WUDA: UNSUPERVISED DOMAIN ADAPTATION BASED ON WEAK SOURCE DOMAIN LABELS

Shengjie Liu¹, Chuang Zhu¹*, Yuan Li², Wenqi Tang¹

¹Beijing University of Posts and Telecommunications, Beijing, China
²Peking University, Beijing, China

ABSTRACT

Unsupervised domain adaptation (UDA) for semantic segmentation addresses the cross-domain problem with fine source domain labels. However, the acquisition of semantic labels is often time-consuming, many scenarios only have weak labels (e.g., bounding boxes). When weak supervision and cross-domain problems coexist, this paper defines a new task: unsupervised domain adaptation based on weak source domain labels (WUDA). To explore solutions for WUDA, this paper proposes two intuitive frameworks and conducts comparative experiments. We observe that the two frameworks behave differently when the datasets change. Therefore, we construct datasets with a wide range of domain shifts and conduct extended experiments to analyze the impact of domain shift changes on the two frameworks. In addition, to measure domain shift, we apply the metric representation shift to urban landscape image segmentation for the first time. The source code and constructed datasets can be obtained from this link: https://github.com/bupt-ai-cz/WUDA.

Index Terms— Semantic segmentation, Weakly supervised learning, Domain adaptation, Domain shift

1. INTRODUCTION

As one of the most popular computer vision technologies, semantic segmentation has developed very mature and is widely used in many scenarios such as autonomous driving, remote sensing image recognition, and medical image processing. However, like most deep learning models, a serious problem faced by semantic segmentation models in the application is the existence of domain shift. A model that performs well in the source domain may experience catastrophic performance degradation in the target domain. For cross-domain problems, many studies on unsupervised domain adaptation can fine-tune a trained segmentation model by using self-training or adversarial training without any additional annotations of the target domain.

In deep learning tasks, massive amounts of samples are required for training to improve the robustness of the model, therefore, the annotation of large-scale datasets is another difficulty in training deep models. In the semantic segmentation task, in order to obtain pixel-wise mask annotations, it takes a lot of time for a single image (e.g., it takes 1.5 hours to label one image in the Cityscapes [1] dataset), and the annotation of the entire dataset requires huge manpower. Therefore, many computer vision datasets have only weak annotations (e.g., bounding boxes, points etc.)

When the cross-domain problem and the weak labels coexist (only source domain bounding boxes and target domain images are available), the domain shift and the weak supervision both bring a negative contribution to the pixel-level semantic segmentation. In this case, it is a challenge to achieve accurate semantic segmentation in the target domain. We define this task as Unsupervised Domain Adaptation Based on Weak Source Domain Labels (WUDA). The schematic diagram of WUDA is shown in Figure 1. Compared with traditional UDA tasks, the newly defined task requires rough source domain labels. Therefore, it is necessary to explore a framework to tackle the problems of transfer learning and weakly supervised segmentation at the same time. Furthermore, considering that domain shift is a vital factor affecting the results of WUDA algorithms, we construct datasets for research under multiple domain shifts.

In summary, this paper makes the following contributions: 1) We define a novel task: unsupervised domain adaptation based on weak source domain labels (WUDA). For this task, we propose two intuitive frameworks. 2) We benchmark typical weakly supervised semantic segmentation, unsupervised domain adaptation, and object detection algorithms under our two proposed frameworks. The best result can reach 82.6% of the UDA method with fine source domain labels. 3) We construct a series of datasets with different domain shifts and further analyze the impact of multiple domain shifts on the two frameworks. To the best of our knowledge, we are the first to use representation shift for domain shift measurement in urban landscape datasets.

2. RELATED WORK

WUDA will involve box-supervised semantic segmentation, unsupervised domain adaptation, object detection, and the assessment of domain shift. In this section, we will review these related works.

2.1. Box-Supervised Semantic Segmentation

In computer vision tasks, the semantic labeling takes far more time compared to bounding box annotations[2], and the need for time saving motivates box-supervised segmentation. BoxSup[3] implements
box-supervised semantic segmentation through iterative training: it uses MCG [4] to generate candidate segments, which are used as supervision to train a Fully Convolutional Network (FCN), then the FCN will re-segment the object in the bounding box and update the candidate segments. Work [2] uses the intersection of GrabCut [5] and MCG as supervision, and the segmentation results reach a level close to fully supervised methods. Actually, Most box-supervised methods employ GrabCut or MCG to obtain segmentation pseudo-labels for supervision.

2.2. Unsupervised Domain Adaptation
Unsupervised Domain Adaptation (UDA) for semantic segmentation is committed to solving the problem of poor model generalization caused by inconsistent data distribution in the source and target domains. Self-training (ST) and adversarial training (AT) are key schemes of UDA: ST schemes [6, 7, 8] typically set a threshold to filter pseudo-labels with high confidence on the target domain, and use the pseudo-labels to supervise target domain training; AT methods [9, 10, 11] usually add a domain discriminator to the model. The adversarial game of the segmenter and the discriminator can make the segmentation results of the source and target domains tend to be consistent.

2.3. Object Detection
Autonomous driving technology has greatly promoted the development of object detection. There are many pioneering works that can be widely used in various object detection tasks, such as some two-stage methods [12, 13, 14] that first perform object extraction, and then classify the extracted objects. Yolo [15] and SSD [16] algorithms can simultaneously achieve object extraction and classification in one network. Object detection also helps to extract bounding boxes in weakly supervised segmentation methods [17, 18].

2.4. Domain Shift Assessment
Domain shift comes from the difference between the source and target domain data. There are various factors (e.g. image content, texture, view angle etc.) that contribute to domain shift. While study [19] suggests that convolutional neural networks is more sensitive to the style (largely determined by the image texture) of the images. Actually, if it involves the calculation of image texture differences, most methods are based on the output features of the middle layer of the neural network. For example, in work [20], the metric representation shift is proposed to measure the domain shift of pathological images, and the mean value of each channel in the feature map is used as the style information of an image. In the pioneering work on image style transfer [21], the style loss is calculated using the Gram matrix of the output features of the two images.

3. CONSTRUCT DATASETS WITH DIFFERENT DOMAIN SHIFTS
In section 5, we will analyze the impact of domain shift changes on the experimental results. Therefore, we first introduce the domain shift metric representation shift for semantic segmentation and construct datasets with different representation shifts.

3.1. Representation Shift for Semantic Segmentation
Karin Stucke et al. [20] propose the metric representation shift to measure the domain shift between pathological image datasets. Representation shift is highly correlated with the classification accuracy in the target domain: a higher representation shift means lower accuracy in the target domain. In the absence of ground truth in the target domain, we can estimate whether the predictions made by a trained model are credible by computing the representation shift.

Similarly, we envision that representation shift is also suitable for measuring domain shift in semantic segmentation tasks. We carry out relevant experiments to prove this. Below follows a specific description of the representation shift used in our experiments:

Consider a semantic segmentation model with a feature extraction module and a classification module. Let $F(x) = \{f_1(x), ..., f_c(x)\}$ denote the extracted feature map of the input image $x$, where $f_c(x) \in \{\mathbb{R}^{h \times w}\}$ represents the $c$th channel of the feature map. The average value of $f_c(x)$ is denoted as $\bar{a}_c(x)$:

$$\bar{a}_c(x) = \frac{1}{h \times w} \sum_{i,j} f_c(x)_{i,j}.$$  

Let $p_{sa}^c$ denotes the continuous distribution of $a_c(x)$ values across the source domain dataset $X_s = \{x_{1}^s, ..., x_{ns}^s\}$, where $n_s$ is the number of images in source domain. Similarly, the distribution $p_{ta}^c$ of the target domain dataset $X_t = \{x_{1}^t, ..., x_{nt}^t\}$ can be obtained.

Then the representation shift $R$ for semantic segmentation is defined as follows:

$$R(X_s, X_t) = \frac{1}{c} \sum_{i=1}^{c} W(p_{sa}^c, p_{ta}^c),$$

where $W$ denotes the Wasserstein distance.

3.2. Feasibility Analysis
To verify the rationality of applying representation shift in image segmentation tasks, we use four common semantic segmentation datasets for our research: Cityscapes [1], BDD100k [22], GTAV [23] and SYNTHIA [24]. Each calculation of representation shift consists of 2 steps: 1) Choose one dataset as the source domain and use it to train a semantic segmentation model. 2) Choose another dataset as the target domain and use the trained model to extract the features of the source and target domains, then calculate the representation shift defined in Eq. 2. We use a DeepLabv3 in step 1 and do not use any data augmentation methods to ensure that the model fits the original images in the source domain.

As can be seen from Figures 3(a) and 3(b), the value of representation shift and mIoU on the target domain have a high degree of negative correlation. The pearson correlations of representation shift and mIoU reach -0.874 and -0.741 for 16 and 19 categories semantic segmentation, respectively. This conclusion is similar to that of the pathological image classification task in work [20], which proves that representation shift is also applicable to domain shift measurement for semantic segmentation.

3.3. Data Construction
For the popularly used dataset pair GTAV-Cityscapes in UDA methods, we perform different data augmentation processes on the target domain (Cityscapes) to simulate different domain shifts: 1) CycleGAN. CycleGAN [25] helps to transfer Cityscapes images into GTAV style. 2) Low frequency exchange. Perform fast Fourier transform (FFT) on the source target domain images separately and swap their low frequency parts, so that the target domain has the
low-frequency features of the source domain. 3) Color augmentation. Randomly adjust the brightness, contrast and color of the image. 4) Image filters. Use image processing software to filter images with the following effects: frosted glass, mural and poster. The constructed datasets cover a wide range of domain shifts as shown in Figure 3(c). This helps in section 5 to analyze the impact of different domain shifts on our proposed frameworks. The constructed datasets of different degrees of domain shift are shown in Figure 2.

4. FRAMEWORKS

Considering that WUDA contains both weak supervision and cross-domain problems, we first propose the framework WSSS-UDA: In order to take advantage of the fine box labels in the source domain, first perform box-supervised segmentation in the source domain, then the problem is transformed into a UDA task. The schematic diagram of the framework WSSS-UDA is shown in Figure 4(a).

However, the cross-domain process is not necessarily done on the semantic segmentation task. According to the study [15], Yolo has a strong generalization ability: when trained on natural images and tested on the artwork, Yolo outperforms the two-stage detection methods (e.g. R-CNN [12]) by a wide margin. Therefore, we implement the cross-domain process on the object detection task and propose the framework TDOD-WSSS (Figure 4(b)): First, use the bounding box labels of the source domain to train the object detection model, then predict bounding boxes on the target domain. Finally, implement box-supervised segmentation in the target domain.

We benchmark typical methods under our proposed frameworks. For weakly supervised semantic segmentation, we first use GrabCut to obtain pseudo-labels, then perform three epochs of self-training to gradually improve the accuracy of the segmentation results. For unsupervised domain adaptation, we benchmark CBST [6], enhanced IAST [26] and CPSL [27] where CBST is an ST-based method, while IAST and CPSL combine ST and AT. Our enhanced IAST optimizes the self-training process of the original IAST. For the object detection method, we adopt the widely used Yolov5 [28].

5. EXPERIMENTS

5.1. Datasets and Evaluation

GTAV [23] and Cityscapes [1] are the most widely used autonomous driving datasets in UDA methods. For WUDA, we also adopt them in our experiments. In addition, in order to simulate the real-to-real scene, we carry out further experiments on the dataset pair BDD100k-Cityscapes.

GTAV and BDD100k do not have the annotation of the bounding boxes. We use the python package scikit-image to perform class-wise object detection.

Fig. 3. Linear regression results of target domain mIoU and representation shift. The strong correlation of the two metrics justifies representation shift for semantic segmentation tasks.

(a) WSSS-UDA: Weakly Supervised Semantic Segmentation + Unsupervised Domain Adaptation

(b) TDOD-WSSS: Target Domain Object Detection + Weakly Supervised Semantic Segmentation

Fig. 4. Frameworks for WUDA. (a) WSSS-UDA first performs box-supervised semantic segmentation in the source domain, then uses UDA methods to improve the generalization ability of the model in the target domain. (b) TDOD-WSSS first trains an object detection model with source domain data, then predicts bounding boxes on the target domain and finally implements weakly supervised segmentation.

5.2. Implementation

All our experiments are implemented with Pytorch on a NVIDIA Tesla V100 with 32 GB memory. In framework WSSS-UDA, to ensure model compatibility, the model used for weakly supervised segmentation is consistent with the corresponding UDA method: CBST uses Deeplabv2 [29] with ResNet-38 [30] as the backbone and our enhanced IAST and CPSL use Deeplabv2 with ResNet-101 as the backbone. All other settings remain the same as the default setting for CBST, IAST and CPSL. In framework TDOD-WSSS, the weakly supervised segmentation step uses Deeplabv2 with ResNet-101 as the backbone. The object detection step uses a randomly initialized Yolov5. All methods in this paper are trained from scratch to prevent information leakage.

5.3. Main Results

The results of the two frameworks for WUDA are shown in Table 1 and 2. On the synthesis-to-real dataset, the results show that framework WSSS-UDA with E-IAST has a higher potential, its highest mIoU can reach 46.4%, and this value reaches 82.6% of E-IAST with fine source domain labels. WSSS-UDA with CPSL achieves close results (45.1%) to E-IAST based WSSS-UDA. However, for WSSS-UDA with CBST, the weak source domain labels bring a relatively large attenuation to the results (45.9% → 26.5%). Framework TDOD-WSSS only achieves a mIoU of 33.0%. This is because the
Table 1. Quantitative results (mIoU) of semantic segmentation on Cityscapes with weak GTA V labels (synthesis→real). The remarks behind framework WSSS-UDA represent which UDA method is used. ‘Fine label’ means that precise semantic labels are used in the source domain.

|               | Road | SW  | Build | Wall | Fence | Pole | TL  | TS  | Veg. | Terrain | Sky  | Person | Rider | Car  | Track | Bus  | Train | Motor | Bike | mIoU |
|---------------|------|-----|-------|------|-------|------|-----|-----|------|---------|------|--------|-------|------|-------|------|-------|-------|-----|-----|
| CBST (fine label) | 91.8 | 53.5 | 80.5  | 32.7 | 21.0  | 34.0 | 28.9| 20.4| 83.9 | 34.2    | 80.9 | 53.1   | 24.0  | 82.7 | 30.3  | 35.9 | 16.0  | 25.9 | 42.8 | 45.9 |
| E-IAST (fine label) | 94.6 | 65.9 | 87.6  | 41.4 | 25.8  | 36.4 | 52.1 | 54.8| 82.7 | 23.0    | 89.1 | 68.0   | 23.6  | 88.2 | 42.4  | 52.5 | 25.6  | 50.8 | 62.0 | 56.2 |
| CPSL (fine label) | 92.3 | 59.9 | 84.9  | 45.7 | 29.7  | 52.8 | 61.5 | 59.5| 87.9 | 41.5    | 85.0 | 73.0   | 35.5  | 90.4 | 48.7  | 73.9 | 26.3  | 53.8 | 53.9 | 60.8 |
| WSSS-UDA (CBST)  | 27.3 | 20.3 | 48.7  | 11.9 | 9.2   | 10.1 | 20.1| 11.3| 65.1 | 23.6    | 46.0 | 50.1   | 15.0  | 73.1 | 16.5  | 6.2  | 23.2  | 26.1 | 26.5 |
| WSSS-UDA (E-IAST) | 82.5 | 39.1 | 82.4  | 34.4 | 34.2  | 21.0 | 44.6 | 43.4| 76.7 | 20.4    | 63.4 | 63.8   | 6.5   | 83.2 | 31.7  | 47.5 | 0.0   | 46.5 | 60.0 | 46.4 |
| WSSS-UDA (CPSL)  | 67.0 | 38.2 | 64.6  | 33.9 | 35.1  | 30.8 | 48.8 | 49.4| 81.2 | 42.9    | 61.9 | 59.2   | 2.2   | 82.6 | 31.6  | 56.8 | 0.0   | 38.8 | 32.5 | 45.1 |
| TDOD-WSSS (Yolov5) | 61.9 | 21.1 | 76.3  | 28.3 | 26.8  | 6.7  | 38.2 | 28.4| 74.3 | 14.4    | 59.5 | 57.0   | 11.4  | 79.3 | 20.7  | 23.0 | 0.0   | 0.0  | 33.0 |

Table 2. Quantitative results (mIoU) of semantic segmentation on Cityscapes with weak BDD100K labels (real→real).

|               | Road | SW  | Build | Wall | Fence | Pole | TL  | TS  | Veg. | Terrain | Sky  | Person | Rider | Car  | Track | Bus  | Train | Motor | Bike | mIoU |
|---------------|------|-----|-------|------|-------|------|-----|-----|------|---------|------|--------|-------|------|-------|------|-------|-------|-----|-----|
| E-IAST (fine label) | 95.1 | 67.2 | 86.7  | 34.0 | 28.4  | 39.9 | 46.9| 52.0| 86.6 | 85.1    | 66.4 | 38.6   | 89.4  | 45.2 | 52.0  | 49.5 | 61.7  | 56.3 |
| WSSS-UDA (E-IAST) | 78.5 | 35.0 | 81.8  | 31.4 | 21.0  | 16.0 | 40.7 | 51.8| 82.3 | 27.6    | 76.1 | 60.7   | 84.7  | 35.8 | 41.1  | 44.4 | 58.7  | 46.9 |
| TDOD-WSSS (Yolov5) | 71.1 | 30.5 | 77.7  | 26.6 | 31.8  | 8.0  | 18.0 | 39.8| 78.6 | 25.0    | 59.9 | 57.0   | 30.2  | 80.2 | 48.2  | 45.1 | 0.0   | 34.2 | 54.9 | 43.0 |

Fig. 5. Visualization of Semantic Segmentation on Cityscapes with different frameworks. Framework WSSS-UDA can achieve results close to UDA methods with fine source labels. However, the results of framework TDOD-WSSS have many wrong segmented regions.

Yolov5 trained on the source domain has misdetections on the target domain. We also employ the two-stage method faster-RCNN for object detection, however, the misdetection in the target domain is catastrophic, and similar result occurs on the real-to-real dataset. Therefore, we do not implement further experiments with two-stage object detection methods.

On the real-to-real dataset, the mIoU value of framework WSSS-UDA reaches 46.9%, which is similar to that on the synthesis-to-real dataset. For framework TDOD-WSSS, the result is 43%, which is a 10 percentage point increase compared to the result on the GTAV-Cityscapes dataset. As the dataset changes, the results of the two frameworks have very different trends. According to Figure 3(a), the domain shifts of GTAV-Cityscapes and BDD100K-Cityscapes are significantly different. We can reasonably hypothesize that the two frameworks have different sensitivities to changes of domain shifts. Therefore, we conduct extended experiments to verify our hypothesis in the next subsection.

Figure 5 shows the qualitative results of semantic segmentation on Cityscapes dataset. With the support of accurate labels in the source domain, the enhanced IAST can achieve very accurate segmentation, and the segmentation error area is very small. When using bounding boxes as the source domain labels, framework WSSS-UDA also achieves generally satisfactory results, however, it is not as detailed as the fully supervised UDA. When it comes to framework TDOD-WSSS, the segmentation results become coarser and have large areas of mis-segmented regions. This is because the misdetection of Yolov5 brings more noise to the subsequent weakly supervised segmentation.

5.4. Analysis under Multiple Domain Shifts

Figure 6 shows the variation of mIoU with domain shift. When the domain shift is moderate or small, the mIoU of WSSS-UDA is more stable than that of TDOD-WSSS, and WSSS-UDA with E-IAST and CPSL can stabilize at a high level. When it comes to large domain shift, the mIoU of both frameworks drops: WSSS-UDA drops slowly and TDOD-WSSS drops significantly.

In general, the domain shift has a larger impact on TDOD-WSSS, while WSSS-UDA is less sensitive to changes in domain shift. WSSS-UDA has potential in the applications of complex target domains.

6. CONCLUSION

This paper defines a novel unsupervised domain adaptation task that requires only source domain bounding boxes for supervision. Meanwhile, we propose two intuitive frameworks for this task. The results show that by using a suitable UDA method in framework WSSS-UDA, the mIoU can reach 82.6% of UDA methods with fine source labels. In addition, we apply representation shift to semantic segmentation of urban landscapes for the first time and analyze the impact of different domain shifts on the two proposed frameworks. Experiments prove that framework WSSS-UDA is more tolerant of domain shift. However, there is still a long way to go for current methods to achieve precise segmentation on WUDA tasks. We hope more excellent solutions will be proposed to tackle WUDA in the future.

Acknowledgement This work was supported in part by the National Key R&D Program of China (2021ZD0109802), and in part by the National Natural Science Foundation of China (81972248).
References

[1] 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 Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 3213–3223.

[2] A. Khoreva, R. Benenson, J. Hosang, M. Hein, and B. Schiele, “Simple does it: Weakly supervised instance and semantic segmentation,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 876–885.

[3] J. Dai, K. He, and J. Sun, “Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation,” in Proceedings of the IEEE international conference on computer vision, 2015, pp. 1635–1643.

[4] J. Pont-Tuset, P. Arbelaez, J. T. Barron, F. Marques, and J. Malik, “Multiscale combinatorial grouping for image segmentation and object proposal generation,” IEEE transactions on pattern analysis and machine intelligence, vol. 39, no. 1, pp. 128–140, 2016.

[5] C. Rother, V. Kolmogorov, and A. Blake, “Interactive foreground extraction using iterated graph cuts,” ACM Transactions on Graphics, no. 23, p. 3, 2004.

[6] Y. Zou, Z. Yu, B. V. Kumar, and J. Wang, “Unsupervised domain adaptation for semantic segmentation via class-balanced self-training,” in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 289–305.

[7] Y. Zou, Z. Yu, X. Liu, B. V. Kumar, and J. Wang, “Confidence regularized self-training,” in The IEEE International Conference on Computer Vision (ICCV), October 2019.

[8] N. Araslanov and S. Roth, “Self-supervised augmentation consistency for adapting semantic segmentation,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 15364–15394.

[9] Y. Luo, L. Zheng, T. Guan, J. Yu, and Y. Yang, “Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2507–2516.

[10] H. Wang, T. Shen, W. Zhang, L.-Y. Duan, and T. Mei, “Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation,” in European conference on computer vision. Springer, 2020, pp. 642–659.

[11] S. Li, F. Lv, B. Xie, C. H. Liu, J. Liang, and C. Qin, “Bi-classifier determinacy maximization for unsupervised domain adaptation,” in AAAI, vol. 2, 2021, p. 5.

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Computer Vision and Pattern Recognition, 2014.

[13] R. Girshick, “Fast r-cnn,” in International Conference on Computer Vision (ICCV), 2015.

[14] 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 (NIPS), 2015.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788.

[16] 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.

[17] S. Lan, Z. Yu, C. Choy, S. Radhakrishnan, G. Liu, Y. Zhu, L. S. Davis, and A. Anandkumar, “Discoboo: Weakly supervised instance segmentation and semantic correspondence from box supervision,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 3406–3416.

[18] J. Lee, J. Yi, C. Shin, and S. Yoon, “Bbam: Bounding box attribution map for weakly supervised semantic and instance segmentation,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 2643–2652.

[19] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, “Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness,” arXiv preprint arXiv:1811.12231, 2018.

[20] K. Stacke, G. Eilertsen, J. Unger, and C. Lundström, “Measuring domain shift for deep learning in histopathology,” IEEE journal of biomedical and health informatics, vol. 25, no. 2, pp. 325–336, 2020.

[21] L. Gatys, A. S. Ecker, and M. Bethge, “Texture synthesis using convolutional neural networks,” Advances in neural information processing systems, vol. 28, 2015.

[22] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, “Bdd100k: A diverse driving dataset for heterogeneous multitask learning,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 2636–2645.

[23] S. R. Richter, V. Vinceet, S. Roth, and V. Koltun, “Playing for data: Ground truth from computer games,” in European conference on computer vision. Springer, 2016, pp. 102–118.

[24] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, “The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 3234–3243.

[25] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in Proceedings of the IEEE international conference on computer vision, 2017, pp. 2223–2232.

[26] K. Mei, C. Zhu, J. Zou, and S. Zhang, “Instance adaptive self-training for unsupervised domain adaptation,” in European conference on computer vision. Springer, 2020, pp. 415–430.

[27] R. Li, S. Li, C. He, Y. Zhang, X. Jia, and L. Zhang, “Class-balanced pixel-level self-labeling for domain adaptive semantic segmentation,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 11 593–11 603.

[28] G. Kocher, A. Stoken, J. Borovec, NanoCode012, ChristopherSTAN, L. Changyu, Laughing, A. Hogan, lorenzomalaman, tkiana, yxNONG, AlexWang1900, L. Diaconu, Marc, wanghaoyang0106, ml5ah, Doug, Hatovix, J. Poznanski, L. Yu, changyu098, P. Rai, R. Ferriday, T. Sullivan, W. Xinyu, YuriRibeiro, E. R. Claramunt, hopesala, pritul dave, and yzchen, “ultralytics/yolov5: v3.0,” Aug. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3983579

[29] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs,” IEEE transactions on pattern analysis and machine intelligence, vol. 40, no. 4, pp. 834–848, 2017.

[30] 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.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.