility $\in [0.65, \infty]$
2) Heavy occlusion: height $\in [50, \infty]$, visibility $\in [0.00, 0.65]$
3) Partial occlusion: height $\in [50, \infty]$, visibility $\in [0.65, 0.90]$
4) Bare: height $\in [50, \infty]$, visibility $\in [0.90, \infty]$
5) Small: height $\in [50, 75]$, visibility $\in [0.65, \infty]$
6) Medium: height $\in [75, 100]$, visibility $\in [0.65, \infty]$
7) Large: height $\in [100, \infty]$, visibility $\in [0.65, \infty]$
8) All: height $\in [20, \infty]$, visibility $\in [0.20, \infty]$

The 'height' in the evaluation setups indicates the pedestrian height while 'visibility' refers to the corresponding pedestrian occlusion level. For example, 'Reasonable' means that the evaluation is conducted on the subset of pedestrians whose height are at least 50 pixels tall and at most 35% of pedestrians are occluded. We use 'Heavy' and 'Partial' to represent the evaluation setups of Heavy occlusion and Partial occlusion in the following part respectively.

We use the evaluation code provided by CityPersons [11] and Caltech [25] to obtain the corresponding MR value.

B. Training Details

Our method is implemented using Pytorch with a single NVIDIA GTX 1080Ti GPU for training and testing. One image is used in each iteration and SGD solver is applied. The VGG-16 [53] network pre-trained from ImageNet [65] is used as backbone network both for CityPersons and Caltech dataset. For CityPersons dataset, the network is trained for 45k iterations with initial learning rate of 0.0025 which is decreased by a factor of 10 at 30k iterations and again at 40k iterations. Weight decay of 0.0005 and momentum of 0.9 are employed. We use a linear warmup strategy for learning rate within first 500 iterations. For Caltech dataset, the network is trained for 80k iterations with initial learning rate of 0.0025 which is decreased by a factor of 10 at 35k iterations. We use the first 300 iterations for learning rate warmup. Other settings are the same for CityPersons dataset.

C. Comparisons With State-of-the-Art Methods

1) CityPersons Dataset: The detection performance comparisons with state-of-the-art methods including Faster R-CNN, Faster R-CNN+Seg, OR-CNN, RepLoss, GDFL, CA-GDFL, re-trained SDS-RCNN,[3] TLL, TLL+MRF, ALFNet, CSP and CSID are shown in Table I. For the state-of-the-art methods, we only list the MR values for the evaluation setups that are released in their papers. We also

[3]We re-train SDS-RCNN as it does not release detection performance and test time on CityPersons dataset.

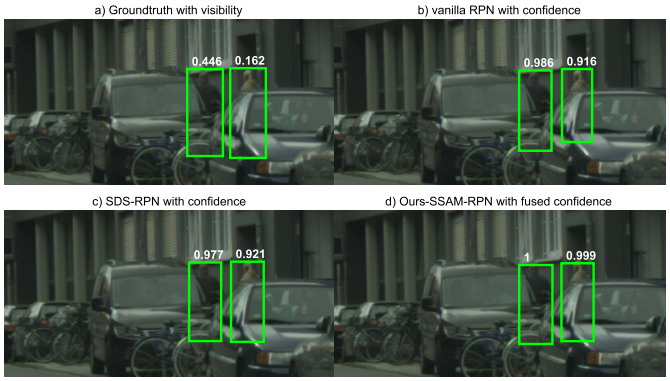


Fig. 10. Visualization of detection results from vanilla RPN b), SDS-RPN c) and our proposed SSAM-RPN d). a) shows the ground truth with visibility. It can be observed that our proposed SSAM-RPN achieves extremely high confidence on both heavily occluded pedestrians while vanilla RPN and SDS-RPN obtain much lower confidence than ours, especially for pedestrian with visibility of 0.162.

re-train SDS-RCNN using mostly the same settings[4] with our proposed method in order to provide a fair comparison with the proposed learning architecture. We conduct experiments for SDS-RCNN on two image resolutions and named them SDS-RCNN_1 and SDS-RCNN_2 respectively as shown in Table I. Ours-SSAM-RPN represents proposed RPN with Semantic Segmentation Aggregation Module as a standalone pedestrian detector. From the table, it can be observed that the proposed method achieves competitive detection performance on several evaluation setups when compared with the best reported detection performance from CSID, including Reasonable, Heavy occlusion, Bare, Large and All setups. Particularly for the Heavy occlusion setup, our proposed method achieves the second lowest MR value which is about 3.8%, 9.5%, 5.4%, 4.5% and 1.9% lower than OR-CNN, RepLoss, TLL+MRF, ALFNet and CSP respectively. Compared with CSID, the MR of our proposed method is only about 0.9% higher. It is noteworthy that the proposed method does not exploit feature maps infusion from multiple inter-media layers which is a key contributor to CSID's better detection performance albeit incurring high computational complexity. RepLoss also obtains same MR value on Reasonable setup with ours but they require a 1.5X upsampled input image (i.e., 3072 $\times$ 1536) which will induce higher computational complexity. In addition, our proposed method only requires bounding box annotations rather than precise pixel-wise annotations in Faster R-CNN+Seg [11] or box-wise visibility annotations in OR-CNN [3]. This implies that the proposed semantic segmentation aggregation module can be easily extended to datasets that only have bounding box annotation and is an effective way to handle heavily occluded pedestrians. GDFL and its extension CA-GDFL obtain better performance than Faster R-CNN. However, their MR values are about 3.9% and 2.7% higher than our proposed method, which demonstrates the advantage of the proposed learning architecture. The re-trained SDS-RCNN (i.e., SDS-RCNN_1) obtains similar detection performance with Faster R-CNN on image resolution of 1600 $\times$ 800, which is about 4.2% higher than our proposed method. This demonstrates the effectiveness of the proposed semantic segmentation aggregation modules.

The test time comparisons with state-of-the-art methods on Citypersons dataset are shown in Table I. We only list the test time of ALFNet, CSP and CSID since other methods have not released their test time on CityPersons dataset in their papers. From the table, it can be observed that the proposed method achieves the lowest test time. Specifically, our proposed method can run about 2.5, 3.0 and 1.5 times faster than ALF, CSP and CSID respectively. It can be observed that ALF, CSP and CSID report their test time when evaluating on input image resolution of 2048 $\times$ 1024 while our reported test time is obtained on input image resolution 1600 $\times$ 800.

In order to compare MR using same input image resolution as ours, we run CSP with 1600 $\times$ 800 input images on the same programming environment and hardware. CSP was chosen for the evaluation as it is the only work that provides the reproduceable trained model as reported in their paper. The MR of CSP increases to 14.5% under Reasonable setup with runtime of about 0.22 seconds per image as illustrated in Table I. Our method with the same input image size of 1600 $\times$ 800 achieves a significantly lower MR of 10.9% with a much lower runtime 110 ms per image. This comparison shows that our proposed method can achieve better detection performance than other pedestrian methods using same resolution image as input, and demonstrates the effectiveness of proposed semantic segmentation aggregation modules. At the same time, our proposed method also demonstrates a new approach for reducing computational complexity of pedestrian detection algorithm which is using low resolution input images. Utilization of low resolution input images offer several advantages in many real-world systems, including less storage requirement, low requirement for CMOS sensor and low power consumption, etc. [29]. The backbone network is another factor that determines the computational complexity. The FLOPs of VGG-16 used in our method is about 4 times higher than ResNet-50 used in CSP, but our method still runs about 2 times faster than CSP on image resolution of 1600 $\times$ 800. This means that the extra computations in CSP are incurred when fusing feature maps from intermediate layers. Similar conclusion can be obtained for CSID. The re-trained SDS-RCNN (i.e., SDS-RCNN_1 and SDS-RCNN_2) achieve inferior inference efficiency compared to the proposed learning architecture (i.e., 2.5 times slower than ours) on image resolution of 1600 $\times$ 800. This is because SDS-RCNN uses VGG-16 network as feature extractor in R-CNN stage which induces much higher computational complexity than ours. It can observed that the extension of GDFL, i.e., CA-GDFL, requires about 0.466 seconds to process one image which is about 4 times slower than the proposed method. As indicated in [50], CA-GDFL is slightly faster than GDFL, which implies that GDFL needs at least 0.466 seconds to process one image from CityPersons dataset. The detection performance and runtime comparisons

[4]We remove the SS2FM and SS2CM of the proposed learning architecture in RPN stage and call it SDS-RPN. The number of proposals fed to R-CNN stage of SDS-RCNN is set to 50 in order to fit the CityPersons dataset. The proposals with IoU larger than 0.7 are labelled as positive (i.e., pedestrian), otherwise they are labelled as negative (i.e., non-pedestrian) as suggested in SDS-RCNN [24].

TABLE II

ABLATION EXPERIMENTS FOR RPN STAGE ON CITYPERSONS DATASET. ✓INDICATES CORRESPONDING COMPONENT IS USED

SSAM-RPN							MR(%)								Test Time
Head_C4		Head_C5			Confidence		Reasonable	Heavy	Partial	Bare	Small	Medium	Large	All	
GC	S2FM	GC	S2FM	S2CM	classification	semantic									
					✓		16.2	53.2	16.5	10.6	21.8	6.6	9.3	42.0	95ms/img
		✓			✓		17.1	54.1	17.7	11.9	23.5	7.7	10.2	41.8	89ms/img
✓		✓			✓		16.6	51.9	17.4	11.3	22.1	6.9	9.8	41.0	88ms/img
✓	✓	✓	✓		✓		16.4	50.8	16.7	11.2	22.7	7.5	9.3	40.8	90ms/img
		✓	✓	✓	✓	✓	13.1	49.0	12.7	8.5	18.6	5.8	7.1	38.6	91ms/img
✓		✓		✓	✓	✓	13.7	48.6	13.2	9.4	17.9	6.2	7.9	38.8	90ms/img
✓	✓	✓	✓	✓	✓		14.2	51.7	14.6	9.9	20.6	7.0	7.8	39.1	91ms/img
✓	✓	✓	✓	✓		✓	16.1	52.8	15.4	10.5	25.8	8.9	8.0	44.2	91ms/img
✓	✓	✓	✓	✓	✓	✓	11.3	47.6	11.1	7.7	17.4	6.3	6.3	36.4	91ms/img

TABLE III

DETECTION PERFORMANCE AND TEST TIME OF PROPOSED METHOD WITH DIFFERENT IMAGE SIZES ON CITYPERSONS DATASET

Proposed Method	MR(%)								Test Time
Test Image Size	Reasonable	Heavy	Partial	Bare	Small	Medium	Large	All	
2048x1024	14.5	50.6	13.9	10.6	17.0	4.8	10.2	37.1	161ms/img
1800x900	12.9	47.7	12.3	9.0	17.4	5.3	7.5	36.5	137ms/img
1600x800	10.9	47.5	10.8	7.4	17.6	6.2	6.2	36.2	110ms/img
1400x700	13.4	48.6	13.2	8.9	20.8	6.0	7.0	39.0	90ms/img
1200x600	16.4	52.7	17.1	10.5	29.9	8.2	7.0	44.3	73ms/img
1024x512	20.8	55.4	21.6	13.2	42.5	10.7	8.0	47.7	56ms/img

TABLE IV

DETECTION PERFORMANCE AND TEST TIME COMPARISON WITH STATE-OF-THE-ARTS ON CALTECH DATASET

Method	MR(%)			GPU	Test Size	Test Time
	Reasonable	Heavy	All			
Faster R-CNN	8.7	54.6	62.6	-	-	
RPN+BF	7.3	54.6	59.9	K40	960 × 720	500ms/img
SDS-RCNN	6.4	-	-	Titan X	960 × 720	210ms/img
GDFL	6.3	-	-	1080Ti	640 × 480	50ms/img
CA-GDFL	6.0	-	-	1080Ti	640 × 480	40ms/img
ALFNet	6.1	51.0	59.1	1080Ti	640 × 480	50ms/img
HyperLearner	5.5	48.7	61.5	-	640 × 480	-
RepLoss	5.0	47.9	59.0	-	640 × 480	-
CSP	4.5	45.8	56.9	1080Ti	640 × 480	60ms/img
Ours	8.9	52.7	58.4	1080Ti	640 × 480	38ms/img
	5.5	38.4	50.5	1080Ti	544 × 408	32ms/img

demonstrate that the proposed method can achieve competitive performance and inference efficiency than state-of-the-art methods and can serve as a strong baseline for future research.

The visualization comparisons of vanilla RPN, our re-trained SDS-RPN and Ours-SSAM-RPN are shown in Fig. 10. It can be observed that Ours-SSAM-RPN consistently obtains better confidence on both pedestrians. More concretely, the visibility of the right pedestrian is only 0.162, which means most of its body is occluded. SDS-RPN obtains confidence with 0.921 which is slightly higher than vanilla RPN, while Our-SSAM-RPN obtains confidence with 0.999. This demonstrates the robustness of Our-SSAM-RPN for detecting heavily occluded pedestrians.

2) Caltech Dataset: Table IV shows the detection performance and test time comparisons with state-of-the-art methods on Caltech dataset. It can be observed that the proposed method achieves slightly higher MR than RepLoss and CSP on Reasonable setup. However, our proposed method obtains better performance with large margin when evaluated on Heavy and All setups, which are 7.4% and 5.4% lower than CSP. In addition, our proposed method can run about 2X faster than CSP on the same platform. These performance achievement and test time efficiency further demonstrate the advantages of the proposed method over state-of-the-art methods.

D. Ablation Study

In this sub-section, we discuss the ablative analysis of the proposed method on CityPersons dataset.

1) Group Convolution (GC): GC is exploited in classification and bounding box regression branch of RPN head which is labelled as GC in Fig. 1. As shown in Table II, GC leads to lower computational complexity, where test time with only GC checked reduces from 95ms per image to 89 ms per image, but at the cost of slight performance degradation.

2) Semantic Segmentation to Feature Module (SS2FM): When SS2FM is applied, the MR is lower than the case when only GC is applied in most evaluation setups as shown in Table II, especially for the Heavy occlusion setup. This indicates that the proposed SS2FM can learn better convolutional features for proposal generations.

3) Semantic Segmentation to Confidence Module (SS2CM): From Table II, it can be observed that MR becomes much lower when SS2CM is exploited. When only components in Head_C5 are applied, the MR is 13.1% on Reasonable setup which is about 3.1% lower than vanilla RPN. When evaluating on Heavy occlusion setup, the performance gain is about 4.2% over vanilla RPN. When GC and SS2CM is used, the performance becomes worse compared to the case where only Head_C5 is employed. This implies that SS2FM plays an important role to improve detection performance. As shown in last row of Table II, further performance gain is obtained when all of the proposed components are exploited. In particular, the MR value of proposed SSAM-RPN is about 4.9%, 5.6%, 5.4% and 5.6% lower than vanilla RPN on Reasonable, Heavy occlusion, Partial occlusion and All setup respectively. We also list the MR when only classification or semantic segmentation confidence is used as pedestrian proposal confidence in Table II. It can be observed that MR with only semantic checked in the Confidence column becomes much higher on all evaluation setups than fused confidence, especially on Small setups. This is due to the fact that semantic segmentation confidence for small pedestrians are easily affected by larger pedestrian regions in the semantic segmentation result, since high semantic segmentation confidence would be extracted from inner larger pedestrian regions. The latter are actually false detections for small pedestrians.

4) Test Image Size: Image upscaling during testing has been used as a simple strategy to improve detection performance in state-of-the-art pedestrian methods including Faster R-CNN, Faster R-CNN+Seg, OR-CNN, RepLoss, CSP and CSID. Table III lists the detection performance and test time of the proposed method with different test image size. It can be observed that the lowest MR is achieved at image size of 1600 $\times$ 800 in most of evaluation setups. Although larger image size is used, the MR with image size of 2048 $\times$ 1024 is much higher than MR with image size of 1600 $\times$ 800, which implies that the semantic segmentation feature map is more accurate for small pedestrians and hence better semantic segmentation confidence is obtained. This can be verified from Large setup in Table III where the MR with image size of 2048 $\times$ 1024 is about 4.2% higher than MR with image size of 1600 $\times$ 800, while it is only 0.6% and 1.4% higher on Small and Medium setups. This also implies that larger input image is required for achieving better detection performance for Small and Medium pedestrians. It is also worth noting that the downsample ratio of detection feature maps is 1/8, which means that there are at most 10 pixels height for Medium pedestrians in the obtained semantic segmentation results that makes it challenging to extract meaningful semantic segmentation confidence. When using smaller image size such as 1024 $\times$ 512, the performance drop on Small setup is much larger than on Medium and Large setups, as more pedestrian details are lost for small pedestrians. In addition, the pedestrians in Small setup (i.e., pedestrian height $\in$ [50, 75] in original image) has only semantic support region with pixel height $\in$ [3.1, 4.7] in semantic segmentation feature map, which is too small for extracting useful semantic segmentation confidence.

5) Anchor Matching Point: The detection performance with different matching point between feature map and anchors are shown in Table VI. d represents the distance between anchor upper boundary and matching point. It can be observed that the best detection performance is achieved when d is set to $0.25h$, which is the anchor upper center. The commonly used matching point, i.e., $d = 0.5h$, obtains much worse performance than our setting, especially for heavily occluded pedestrians.

6) R-CNN Pooling: Table V shows the MR of the proposed method with different pooling setups when extracting features for R-CNN. The digit with ROIAlign is the sample ratio of ROIAlign pooling [17]. It can be observed that obvious performance gains are obtained on all evaluation setups when ROIAlign pooling is exploited. This indicates that some discriminative information are lost in ROI pooling operation, especially for small pedestrians, which is the reason for the pooling bin collapse described in [46].

7) Influence of Semantic Segmentation Results: Since we use weakly box-wise annotations to supervise the task of semantic segmentation, there are some prediction errors in semantic segmentation results as shown in Fig. 5b. Intuitively, the perfect semantic segmentation result should have positive influence on detection performance. In order to verify this assumption, we replace the learned semantic segmentation result using ground truth (GT) semantic segmentation with box-wise annotations in the SS2CM. The corresponding detection performance of our SSAM-RPN and Ours are listed in Table VII. It can be observed that all detection with GT semantic segmentation improves except for SSAM-RPN in Large setup. This may be caused by the problem of misalignment when resizing GT semantic segmentation. The Small setups have the largest performance improvement when using GT semantic segmentation. This is because in the learned semantic segmentation results, the Small pedestrian regions are too small to provide sufficient semantic segmentation support which affects its detection performance.

TABLE V

DETECTION PERFORMANCE AND TEST TIME OF PROPOSED METHOD WITH VARIED POOLING SETUPS ON CITYPERSONS DATASET

Proposed Method	MR(%)			
R-CNN Pooling	Reasonable	Heavy	Partial	All
ROI	12.6	48.8	12.4	38.1
ROIAlign_1	11.0	47.4	11.0	36.2
ROIAlign_2	11.0	47.5	10.8	36.2
ROIAlign_4	10.9	47.5	10.8	36.2
ROIAlign_6	11.0	47.5	10.8	36.2
ROIAlign_7	11.0	47.5	10.8	36.2

TABLE VI

DETECTION PERFORMANCE OF SSAM-RPN WITH VARIED MATCHING POINT BETWEEN FEATURE MAP AND ANCHORS ON CITYPERSONS DATASET

	MR(%)			
d	Reasonable	Heavy	Partial	All
$0.0h$	14.7	49.6	14.7	39.0
$0.25h$	11.3	47.6	11.1	36.4
$0.5h$	14.4	50.4	13.2	40.66
$0.75h$	14.3	50.0	14.1	39.0

8) Backbone Network Selection: VGG [53], InceptionNet [66], ResNet [27], and Densenet [67] are well-known networks for object classification and serve as good backbone networks for object detection. To re-purpose the classification task to detection task, an additional step to construct high resolution feature maps is required. This plays an important role in the overall accuracy and computational complexity of object detection. ResNet reformulates the network layers as learning residual function with reference to the layer inputs. InceptionNet introduces factorized convolution in order to reduce the computational complexity. DenseNet proposes to connect each layer to every other layer in a feed-forward fashion which can strengthen feature propagation and feature reuse. Although these three backbone networks achieve better performance or lower computational complexity for image classification compared to the VGG-16 network, they turn out to be inferior options for the proposed pedestrian detection architecture. Our experiments show that these backbone networks would incur degraded accuracy or longer test time when incorporated in the proposed learning architecture as shown in Table VIII. Note that we have tailored ResNet-50, InceptionNet-V3 and DenseNet-169 by removing downsampling function of last two convolution blocks to ensure that the scale of the final feature map is 1/8 of the input image size, which is consistent with our tailored VGG-16 backbone network. This additional step to construct high resolution feature maps is required to re-purpose the classification task of the original models to detection task.

It can be observed from Table VIII that the proposed learning architecture with these three backbone networks

TABLE VII

DETECTION PERFORMANCE WITH GROUND TRUTH (GT) BOX-WISE ANNOTATIONS OF PROPOSED METHOD ON CITYPERSONS DATASET

Setting		MR(%)							
		Reasonable	Heavy	Partial	Bare	Small	Medium	Large	All
Without GT Semantic Segmentation	SSAM-RPN	11.3	47.6	11.1	7.7	17.4	6.3	6.3	36.4
	Ours	10.9	47.5	10.8	7.4	17.6	6.2	6.2	36.2
With GT Semantic Segmentation	SSAM-RPN	10.8	46.5	10.5	7.4	15.0	5.5	6.7	34.6
	Ours	10.4	46.0	10.1	7.1	15.3	5.6	6.1	34.3

TABLE VIII

COMPUTATIONAL BURDEN AND TEST TIME OF CSP AND PROPOSED METHOD WITH VARIED BACKBONE NETWORKS ON CITYPERSONS DATASET. ALL BACKBONE NETWORKS ARE TAILORED TO THE PROPOSED ARCHITECTURE. THE TEST TIME IS OBTAINED ON NVIDIA 1080TI GPU

Method	Test Image Size					
	1600x800			2048x1024		
	Reasonable MR(%)	GFLOPs	Test Time	Reasonable MR(%)	GFLOPs	Test Time
CSP_ResNet-50	14.5	1475.1	223ms	11.0	2416.9	330ms
CSP_only_ResNet-50	-	159.0	91ms	-	260.5	154ms
Ours_ResNet-50	21.9	687.3	182ms	24.4	1097.1	271ms
Ours_InceptionNet-V3	22.6	587.9	159ms	22.4	953.7	224ms
Ours_DenseNet-169	21.5	446.7	219ms	22.5	701.0	332ms
Ours_VGG-16	10.9	568.1	110ms	14.5	909.3	161ms

achieved much higher MR value under Reasonable setup. This is because in the original three backbone networks, the feature map size from the last convolution block is 1/32 of the input image size. When the networks are tailored to the proposed architecture, the feature map size from the last convolution block increases to 1/8 of the input image size. This will significantly affect the feature learning capability of the networks, which were originally designed for image classification. In contrast, our tailoring strategy only influences feature computation from conv5 layers of VGG-16, in which the feature map size increases from 1/16 to 1/8 of the input image size. Thus, the tailoring strategy on VGG-16 has marginal influence on feature learning, enabling it to obtain better features for detection compared to the tailored ResNet-50, InceptionNet-V3 and DenseNet-169.

In addition to detection performance, it can be observed from Table VIII that the GFLOPs of the proposed architecture with tailored VGG-16 is lower than that with tailored ResNet-50 and InceptionNet-V3. This is because the tailoring strategy has significantly increased the complexity of ResNet-50 and InceptionNet-V3 by a factor 4x for the penultimate convolution block and 16x for the last convolution block. Since our tailoring strategy induces high computational complexity for backbone networks, one may consider other ways to obtain the feature maps generated by backbone networks for detection. We analysed the GFLOPs of CSP with ResNet-50 as backbone network and reported the results in Table VIII. It can be observed that the GFLOPs of CSP is much higher than Ours_VGG-16 even though ResNet-50 has lower computational complexity. Most of the computations come from the feature maps combination as the ResNet-50 layers contribute to only about 11% of the total computations. Furthermore, the tailored DenseNet-169 has lower GFLOPs than the tailored VGG-16 but requires longer test time. This is because the vanilla DenseNet-169 implementation requires a significant amount of GPU memory as feature maps quadratically grow with network depth [67]. A memory-efficient DenseNet-169 implementation [68] has been previously proposed to reduce its memory requirement and this is widely used in the literature. However, this implementation comes at the cost of higher computation time [69]. In addition, the tailored DenseNet-169 has many more layers (e.g., convolution and batchnorm layers) than the tailored VGG-16, and hence it requires more memory accesses leading to longer test time as reported in [70].

Our experiments and analysis conclusively show that although the VGG-16 backbone network is not the least compute intensive model for classification task, it is the best backbone network for the proposed learning architecture for pedestrian detection as it leads to lowest computational complexity, and is able to achieve good accuracy on low resolution input images.

9) The Inception Module From InceptionNet: The inception modules in InceptionNet are able to achieve high computational efficiency for image classification. However, there are some limitations when inception modules are used for pedestrian detection. Firstly, the inception module mainly focuses on using smaller spatial filters (e.g., 3×3 or 1×1) to replace large spatial filters (e.g., 5×5 or 7×7). However, the filters in VGG-16 (i.e., 3×3) are already small spatial filters. A possible way of using inception module is to employ a combination of 3×1 convolution followed by 1×3 convolution. As highlighted in [66], these asymmetric convolutions obtain good results on medium grid size (e.g., $m \times m$, where m ranges between 12 and 20). However, feature maps for CityPersons dataset is 200×100 for input image size of 1600×800. Secondly, as pointed out in [66], the asymmetric convolution does not work well on early network layers and only obtains good results by using 1×7 convolution followed by 7×1 convolution for image classification in deeper network layers. However, these kinds of filters that are used in deep network layers are not suitable for pedestrian detection. This is because global information is necessary for obtaining good result for image classification, while more detail information are required for detecting small pedestrians. For example, a 20×50 pixels pedestrian in original image size of 2048×1024 only corresponds to about 2×5 region in our setting (i.e., input image size of 1600×800). The larger

asymmetric filters in deeper network layers would introduce too much non-pedestrian information (e.g., background) and hence affect detection accuracy. Thirdly, the inception module induces much more layers, which will increase the test time due to large number of memory accesses.

V. Conclusion

In this work, we propose a unified neural network architecture to explore the semantic segmentation result for pedestrian detection. Two semantic segmentation aggregation modules, i.e., Semantic Segmentation to Feature Module and Semantic Segmentation to Confidence Module, are proposed to fully exploit features from semantic segmentation result. In addition, a simple and effective anchor matching point transfer is proposed to alleviate the problem of feature misalignment for heavily occluded pedestrians. We conduct extensive experiments to demonstrate the effectiveness of our proposed work. Our proposed pedestrian detector can achieve competitive detection performance with the highest inference efficiency on both CityPersons and Caltech datasets.

References

[1] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[2] C. Zhou and J. Yuan, "Bi-box regression for pedestrian detection and occlusion estimation," in *Proc. ECCV*, 2018, pp. 135–151.

[3] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Occlusion-aware R-CNN: Detecting pedestrians in a crowd," in *Proc. ECCV*, 2018, pp. 637–653.

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

[5] G. Brazil and X. Liu, "Pedestrian detection with autoregressive network phases," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7231–7240.

[6] W. Liu, S. Liao, W. Ren, W. Hu, and Y. Yu, "High-level semantic feature detection: A new perspective for pedestrian detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5187–5196.

[7] W. Liu, S. Liao, W. Hu, X. Liang, and X. Chen, "Learning efficient single-stage pedestrian detectors by asymptotic localization fitting," in *Proc. ECCV*, 2018, pp. 618–634.

[8] T. Song, L. Sun, D. Xie, H. Sun, and S. Pu, "Small-scale pedestrian detection based on somatic topology localization and temporal feature aggregation," 2018, *arXiv:1807.01438*. [Online]. Available: http://arxiv.org/abs/1807.01438

[9] C. Lin, J. Lu, G. Wang, and J. Zhou, "Graininess-aware deep feature learning for pedestrian detection," in *Proc. ECCV*, 2018, pp. 732–747.

[10] S. Wang, J. Cheng, H. Liu, and M. Tang, "PCN: Part and context information for pedestrian detection with CNNS," in *Proc. BMVC*, 2017, pp. 1–13.

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

[12] 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. (CVPR)*, Jun. 2016, pp. 1259–1267.

[13] P. Dollar, 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, Aug. 2014.

[14] J. Zhang, L. Lin, Y.-C. Chen, Y. Hu, S. C. Hoi, and J. Zhu, "CSID: Center, scale, identity and density-aware pedestrian detection in a crowd," in *Proc. ICCV*, 2019, pp. 1–11.

[15] 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.

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

[17] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. ICCV*, 2017, pp. 2961–2969.

[18] X. Chen, R. Girshick, K. He, and P. Dollár, "TensorMask: A foundation for dense object segmentation," 2019, *arXiv:1903.12174*. [Online]. Available: http://arxiv.org/abs/1903.12174

[19] A. Kirillov, R. Girshick, K. He, and P. Dollar, "Panoptic feature pyramid networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6399–6408.

[20] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[21] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *Proc. ECCV*, 2014, pp. 297–312.

[22] 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.

[23] X. Du, M. El-Khamy, V. I. Morariu, J. Lee, and L. Davis, "Fused deep neural networks for efficient pedestrian detection," 2018, *arXiv:1805.08688*. [Online]. Available: http://arxiv.org/abs/1805.08688

[24] G. Brazil, X. Yin, and X. Liu, "Illuminating pedestrians via simultaneous detection and segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4950–4959.

[25] P. Dollar, 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.

[26] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

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

[28] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2403–2412.

[29] R. LiKamWa, B. Priyantha, M. Philipose, L. Zhong, and P. Bahl, "Energy characterization and optimization of image sensing toward continuous mobile vision," in *Proc. 11th Annu. Int. Conf. Mobile Syst., Appl., Services (MobiSys)*, 2013, pp. 69–82.

[30] Y. Zhang and Q. Yang, "A survey on multi-task learning," 2017, *arXiv:1707.08114*. [Online]. Available: http://arxiv.org/abs/1707.08114

[31] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, 1997.

[32] C. Doersch and A. Zisserman, "Multi-task self-supervised visual learning," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2051–2060.

[33] T. Evgeniou and M. Pontil, "Regularized multi-task learning," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2004, pp. 109–117.

[34] A. Kumar and H. Daume, III, "Learning task grouping and overlap in multi-task learning," in *Proc. ICML*, 2012, pp. 1–8.

[35] J. Baxter, "A Bayesian/information theoretic model of learning to learn via multiple task sampling," *Mach. Learn.*, vol. 28, no. 1, pp. 7–39, Jul. 1997.

[36] L. Duong, T. Cohn, S. Bird, and P. Cook, "Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process.*, 2015, pp. 845–850.

[37] Y. Yang and T. M. Hospedales, "Trace norm regularised deep multi-task learning," in *Proc. ICLR*, 2017, pp. 1–4.

[38] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert, "Cross-stitch networks for multi-task learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3994–4003.

[39] L. Trottier, P. Giguère, and B. Chaib-draa, "Multi-task learning by deep collaboration and application in facial landmark detection," 2017, *arXiv:1711.00111*. [Online]. Available: http://arxiv.org/abs/1711.00111

[40] R. Kawakami, R. Yoshihashi, S. Fukuda, S. You, M. Iida, and T. Naemura, "Cross-connected networks for multi-task learning of detection and segmentation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 3636–3640.

[41] J. Cao, Y. Li, and Z. Zhang, "Partially shared multi-task convolutional neural network with local constraint for face attribute learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4290–4299.

[42] S. Ruder, "An overview of multi-task learning in deep neural networks," 2017, *arXiv:1706.05098*. [Online]. Available: http://arxiv.org/abs/1706.05098

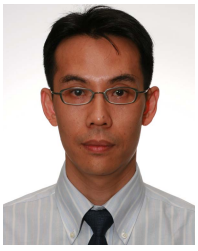
[43] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, "Facial landmark detection by deep multi-task learning," in *Proc. ECCV*, 2014, pp. 94–108.
[44] R. Ranjan, V. M. Patel, and R. Chellappa, "HyperFace: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 1, pp. 121–135, Jan. 2019.
[45] M. Marsden, K. McGuinness, S. Little, and N. E. O'Connor, "Resnetcrowd: A residual deep learning architecture for crowd counting, violent behaviour detection and crowd density level classification," in *Proc. 14th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Aug. 2017, pp. 1–7.
[46] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster R-CNN doing well for pedestrian detection?" in *Proc. ECCV*, 2016, pp. 443–457.
[47] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *Proc. ECCV*, 2016, pp. 354–370.
[48] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," 2017, *arXiv:1709.01507*. [Online]. Available: http://arxiv.org/abs/1709.01507
[49] S. Zhang, J. Yang, and B. Schiele, "Occluded pedestrian detection through guided attention in CNNs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6995–7003.
[50] C. Lin, J. Lu, G. Wang, and J. Zhou, "Graininess-aware deep feature learning for robust pedestrian detection," *IEEE Trans. Image Process.*, vol. 29, pp. 3820–3834, Jan. 2020.
[51] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3150–3158.
[52] S. Fidler, R. Mottaghi, A. Yuille, and R. Urtasun, "Bottom-up segmentation for top-down detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3294–3301.
[53] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556
[54] 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.
[55] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.
[56] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo, "ERFNet: Efficient residual factorized ConvNet for real-time semantic segmentation," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 263–272, Jan. 2018.
[57] H. Li, P. Xiong, H. Fan, and J. Sun, "DFANet: Deep feature aggregation for real-time semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9522–9531.
[58] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A deep neural network architecture for real-time semantic segmentation," 2016, *arXiv:1606.02147*. [Online]. Available: http://arxiv.org/abs/1606.02147
[59] S. Mehta, M. Rastegari, A. Caspi, L. Shapiro, and H. Hajishirzi, "ESPNet: Efficient spatial pyramid of dilated convolutions for semantic segmentation," in *Proc. ECCV*, 2018, pp. 552–568.
[60] S. Mehta, M. Rastegari, L. Shapiro, and H. Hajishirzi, "ESPNetv2: A light-weight, power efficient, and general purpose convolutional neural network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9190–9200.
[61] S. Shao *et al.*, "CrowdHuman: A benchmark for detecting human in a crowd," 2018, *arXiv:1805.00123*. [Online]. Available: http://arxiv.org/abs/1805.00123
[62] C. Change Loy *et al.*, "WIDER face and pedestrian challenge 2018: Methods and results," 2019, *arXiv:1902.06854*. [Online]. Available: http://arxiv.org/abs/1902.06854
[63] 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.
[64] S. Zhang, R. Benenson, and B. Schiele, "Filtered channel features for pedestrian detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, p. 4.
[65] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[66] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.
[67] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.
[68] G. Pleiss, D. Chen, G. Huang, T. Li, L. van der Maaten, and K. Q. Weinberger, "Memory-efficient implementation of DenseNets," 2017, *arXiv:1707.06990*. [Online]. Available: http://arxiv.org/abs/1707.06990
[69] G. Pleiss, D. Chen, G. Huang, T. Li, L. van der Maaten, and K. Q. Weinberger. (2017). *A Memory-Efficient Implementation of Densenets*. [Online]. Available: https://github.com/gpleiss/efficient_densenet_pytorch
[70] (2018). *Optimize Layers Structure of Keras Model to Reduce Computation Time*. [Online]. Available: https://github.com/ZFTurbo/Keras-inference-time-optimizer

Chengju Zhou received the M.S. degree from the School of Computer Science and Technology, Tianjin University, China, in 2015. He is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering, Nanyang Technological University, Singapore.

His current research interest includes object detection for urban traffic scene understanding.

Meiqing Wu (Member, IEEE) received the Ph.D. degree from the School of Computer Science and Engineering, Nanyang Technological University, Singapore, in 2016.

She is currently working as a Research Fellow with the School of Computer Engineering, Nanyang Technological University. Her current research interests include stereo vision, motion analysis, object detection, and tracking for urban traffic scene understanding.



Siew-Kei Lam (Senior Member, IEEE) received the B.A.Sc., M.Eng., and Ph.D. degrees from Nanyang Technological University (NTU), Singapore. He is currently an Assistant Professor with the School of Computer Engineering (SCE), NTU. He has published over 75 international refereed journals and conferences in design methodologies for heterogeneous and reconfigurable systems, embedded vision and autonomous systems, and high-speed computer arithmetic. His research interest includes realizing custom computing solutions in embedded systems.