

Weakly Supervised Adversarial Domain Adaptation for Semantic Segmentation in Urban Scenes

Qi Wang, *Senior Member, IEEE*, Junyu Gao, and Xuelong Li, *Fellow, IEEE*

Abstract—Semantic segmentation, a pixel-level vision task, is developed rapidly by using convolutional neural networks (CNNs). Training CNNs requires a large amount of labeled data, but manually annotating data is difficult. For emancipating manpower, in recent years, some synthetic datasets are released. However, they are still different from real scenes, which causes that training a model on the synthetic data (source domain) cannot achieve a good performance on real urban scenes (target domain). In this paper, we propose a weakly supervised adversarial domain adaptation to improve the segmentation performance from synthetic data to real scenes, which consists of three deep neural networks. To be specific, a detection and segmentation (“DS” for short) model focuses on detecting objects and predicting segmentation map; a pixel-level domain classifier (“PDC” for short) tries to distinguish the image features from which domains; an object-level domain classifier (“ODC” for short) discriminates the objects from which domains and predicts the objects classes. PDC and ODC are treated as the discriminators, and DS is consider as the generator. By the adversarial learning, DS is supposed to learn domain-invariant features. In experiments, our proposed method yields the new record of mIoU metric in the same problem.

I. INTRODUCTION

Semantic segmentation is a fundamental task in computer vision, which is viewed as a union of image segmentation, object localization and multi-object recognition. For the specific scenes (such as urban and indoor scenes), the task can be named as fully scene labeling/parsing, which requires to predict the label for each pixel. This paper will focus on the fully urban scenes labeling.

Recently, convolutional neural networks (CNNs) have obtained the amazing performances in the three fundamental vision tasks: image classification [1], [2], [3], object detection [4], [5], and semantic segmentation [6]. However, training CNNs requires a large amount of labeled data. Especially, for the scene labeling, annotating images for each pixel is more difficult and expensive than the other two tasks. Thus, the current pixel-wise urban datasets (such as CamVid [7] and Cityscapes [8]) contain no more than 10,000 images, which

This work was supported by the National Natural Science Foundation of China under Grant U1864204 and 61773316, Natural Science Foundation of Shaanxi Province under Grant 2018KJXX-024, and Project of Special Zone for National Defense Science and Technology Innovation.

Qi Wang, Junyu Gao and Xuelong Li are with the School of Computer Science and the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi’an 710072, China (e-mail: crabwq@gmail.com; gjy3035@gmail.com; xuelong_li@nwpu.edu.cn).

©20XX IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

is insufficient for some practical applications (e.g. self-driving cars).

In order to address the data shortage problem, some weakly supervised methods [9], [10] try to segment the image by exploiting some weak labels (image-level or object-level labels). However, they only focus on the salient foreground objects segmentation in simple scenes. In the urban scenes, the above methods cannot effectively learn discriminative features from the weakly labels because of many objects with different scales and occlusion, especially background objects (such as road, sky, building and so on). To the best of our knowledge, no algorithm tackles the labeling of full scenes via the weakly supervised learning.

In addition to the strategy at the methodology level, a potential idea is to exploit the synthetic data to prompt the performance in the real world. In recent years, some large-scale synthetic datasets [11], [12] are released, which are generated by computer graphics or crawled from some computer games. The emergence of synthetic datasets greatly emancipates manpower. Unfortunately, there exist significant domain gaps between the synthetic images and real images, including image textures, architectural styles, road materials and so on. As a result, it leads to poor performances when applying the model trained on synthetic images to real scenes. This phenomenon shows that existing supervised strategies may over learn the local discriminative features in the given training data space.

The above cross-domain (from the synthetic data to the real-world scenes) semantic segmentation attracts many researchers’ attentions. There are two unsupervised FCN-based domain adaptation methods [13], [14] to address the cross-domain problem. However, they only focus on the local pixel-level features while ignore structured object-level features in the scenes. As a matter of fact, some object-level features in the synthetic scenes are similar to that in real urban scenes, which are more robust than the pixel-level features for the cross-domain task. In general, the cross-domain generalization ability of object detection model are stronger than that of segmentation models.

Motivated by the above observation and some recent adversarial learning works and unsupervised methods [15], [16], [17], [18], in this paper, a weakly supervised adversarial domain adaptation approach is proposed to improve the segmentation performance from synthetic data (source domain) to real scenes (target domain). Figure 1 briefly shows the problem setting: the source domain needs to provide the pixel-level and object-level labels, and the target domain only provides the object-level labels.

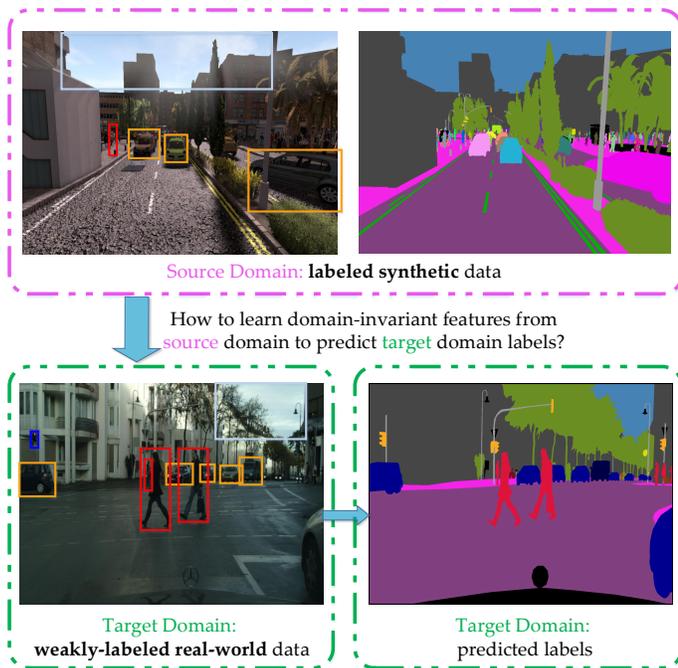


Fig. 1: Weakly supervised domain adaptation approach for semantic segmentation in real urban scenes. Given a source domain (synthetic data) with pixel/object-level labels, and a target domain (real-world scenes) with only object-level labels, our goal is train a segmentation model to predict the per-pixel labels of the target domain.

Figure 2 illustrates the entire framework. To be specific, the proposed method consists of three deep neural networks, a multi-task model for object Detection and semantic Segmentation (DS), a Pixel-level Domain Classifier (PDC) and an Object-level Domain Classifier (ODC). DS integrates a detection network and a segmentation network into one architecture. The former focuses on learning object-level features to localize the objects' bounding boxes, and the latter aims to learn local features to classify each pixel. PDC is fed with the feature maps of the segmentation network, and outputs their domain (source or target domain) for each pixel. ODC is fed with the objects features of detection network, then outputs objects category and domain class. Similar to the generative adversarial learning [19], DS model can be treated as a generator, and PDC/ODC models are regarded as two discriminators. After the adversarial training, DS model can learn domain-invariant features at the pixel and object levels to confuse PDC and ODC.

In summary, the main contributions of this paper are:

- 1) To our best knowledge, this paper is one of the first attempts to propose a weakly supervised method for fully urban scenes labeling, which employs the cross-domain problem. It can extract more robust domain-invariant features than the traditional FCN-based methods.
- 2) This paper designs two domain classifiers at the pixel/object levels to distinguish which domain the image features come from. By adversarial training, the domain gap can be effectively reduced.

- 3) The proposed method yields a new record of mIoU accuracy on the cross-domain fully urban scenes labeling.

II. RELATED WORK

In this section, we briefly review the important works about the two most related tasks: fully/weakly supervised semantic segmentation, domain adaptation with deep learning.

Semantic segmentation. In 2014, fully convolutional network (FCN) proposed by Long *et al.* [6] achieves a significant improvement in the field of some pixel-wise tasks (such as semantic segmentation, saliency detection, crowd density estimation and so on), which is a fully supervised method. After that, more and more methods [20], [21], [22], [23], [24], [25] based on FCNs are presented. Zheng *et al.* [20] propose an interpretation of dense conditional random fields as recurrent neural networks, which is appended to the top of FCN. Seg-net [21] and U-net [22] develop a symmetrical encoder-decoder architecture to prompt the performance output maps. Yu and Loltum [23] propose a dilated convolution operation to aggregate multi-scale contextual information. Zhao *et al.* [24] design a pyramid pooling module in FCN to exploit the capability of global context information. He *et al.* [26] propose a supervised multi-task learning for instance segmentation, which does not segment the background objects. Wang *et al.* [25] present a FCN to combine RGB images and contour information for road region segmentation.

Recently, some weakly-supervised methods [9], [27], [28], [29], [10] are presented to save the costs of annotating ground truth. Papandreou *et al.* [9] adopt on-line EM (Expectation-Maximization) methods training segmentation model from image-level and bounding-box labels. [27], [28] apply a progressively learning strategy to train DCNN from the image-level images. Souly *et al.* [29] apply a Generative Adversarial Networks (GANs) in which a generator network provides extra training data to a classifier. Oh *et al.* [10] exploit the saliency features as additional knowledge and mine prior information on the object extent and image statistics to segment the object regions. It is noted that the above mentioned weakly-supervised methods do not focus on labeling of full scenes. They aim to segment the salient foreground objects in the simple scenes.

Domain adaptation. There are two main streams to study domain adaptation. Some methods [30], [31], [32], [33], [15] attempt to minimize the domain gap via adversarial training. [30], [31], [32] propose a Domain-Adversarial Neural Network, which minimizes the domain classification loss. Muhammad *et al.* [33] propose an DRCN to reconstruct target domain images by optimizing a domain classifier. Tzeng *et al.* [15] present a generalized framework for adversarial adaptation, which help us understand the benefits and key ideas from GANs-based methods.

Other methods [34], [35], [36], [16] adopt the Maximum Mean Discrepancy (MMD) [37] to alleviate domain shift. MMD measures the difference between features extracted from each domain. Tzeng *et al.* [34] computes the MMD loss at one layer and Long *et al.* [35] minimizing MMD losses at multi-layer Deep Adaptation Network. Bousmalis

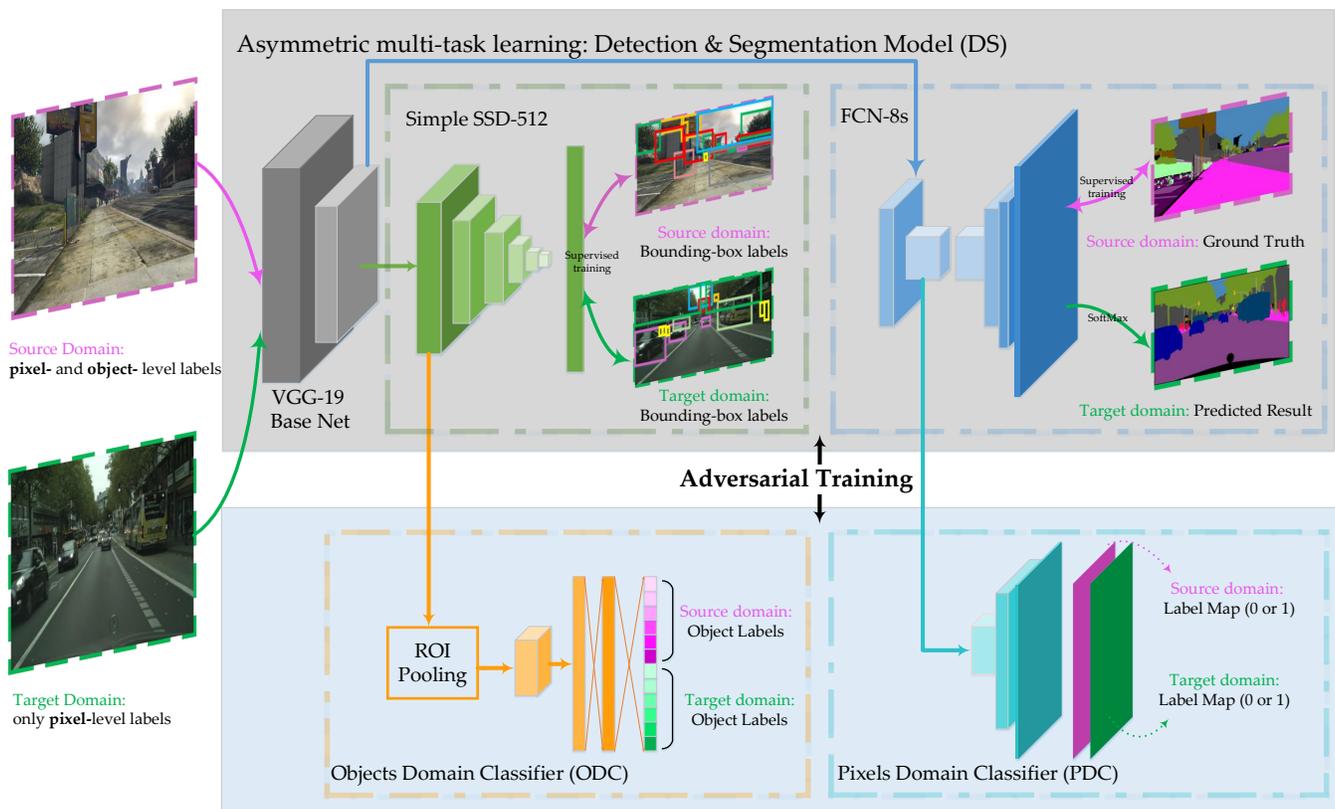


Fig. 2: The flowchart of the proposed weakly supervised adversarial domain adaptation. On the top, the asymmetric multi-task learning model is depicted, which consists of a detection model and a segmentation model (DS). During the training stage, a pair of images from two domains are fed to the DS model. The magenta and green curve arrows represent the input/output of source and target domain, respectively. Further, the two-way arrow shows that the data flow is involved in the training process. From this figure, source images take part in the object- and pixel- level training, while target images only participate in the object-level training. On the bottom, the two domain classifiers (PDC and ODC) at the object- and pixel- levels are demonstrated. The feature maps of two streams in DS are respectively fed to PDC and ODC, respectively. By alternately adversarial optimizing DS and two domain classifiers, the final DS will be obtained. During the testing phase, the test images are only fed to the segmentation stream in DS to predict the pixel-level score map.

et al. [36] propose a Domain Separation Networks (DSN) to learn domain-invariant features by explicitly separating representations private to each domain. Further, Long *et al.* [16] combines Joint Adaptation Networks (JAN) with adversarial training strategy.

Domain adaptation for semantic segmentation. Hoffman *et al.* [13] firstly propose an unsupervised domain adaptation for segmentation, which combines global and category adaptation in the adversarial learning. It effectively reduces the domain gap at the pixel level. Zhang *et al.* [14] adopt a curriculum-style domain adaption and predict global and local label distributions at image and superpixel levels, respectively.

III. APPROACH

This section describes the detailed methodology of the proposed weakly supervised adversarial domain adaptation for semantic segmentation. In order to reduce the domain gap, the inter- and intra- object features are considered in the neural network. In addition, by alternately adversarial optimizing DS and two domain classifiers (PDC and ODC), the domain gap

of learned features by DS can be alleviated effectively. Figure 2 illustrates the entire framework.

Before the detailed description, it is necessary to recall our faced cross-domain semantic segmentation by mathematical notations. A source domain \mathcal{S} from a synthetic urban dataset provides images $I_{\mathcal{S}}$, pixel-level annotations $A_{\mathcal{S}}^{pix}$, and object-level annotations $A_{\mathcal{S}}^{obj}$; and a target domain \mathcal{T} from real world provides images $I_{\mathcal{T}}$, only object-level annotations $A_{\mathcal{T}}^{obj}$. Note that \mathcal{S} and \mathcal{T} share the same label space \mathbb{R}^C , where C is the number of categories. In a word, given $I_{\mathcal{S}}$, $A_{\mathcal{S}}^{pix}$, $A_{\mathcal{S}}^{obj}$, $I_{\mathcal{T}}$ and $A_{\mathcal{T}}^{obj}$, the goal is to train a segmentation model to predict pixel-wise score map of \mathcal{T} .

Under the above definitions, the purpose of this paper is that how to reduce the domain gap between \mathcal{S} and \mathcal{T} .

A. Weak supervision for segmentation

Almost all of deep methods for semantic segmentation are based on FCN owing to its powerful learning ability. However, FCN-based methods perform not well for our faced cross-domain problems. The main reasons are that semantic segmentation is considered as a pixel-wise classification problem, and

many FCN-based methods focus on the local features (texture, color and so on) and ignore large-scale structured features. Unfortunately, the differences of texture, color or other local features are obvious in the different domains. On the contrary, the structured features and the contextual information are consistent with different domains, for instance, pedestrian posture, vehicle appearance and the position relation of objects. Thus, it is important to extract object-level features for cross-domain semantic segmentation.

Previous works [38], [26] tackle object detection and segmentation simultaneously in a single framework. However, as for the target domain with only bounding-box labels, the above supervised methods are impracticable. In this work, we propose an asymmetric multi-task learning to handle it, which consists of Detection and Segmentation streams (DS). During the training stage, a pair of images from two domains are fed to the neural network: the source images are involved in the entire model's training; the target images only participate in the detection stream's training. At the testing phase, the test images are only fed to the segmentation stream in DS to predict the pixel-wise score map. Compared with Mask RCNN [26], our model consists of two streams (shown in Fig. 2), which is an asymmetric multi-task learning on the two domains. But Mask RCNN [26] must detect the objects first and then segment them. In other word, the detection result of Mask RCNN is essential while ours is auxiliary in the test stage.

To be specific, an FCN-8s [6] is combined with a simple SSD-512 [5] into one architecture, in which the first four groups of convolutional layers are shared (named as *Base Net*). The FCN-8s aims to localize the objects' boundaries and per-pixel segmentation, and the SSD-512 focuses on learning object-level features to localize the objects' bounding boxes. Unlike the traditional detection methods, our SSD-512 can learn not only the structured objects (such as pedestrian, car, bicycle, and so on) but also some unstructured objects (e.g., road, sky, building, etc.). For the structured feature, it is an internal feature of a single object. For example, usually, the pedestrian has one head, two arms, two legs and so on, and these parts present a certain position distribution. Similarly, other objects (cars, truck, traffic sign/light) have specific structured features. About the these large unstructured objects, they contain more contextual information, which is a type of intra-object features. For example, the building is usually located under the sky in urban images, and the rectangular road region may cover the part of vehicles, pedestrians, sidewalks. Similar object relations can be regarded as a type of inter-object feature.

The proposed DS model is trained through following loss:

$$\begin{aligned} \mathcal{L}_{DS} = & \mathcal{L}_{seg}(I_S, A_S^{pix}) \\ & + \mathcal{L}_{det}(I_S, A_S^{obj}) + \mathcal{L}_{det}(I_T, A_T^{obj}), \end{aligned} \quad (1)$$

where $\mathcal{L}_{seg}(I_S, A_S^{pix})$ is 2D Cross Entropy Loss, the standard supervised pixel-wise classification objective. $\mathcal{L}_{det}(I_S, A_S^{obj})$ and $\mathcal{L}_{det}(I_T, A_T^{obj})$ are MultiBox objective loss functions [5] for the detection task, which is a weighted sum of the localization loss and the confidence loss.

B. Adversarial domain adaptation

Although the proposed weak supervision learns some domain-invariant features (including the structured intra-object feature and the contextual inter-object feature), other domain gap (such as texture, color and so on) is still not alleviated. These differences between synthetic and real-world domains are inherent. For the traditional supervised deep learning, the trained model only learns the discriminative features according to given labeled synthetic data. However, there is a problem that the learned discriminative features are not universal for real-world data.

Adversarial learning [15] provides a good framework to tackle the above problem, which pits two networks against each other. On the one hand, a domain classifier is trained to distinguish which domain the learned features are from. On the other hand, the original main model is supposed to learn not only the discriminative features to label scenes but also the domain-invariant features to confuse the domain classifier. By alternately training the two models, the extracted features from main model are invariant with respect to the domain gap.

In this paper, the Pixel-level and Object-level Domain Classifiers (PDC and ODC) are designed as the discriminators, and DS is treated as the generator in the GAN theory. Through the adversarial training, DS is supposed to learn domain-invariant features to confuse PDC and ODC.

C. Pixel-level adaptation

Since basic labeling unit of semantic segmentation is the pixel, correspondingly, a pixel-level domain classifier (PDC) is built to distinguish domain source (source domain or target domain) for each pixel. It receives the feature inputs from the segmentation stream in DS and outputs 2-channel score map with the original image's size to represent the confidence scores of per-pixel domain classes. To be specific, it consists of a convolutional layer and two de-convolutional layers. The bottom-right sub figure in Fig. 2 shows the network architecture of PDC.

Given the feature input, the PDC loss is computed as follows:

$$\begin{aligned} \mathcal{L}_{PDC} = & - \sum_{O_S^{seg} \in \mathcal{S}} \sum_{h \in H} \sum_{w \in W} \log(p(O_S^{PDC})) \\ & - \sum_{O_T^{seg} \in \mathcal{T}} \sum_{h \in H} \sum_{w \in W} \log(1 - p(O_T^{PDC})), \end{aligned} \quad (2)$$

where O_S^{PDC} and O_T^{PDC} are pixel-wise 2D-channel score map with size of $H \times W$ for source and target feature inputs, H and W denote the height and width of the original image, and $p(\cdot)$ is the soft-max operation for each pixel.

At the same time, here, the inverse of PDC loss, $\mathcal{L}_{PDC_{inv}}$ is defined as:

$$\begin{aligned} \mathcal{L}_{PDC_{inv}} = & - \sum_{O_S^{seg} \in \mathcal{S}} \sum_{h \in H} \sum_{w \in W} \log(1 - p(O_S^{PDC})) \\ & - \sum_{O_T^{seg} \in \mathcal{T}} \sum_{h \in H} \sum_{w \in W} \log(p(O_T^{PDC})). \end{aligned} \quad (3)$$

However, optimizing Eq. (2) and Eq. (3) are prone to oscillation. In fact, during the practical training phase, a domain confusion objective [30] is adopted to replace Eq. (3), which is defined as below:

$$\hat{\mathcal{L}}_{PDC_{inv}} = \frac{1}{2}(\mathcal{L}_{PDC} + \mathcal{L}_{PDC_{inv}}). \quad (4)$$

Finally, the objectives are written as follows:

$$\min_{\theta_{PDC}} \mathcal{L}_{PDC}, \quad (5)$$

$$\min_{\theta_{DS}} \mathcal{L}_{DS} + \hat{\mathcal{L}}_{PDC_{inv}}, \quad (6)$$

where θ_{PDC} and θ_{DS} denote the network parameters of PDC and DS, respectively. During the training stage, the parameters of the two models are updated in turns by minimizing Eq. (5) and Eq. (6). To be specific, a) fix θ_{DS} , and update θ_{PDC} by optimizing Eq. (5); b) fix θ_{PDC} , and update θ_{DS} by optimizing Eq. (6).

D. Object-level adaptation

In Section III-A, the object detection task is introduced in the segmentation network. Naturally, we also think modeling an object-level domain classifier (ODC) is important to extract domain-invariant features. The goal of ODC is distinguishing the object features belong to which category and come from which domain. As for some traditional domain classifiers, they only need to distinguish the data source. Here, the proposed ODC can classify the objects class, which also guides the SSD-512 can more easily learn discriminative object features.

For getting the accurately object features from the feature maps of input images, the ROI (region of interest) pooling operation [39] is a good choice. Note that the location information in ROI pooling is provided by the ground truth. In SSD-512, the filters of different layers are sensitive to the objects with different scales. Especially, the several top layers' spatial outputs are very small (16×16 , 8×8 , 4×4 and 2×2) so that ROI pooling cannot accurately extract the object features. Thus, we select the feature map with $H \times W$ of 32×32 to extract the object features.

After the ROI pooling, object features with the same size are fed to ODC, which is a simple classification network. In order to classify the category and domain simultaneously, the last feature vector is mapped into a $2 \times N$ -D confidence vector by the linear operation. N is the number of object classes. The items of $1 \sim N$ and $(N + 1) \sim N * 2$ in the confidence vector represent the scores of N classes in source domain and target domain, respectively. The bottom-left sub figure in Figure 2 describes the network design of ODC.

In ODC, each label is a one-hot vector. For the clearer expression of each label, it is necessary to formulate the one-hot vector. As for the N -D one-hot vector $Y_N(c) = [y_1, y_2, \dots, y_N]$, each component is defined as follows:

$$y_i = \begin{cases} 1, & \text{if } i = c \\ 0, & \text{otherwise} \end{cases}. \quad (7)$$

Then, the labels definitions are reported in ODC as below. To be specific, for an object with class c from the source

domain, a one-hot vector $A_S^c = Y_{2N}(c)$ is generated as the label. Similarly, the label of target domain is $A_T^c = Y_{2N}(N + c)$. Finally, our goal is optimizing the ODC loss as below:

$$\mathcal{L}_{ODC} = CEL(p(O_S^{ODC}), A_S^c) + CEL(p(O_T^{ODC}), A_T^c), \quad (8)$$

where O_S^{ODC} and O_T^{ODC} denote the score vector for each object feature, $p(\cdot)$ is the soft-max operation for each pixel, CEL function is the standard Cross Entropy Loss.

At the same time, the inverse of ODC loss should be computed to guide SSD-512 to learn domain-invariant features. To be specific, the inverse labels of the both are defined as follows: $A_{S_{inv}}^c = Y_{2N}(N + c)$ and $A_{T_{inv}}^c = Y_{2N}(c)$, and the inverse of ODC loss, $\mathcal{L}_{ODC_{inv}}$ is defined as:

$$\mathcal{L}_{ODC_{inv}} = CEL(p(O_S^{ODC}), A_{S_{inv}}^c) + CEL(p(O_T^{ODC}), A_{T_{inv}}^c). \quad (9)$$

In order to avoid the oscillation, the domain confusion objective similar to Eq. (4) are used:

$$\hat{\mathcal{L}}_{ODC_{inv}} = \frac{1}{2}(\mathcal{L}_{ODC} + \mathcal{L}_{ODC_{inv}}). \quad (10)$$

Given Eq. (8) and Eq. (10), similar to Section III-C, by iteratively optimizing ODC and DS, the final DS is obtained.

Overall, for the full models (including DS, PDC and ODC) training, the objectives are written as follows:

$$\min_{\theta_{PDC}} \mathcal{L}_{PDC}, \quad (11)$$

$$\min_{\theta_{ODC}} \mathcal{L}_{ODC}, \quad (12)$$

$$\min_{\theta_{DS}} \mathcal{L}_{DS} + \hat{\mathcal{L}}_{PDC_{inv}} + \hat{\mathcal{L}}_{ODC_{inv}}, \quad (13)$$

where θ_{PDC} , θ_{ODC} and θ_{DS} denote the network parameters of PDC, ODC and DS, respectively. During the training stage, the parameters of the three models are updated in turns by minimizing Eq. (11), Eq. (12) and Eq. (13). To be specific, a) fix θ_{DS} , simultaneously update θ_{PDC} and θ_{ODC} by optimizing Eq. (11) and (12); b) fix θ_{PDC} and θ_{ODC} , simultaneously update θ_{DS} by optimizing Eq. (13).

E. Network Architecture

In this section, the connections of the three models (DS, PDC and ODC) and data flow are described in Full Model. In DS, SSD-512 is attach to the 12-th convolutional layer (namely “conv4_4*” layer) of VGG-19. It receives the 512-channel feature map with the 1/16 size of the original input. FCN-8s integrates the outputs of conv3_4*, conv4_4* and conv5_4* layers to predict the final segmentation map. In order to obtain a better performance for segmentation, some feature maps from two streams in DS are concatenated at the channel axis, which have the same height and width size. To be specific, the conv5_4*’s output and conv6_2†’s output are concatenated together. Note that “*” represents the layer name is from VGG-19 Network ¹, and “†” denotes the layer name

¹<https://gist.github.com/ksimonyan/3785162f95cd2d5fee77>

comes from SSD Network ².

As for two discriminators, PDC's input is from the feature map of conv5_4* layer, and ODC receives the pooled features by ROI pooling operation.

IV. EXPERIMENTS

In this section, we respectively report the experimental details and the results of our proposed models, and compare with some existing methods for the same problem.

A. Datasets

In order to evaluate our methods, the two popular synthetic datasets are selected: GTA5 [11] and SYNTHIA [12] as the source domain and choose the Cityscapes [8] as the target domain.

GTA5 is collected from *Grand Theft Auto V*, which is a realistic open-world computer game developed by *Rockstar Games*. It contains 24,996 scenes with image size of 1914×1052 (other abnormal resolution is 1914×1046) pixels. All scenes are generated from a fictional city of Los Santos in the game, which are based on Los Angeles in Southern California. The annotation classes are compatible with two main datasets: Cityscapes and CamVid [7]. In the experiments, the target domain is Cityscapes, so we choose the 19-class ground truth.

SYNTHIA is SYNTHetic collection of Imagery and Annotations, a large-scale collection of photo-realistic frames rendered from some virtual cities, which contains 2 image datasets and 7 video sequences, with a resolution 1280×760 . In this paper, we use a subnet of SYNTHIA, called SYNTHIA-RAND-CITYSCAPES as the source domain, of which label space is compatible with the Cityscapes. To be specific, the subnet contains 9,400 images with 13-class categories.

Cityscapes is a real-world urban scenes dataset, which are collected from 50 European cities. In the dataset, about 5,000 images, with high resolution 2018×1024 , are fine annotated at pixel level, which are divided into three subnets with numbers 2,975, 500 and 1,525 for training, validation and testing. It defines 19 common object categories in urban scenes for semantic segmentation. In this paper, all models are tested on the Cityscapes val dataset.

Bounding-box labels The above three datasets do not provide the object-level annotations. Thus, we need to generate them for DS and ODC. To be specific, the bounding boxes of background objects (sky, building, road, etc.) are collected by transforming from the pixel-wise ground truth. As for the foreground objects (such as pedestrian, bike, car, and so on), only from the per-pixel labels, the bounding boxes of some occluded objects cannot be accurately generated. Therefore, a powerful detection model is adopted, DSOD-300 [40] to detect the foreground objects, which is trained on PASCAL VOC 2007 detection dataset.

B. Evaluation and experimental setup

1) *Evaluation*: In the semantic segmentation field, a main metric is Intersection-over-Union (IoU), which is firstly proposed in PASCAL VOC [41]. Concretely,

$$IoU = \frac{TP}{TP + FP + FN}, \quad (14)$$

where TP, FP and FN are the numbers of true positive, false positive, and false negative pixels, respectively.

2) *Experimental setup*: **Implementation Details**: In DS model, the VGG-19 net [2] is adopted as the basic neural network. Base on it, the segmentation model is built like the FCN-8s [6], and the detection model is similar to SSD-512 [5]. The DS network input is the RGB image with size of 512×512 px. During the training stage, the learning rate of basic network is set as 10^{-4} , and those of the segmentation and detection streams are set as 10^{-2} . PDC and ODC's learning rates are initialized at 10^{-4} . DS, PDC and ODC are optimized by SGD, Adam and Adam, respectively.

Our stepwise experiments.

- **DS**: DS is directly trained without domain adaptation from the source domain to the target domain.
- **DS + PDC**: Based on DS model, PDC is added to the training process by the adversarial learning.
- **Full (DS + PDC + ODC)**: In addition to PDC, ODC is also added to the training process by the adversarial learning.
- **Full†**: Furthermore, the resnet-152 [3] is used to initialize the Base Net to verify the proposed method. Other settings are the same as **Full**.

Other comparison experiments.

- **FCNs in the wild (FCN Wld) [13]**: This work is the first to tackle the same problem as ours. The authors of FCN Wld propose an unsupervised adversarial domain adaptation. Note that the pre-trained model is the dilated VGG-16 [23].
- **CDA [14]**: This work is the other existing one to the best of our knowledge. The authors of CDA propose a curriculum-style domain adaptation approach to this problem. For a higher performance, the authors exploit the additional data to train an SVM for superpixel classification. Note that the pre-trained model is the VGG-19, which is the same as ours.

In the experiments, their no adaptation and final results are listed for comparison with our stepwise experiments.

C. GTA5 \rightarrow Cityscapes

Table I lists the qualitative results of some methods for the shift from GTA5 to Cityscapes, including FCN Wld [13], CDA [14] and our stepwise experiments: DS, DS+PDC, Full and Full†. The bold fonts represent the best of the corresponding column.

From the final results, we can see Full† model achieves the best result: mean IoU of 37.4%. Based on the pre-trained models with the similar learning ability, the mean IoU of Full model (33.1%) also outperforms that of FCN Wld (27.1%) and CDA (28.9%). As for the results of the three methods with no

²https://github.com/weiliu89/caffe/blob/ssid/examples/ssid/ssid_pascal.py

TABLE I: Domain adaptation from GTA5 to the Cityscapes val dataset: the comparison results of the mainstream methods and ours.

| Method % | road | sidewalk | building | wall | fence | pole | t light | t sign | veg | terrain | sky | person | rider | car | truck | bus | train | mbike | bike | mIoU |
|---------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------------|-------------|-------------|-------------|
| NoAdapt [13] | 31.9 | 18.9 | 47.7 | 7.4 | 3.1 | 16.0 | 10.4 | 1.0 | 76.5 | 13.0 | 58.9 | 36.0 | 1.0 | 67.1 | 9.5 | 3.7 | 0.0 | 0.0 | 0.0 | 21.1 |
| FCN Wld [13] | 70.4 | 32.4 | 62.1 | 14.9 | 5.4 | 10.9 | 14.2 | 2.7 | 79.2 | 21.3 | 64.6 | 44.1 | 4.2 | 70.4 | 8.0 | 7.3 | 0.0 | 3.5 | 0.0 | 27.1 |
| NoAdapt [14] | 18.1 | 6.8 | 64.1 | 7.3 | 8.7 | 21.0 | 14.9 | 16.8 | 45.9 | 2.4 | 64.4 | 41.6 | 17.5 | 55.3 | 8.4 | 5.0 | 6.9 | 4.3 | 13.8 | 22.3 |
| CDA [14] | 74.9 | 22.0 | 71.7 | 6.0 | 11.9 | 8.4 | 16.3 | 11.1 | 75.7 | 13.3 | 66.5 | 38.0 | 9.3 | 55.2 | 18.8 | 18.9 | 0.0 | 16.8 | 14.6 | 28.9 |
| Our methods: | | | | | | | | | | | | | | | | | | | | |
| DS (NoAdapt) | 65.4 | 32.4 | 68.1 | 14.5 | 24.8 | 10.5 | 4.1 | 2.0 | 81.4 | 34.6 | 76.5 | 31.1 | 0.8 | 51.6 | 16.3 | 8.7 | 0.0 | 2.6 | 0.0 | 27.7 |
| DS+PDC | 71.4 | 32.6 | 76.4 | 28.0 | 24.9 | 10.5 | 4.4 | 3.8 | 80.6 | 29.2 | 77.4 | 33.7 | 1.8 | 53.6 | 19.6 | 18.5 | 0.0 | 3.5 | 0.0 | 30.0 |
| Full | 85.3 | 43.6 | 78.5 | 28.3 | 25.2 | 10.5 | 10.5 | 6.7 | 81.4 | 33.6 | 74.3 | 36.7 | 3.0 | 73.0 | 20.2 | 13.4 | 0.0 | 4.7 | 0.0 | 33.1 |
| Full† | 89.4 | 46.4 | 78.7 | 34.0 | 26.9 | 15.6 | 11.8 | 8.5 | 81.8 | 40.5 | 78.6 | 36.4 | 7.3 | 77.9 | 31.9 | 33.9 | 0.0 | 8.4 | 2.4 | 37.4 |

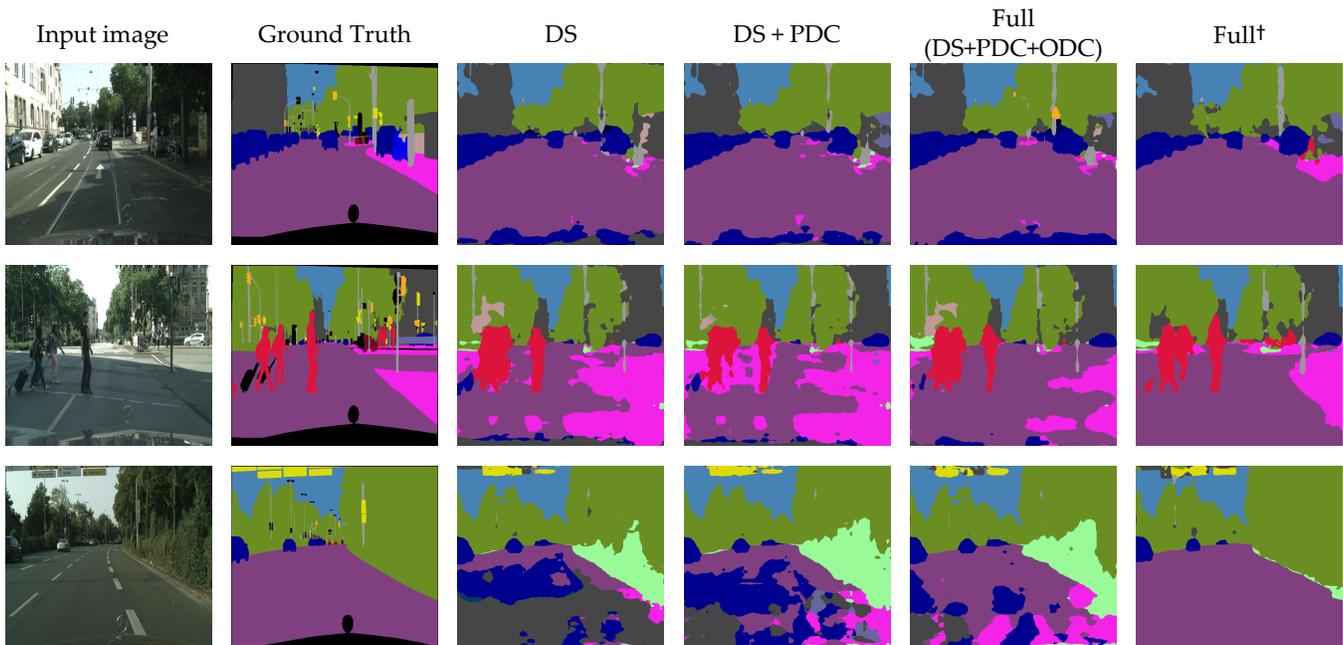


Fig. 3: Exemplar results of the Cityscapes val dataset. (Source domain: GTA5)

adaptation, DS improves the mean IoU (from 21.1/22.3% to 27.7%, increasing by 31.3/24.2%, respectively) significantly. Concretely, the performances for almost all of the categories increase remarkably, which shows the effectiveness of exploiting the bounding-box labels for semantic segmentation. At the same time, it also confirms our observation that object-level features are more robust than local pixel-level features in the cross-domain problem. According to the results of DS+PDC and Full, ODC plays a more important role in learning domain-invariant features than PDC (improvement of 3.1% versus 2.3%).

In order to analyze the semantic segmentation performance further and intuitively, Figure 3 shows the visualization results

of our step-by-step methods. The images in the first column are selected from the Cityscapes val dataset. The second column shows the ground truth, and the remaining columns illustrate the predicted labels of DS, DS+ODC, Full and Full† in turns. On the whole, After considering PDC, some segmentation mistakes are removed effectively. From the image in the 2-nd row, introducing the object-level adversarial learning, objects (such as the pedestrian) can be elaborately segmented. Based on the ResNet-152, a better segmentation result can be obtained, which shows the powerful feature learning ability of the residual network.

TABLE II: Domain adaptation from SYNTHIA to the Cityscapes val dataset: the comparison results of the mainstream methods and ours.

| Method % | road | sidewalk | building | wall | fence | pole | t light | t sign | veg | sky | person | rider | car | bus | mbike | bike | mIoU |
|---------------------|-------------|-------------|-------------|-------------|------------|-------------|------------|-------------|-------------|-------------|-------------|------------|-------------|-------------|------------|-------------|-------------|
| NoAdapt [13] | 6.4 | 17.7 | 29.7 | 1.2 | 0.0 | 15.1 | 0.0 | 7.2 | 30.3 | 66.8 | 51.1 | 1.5 | 47.3 | 3.9 | 0.1 | 0.0 | 17.4 |
| FCN Wld [13] | 11.5 | 19.6 | 30.8 | 4.4 | 0.0 | 20.3 | 0.1 | 11.7 | 42.3 | 68.7 | 51.2 | 3.8 | 54.0 | 3.2 | 0.2 | 0.6 | 20.2 |
| NoAdapt [14] | 5.6 | 11.2 | 59.6 | 0.8 | 0.5 | 21.5 | 8.0 | 5.3 | 72.4 | 75.6 | 35.1 | 9.0 | 0.0 | 0.0 | 0.5 | 18.0 | 22.0 |
| CDA [14] | 65.2 | 26.1 | 74.9 | 0.1 | 0.5 | 10.7 | 3.7 | 3.0 | 76.1 | 70.6 | 47.1 | 8.2 | 43.2 | 20.7 | 0.7 | 13.1 | 29.0 |
| Our methods: | | | | | | | | | | | | | | | | | |
| DS (NoAdapt) | 52.8 | 24.2 | 66.9 | 6.2 | 0.0 | 7.5 | 0.0 | 0.0 | 79.5 | 75.8 | 37.8 | 4.7 | 64.2 | 19.2 | 0.6 | 16.3 | 28.5 |
| DS+PDC | 71.7 | 34.6 | 74.6 | 11.0 | 0.2 | 11.6 | 0.0 | 2.9 | 79.9 | 78.6 | 39.7 | 8.6 | 55.3 | 20.5 | 0.9 | 13.7 | 31.4 |
| Full | 87.4 | 43.4 | 78.0 | 16.8 | 1.8 | 11.7 | 0.0 | 2.9 | 80.1 | 80.5 | 38.1 | 8.1 | 0.0 | 26.2 | 1.4 | 19.7 | 35.7 |
| Full† | 90.2 | 50.2 | 76.6 | 15.9 | 0.1 | 8.6 | 0.0 | 1.2 | 76.8 | 82.6 | 36.9 | 7.1 | 76.7 | 30.2 | 0.0 | 8.3 | 35.2 |

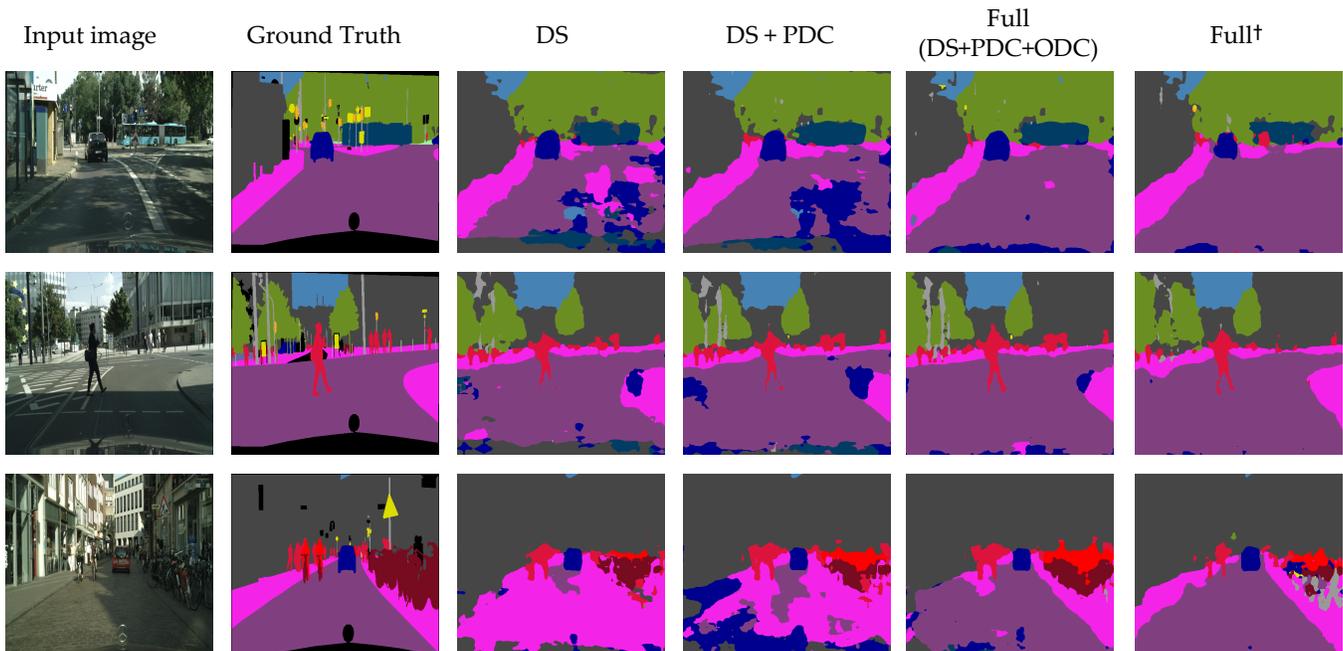


Fig. 4: Exemplar results of the Cityscapes val dataset. (Source domain: SYNTHIA)

D. SYNTHIA \rightarrow Cityscapes

The results of FCN Wld [13], CDA [14] and our stepwise experiments (DS, DS+PDC, Full and Full†) are listed in Table II, which are adapted from SYNTHIA to the Cityscapes. The bold fonts represent the best of the corresponding column. For a fair comparison with CDA [14], it is noted that the IoU performances of the three items (terrain, truck and train) are removed. This is because the three kinds of objects are not annotated in the source domain: SYNTHIA.

Similar to Section IV-C, the proposed method obtains the best performance (35.7%). Compared with the previous best method CDA (29.0%), the result of Full contributes 6.7% raw and 23.1% relative mean IoU improvement. For the three

no adaptation methods, our method prompt many objects' results greatly. In particularly, sky's segmentation IoU achieves the improvement from $\sim 6\%$ to $\sim 53\%$. According to the mean IoU of DS+PDC and Full, ODC's contribution (4.3%) is greater than PDC's (2.9%). Compared with Full, we find that the result of Full† has the slight reduction (from 35.7% to 35.2%). The main reason may be the domain gap between SYNTHIA and Cityscapes is larger than that between GTA5 and Cityscapes. Although Full† is initialized at ResNet-152, some domain gaps cannot be reduced effectively.

For reporting the advantages of our algorithms, Figure 4 shows three typical exemplar labeling results. From the visualization results of Column 5 and 6, there is little difference

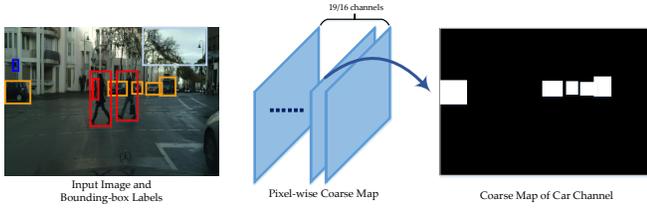


Fig. 5: The demonstration of the coarse map.

between Full and Full[†]. Some objects’ performances of Full[†] are worse than that of Full, such as the bus in Row 1 and the bicycle in Row 3. The other columns show the similar phenomenon to Figure 3.

E. Ablation Study for Bounding-box Labels

In this paper, the proposed approach exploits the bounding-box labels to train an object detector. Another treatment is: bounding-box labels can be mapped into a coarse segmentation mask, which will also promote to train a coarse semantic segmentor. For a further comparison between the object detector and the coarse segmentor, three groups of experiments with single FCN model (FCN-8s) are conducted:

- 1) training with only bounding-box labels on Cityscapes training set;
- 2) training with only per-pixel labels on GTA5 and bounding-box labels on Cityscapes;
- 3) training with only per-pixel labels on SYNTHIA and bounding-box labels on Cityscapes.

The evaluation of the above experiments is on the val set of Cityscapes. In the three experiments, it is noted that the bounding-box labels represent the coarse maps generated by the bounding-box labels. Specifically, the generation process of coarse maps is explained as below: it is a 19- or 16-channel tensor corresponding to the number of categories in two adaptation experiments (namely GTA5 \rightarrow Cityscapes and SYNTHIA \rightarrow Cityscapes). Each channel with size of input image is a mask for the corresponding category. Figure 5 illustrates the generation process.

For the first experiment, semantic segmentation is a single-label task (each pixel only has a single label), which outputs the exclusive result. However, because of the overlapped bounding-box labels, the generated rough labels are overlapped. Thus, the above experiments are treated as a multi-label task (each pixel has multiple labels) during the training phase. For comparison with proposed method, the best score from the multi-label outputs are selected as single-label prediction. As for the last two experiments, the FCN has two prediction operations, namely single label on source domain and multi labels on target domain.

Table III reports the results of the above three groups experiments and the proposed DS model. The DS and Full are our proposed method, which are explained in Section IV-B2. “City” is shortened form of Cityscapes dataset. From the results of single FCN #1-x, given the bounding-box labels, the FCN model can learn coarse features to classify each. However, the performance is poor because of some noises in

the labels. In the second group of experiment, the DS and Full model respectively outperform the single FCN #2-1 and #2-2. Compared with the single FCN, DS exploits the bounding-box labels to train a detector on the two domains. Thus, DS can extract the structured inter- and inter-object features, which are more domain-invariant than local features via the single FCN. Besides, in the adaptation experiments, Full model is trained by the hierarchical (pixel- and object- level) adversarial learning, which is more robust than only pixel-level adversarial learning in the single FCN. The same phenomenon is shown as in the third experiments.

In summary, the proposed method that trains a detector is more effective than the single FCN in the domain adaption.

F. DS vs. Mask RCNN

From the aspect of the two paper’s purpose, Mask RCNN [26] is a supervised method for instance segmentation, which does not segment the background objects. For the aspect of architecture, Mask RCNN must detect the objects first and then segment them. Our model consists of two streams, which is an asymmetric multi-task learning on the two domains. In other word, the detection result of Mask RCNN is essential while ours is auxiliary in the test stage.

Even if there are differences, Mask RCNN and DS are both multi-task learning framework for object detection and segmentation. Thus, we conduct two groups of no adaptation experiments using the two algorithms. To be specific, train DS and Mask RCNN on synthetic dataset and test them on the real data. In the Mask RCNN experiments, we implement the code from maskrcnn-benchmark [42]. Table IV reports the results of two groups of experiments. From it, Mask RCNN outperforms the proposed DS. We think the main reason is that they have different architectures. As for the two multi-task schemes, Mask RCNN is a sequential architecture, of which segmentation module directly exploits the features from detection. DS is an asymmetric multi-task architecture, of which detection and segmentation modules only share the base features from the backbone. In general, the Mask RCNN’s sequential architecture is better than the asymmetric architecture. However, during the test phase, the latter does not need the detection but the former must firstly detect the bounding box. In terms of runtime, DS is faster than Mask RCNN.

TABLE IV: The comparison results of DS and Mask RCNN.

| Methods | Domain | | mean IoU |
|----------------|--------|--------|-------------|
| | source | target | |
| DS | GTA5 | City | 27.7 |
| Mask RCNN [26] | GTA5 | City | 29.3 |
| DS | SYN | City | 28.5 |
| Mask RCNN [26] | SYN | City | 30.1 |

G. $2 \times N$ -class ODC vs. 2-class ODC

Traditional domain discriminator only classifies the features’ sources, which is a binary classification. In our ODC,

TABLE III: The comparison results of three groups of ablation experiments and our proposed DS/Full model.

| Methods | Domain | | Adaptation | Labels | | mean IoU |
|-----------------|--------|--------|------------|-----------|-----------|----------|
| | source | target | | bbx | per-pixel | |
| single FCN #1-1 | City | | ✗ | City | ✗ | 23.0 |
| single FCN #2-1 | GTA5 | City | ✗ | src & tgt | src | 22.3 |
| single FCN #2-2 | GTA5 | City | ✓ | src & tgt | src | 28.9 |
| DS #2 | GTA5 | City | ✗ | src & tgt | src | 27.7 |
| Full #2 | GTA5 | City | ✓ | src & tgt | src | 33.1 |
| single FCN #3-1 | SYN | City | ✗ | src & tgt | src | 19.4 |
| single FCN #3-2 | SYN | City | ✓ | src & tgt | src | 24.2 |
| DS #3 | SYN | City | ✗ | src & tgt | src | 28.5 |
| Full #3 | SYN | City | ✓ | src & tgt | src | 35.7 |

we attempt to make it learn the object label and source label of each feature simultaneously. The proposed $2 \times N$ -class ODC contains more neural units in fully-connected layer than traditional 2-class ODC. Note that N denotes the number of categories in the dataset. By the supervised training, some specific units of the $2 \times N$ -class ODC strongly respond to the specific object category. Thus, the adversarial loss of the specific category cannot suffer from the effects of the other categories. For the 2-class ODC, due to lack of the supervision at the object level, it cannot learn the above ability of the $2 \times N$ -class ODC. In summary, the $2 \times N$ -class ODC provides more accurate loss than the 2-class ODC. Table V reports the results of the full models with the $2 \times N$ -class ODC and the 2-class ODC. From it, we find the mIoU of the former is better than that of the latter.

TABLE V: The comparison results of the Full models with the 2-class ODC and the $2 \times N$ -class ODC.

| Methods | Domain | | mean IoU |
|-------------------------|--------|--------|-------------|
| | source | target | |
| 2-class ODC | GTA5 | City | 31.5 |
| $2 \times N$ -class ODC | GTA5 | City | 33.1 |
| 2-class ODC | SYN | City | 34.8 |
| $2 \times N$ -class ODC | SYN | City | 35.7 |

V. CONCLUSION

In this paper, we propose a weakly supervised adversarial domain adaptation to improve the segmentation performance from synthetic data to real-world data. To be specific, a weakly supervised model for object detection and semantic segmentation is built, name as DS model, which extract more robust domain-invariant features than the traditional FCN-based methods. In addition, the pixel-/object- level domain classifiers are designed to guide the DS model to learn domain-invariant features by the adversarial learning, which can reduce the domain gap effectively. Our method outperforms all the existing method that do domain adaptation from synthetic scenes to real-world urban scenes for semantic segmentation. In the future work, we will further explore the object relations in the scenes, which is a key domain-invariant feature in the cross-domain semantic segmentation.

REFERENCES

- [1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [2] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
- [3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.
- [5] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.
- [6] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.
- [7] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.
- [8] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [9] G. Papandreou, L.-C. Chen, K. Murphy, and A. L. Yuille, "Weakly-and semi-supervised learning of a dcnn for semantic image segmentation," *arXiv preprint arXiv:1502.02734*, 2015.
- [10] S. J. Oh, R. Benenson, A. Khoreva, Z. Akata, M. Fritz, and B. Schiele, "Exploiting saliency for object segmentation from image level labels," in *2017 IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5038–5047.
- [11] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *European Conference on Computer Vision (ECCV)*, vol. 9906, 2016, pp. 102–118.
- [12] G. Ros, L. Sellart, J. Materzynska, D. Vázquez, and A. M. López, "The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *2016 IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3234–3243.
- [13] J. Hoffman, D. Wang, F. Yu, and T. Darrell, "Fcns in the wild: Pixel-level adversarial and constraint-based adaptation," *arXiv preprint arXiv:1612.02649*, 2016.
- [14] Y. Zhang, P. David, and B. Gong, "Curriculum domain adaptation for semantic segmentation of urban scenes," in *The IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2020–2030.
- [15] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2962–2971.
- [16] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, 2017, pp. 2208–2217.
- [17] Q. Wang, M. Chen, F. Nie, and X. Li, "Detecting coherent groups in crowd scenes by multiview clustering," *IEEE transactions on pattern analysis and machine intelligence*, 2018.

- [18] Q. Wang, Z. Qin, F. Nie, and X. Li, "Spectral embedded adaptive neighbors clustering," *IEEE transactions on neural networks and learning systems*, no. 99, pp. 1–7, 2018.
- [19] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.
- [20] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr, "Conditional random fields as recurrent neural networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1529–1537.
- [21] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.
- [22] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 234–241.
- [23] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.
- [24] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6230–6239.
- [25] Q. Wang, J. Gao, and Y. Yuan, "Embedding structured contour and location prior in siamesed fully convolutional networks for road detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 230–241, 2018.
- [26] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.
- [27] Y. Wei, X. Liang, Y. Chen, X. Shen, M. M. Cheng, J. Feng, Y. Zhao, and S. Yan, "Stc: A simple to complex framework for weakly-supervised semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 11, pp. 2314–2320, 2017.
- [28] B. Jin, M. V. Ortiz Segovia, and S. Susstrunk, "Webly supervised semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3626–3635.
- [29] N. Souly, C. Spampinato, and M. Shah, "Semi supervised semantic segmentation using generative adversarial network," in *The IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 5688–5696.
- [30] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko, "Simultaneous deep transfer across domains and tasks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4068–4076.
- [31] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *International Conference on Machine Learning*, 2015, pp. 1180–1189.
- [32] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, no. 59, pp. 1–35, 2016.
- [33] M. Ghifary, W. B. Kleijn, M. Zhang, D. Balduzzi, and W. Li, "Deep reconstruction-classification networks for unsupervised domain adaptation," in *European Conference on Computer Vision*. Springer, 2016, pp. 597–613.
- [34] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," *arXiv preprint arXiv:1412.3474*, 2014.
- [35] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *International Conference on Machine Learning*, 2015, pp. 97–105.
- [36] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan, "Domain separation networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 343–351.
- [37] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *Journal of Machine Learning Research*, vol. 13, no. Mar, pp. 723–773, 2012.
- [38] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *European Conference on Computer Vision*. Springer, 2014, pp. 297–312.
- [39] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.
- [40] Z. Shen, Z. Liu, J. Li, Y.-G. Jiang, Y. Chen, and X. Xue, "Dsod: Learning deeply supervised object detectors from scratch," in *The IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 1919–1927.
- [41] M. Everingham, S. A. Eslami, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *International journal of computer vision*, vol. 111, no. 1, pp. 98–136, 2015.
- [42] F. Massa and R. Girshick, "maskrcnn-benchmark: Fast, modular reference implementation of Instance Segmentation and Object Detection algorithms in PyTorch," <https://github.com/facebookresearch/maskrcnn-benchmark>, 2018, accessed: [Insert date here].



Qi Wang (M'15-SM'15) received the B.E. degree in automation and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, China, in 2005 and 2010, respectively. He is currently a Professor with the School of Computer Science and with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision and pattern recognition.



Junyu Gao received the B.E. degree in computer science and technology from the Northwestern Polytechnical University, Xi'an 710072, Shaanxi, P. R. China, in 2015. He is currently pursuing the Ph.D. degree from Center for Optical Imagery Analysis and Learning, Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision and pattern recognition.

Xuelong Li (M'02-SM'07-F'12) is a full professor with the School of Computer Science and the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, Shaanxi, P. R. China.