Domain-Agnostic Prior for Transfer Semantic Segmentation

Xinyue Huo[1,2] Lingxi Xie[2] Hengtong Hu[2,3] Wengang Zhou[1] Houqiang Li[1] Qi Tian[2]
[1]University of Science and Technology of China [2]Huawei Inc. [3]Hefei University of Technology
xinyueh@mail.ustc.edu.cn 198808xc@gmail.com huhengtong.hfut@gmail.com
{zhwg,lihq}@ustc.edu.cn tian.qi1@huawei.com

Abstract

Unsupervised domain adaptation (UDA) is an important topic in the computer vision community. The key difficulty lies in defining a common property between the source and target domains so that the source-domain features can align with the target-domain semantics. In this paper, we present a simple and effective mechanism that regularizes cross-domain representation learning with a ***domain-agnostic prior*** *(DAP) that constrains the features extracted from source and target domains to align with a domain-agnostic space. In practice, this is easily implemented as an extra loss term that requires a little extra costs. In the standard evaluation protocol of transferring synthesized data to real data, we validate the effectiveness of different types of DAP, especially that borrowed from a text embedding model that shows favorable performance beyond the state-of-the-art UDA approaches in terms of segmentation accuracy. Our research reveals that UDA benefits much from better proxies, possibly from other data modalities.*

1. Introduction

In the deep learning era, the most powerful approach for visual recognition is to train deep neural networks with abundant, labeled data. Such a data-driven methodology suffers the difficulty of transferring across domains, which raises an important research field named domain adaptation. This paper focuses on the setting of unsupervised domain adaptation (UDA), which assumes that the source domain offers full supervision but the target domain has no annotations available. In particular, we investigate semantic segmentation – provided the increasing amount of unlabeled data and the expensiveness of annotation, it becomes increasingly important to gain the ability of transferring models from a known domain (*e.g.*, labeled or synthesized data).

The goal of semantic segmentation is to assign each pixel with a class label. In the scenarios that annotations are absent, this is even more challenging than image-level prediction (*e.g.*, classification) because the subtle differences

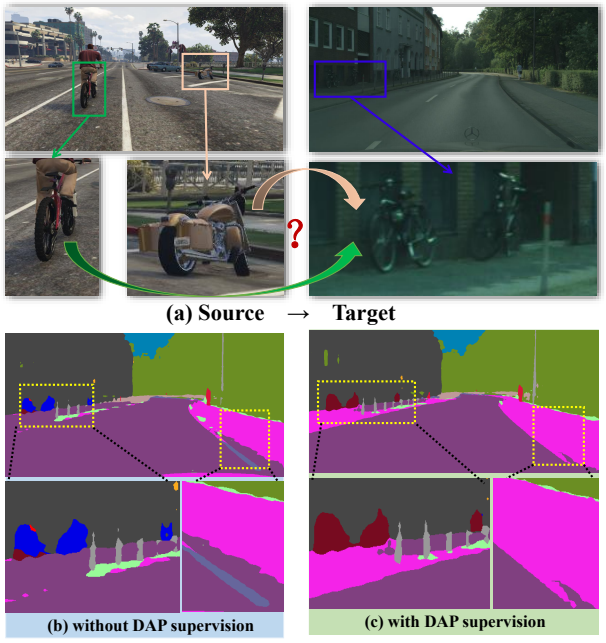

Figure 1. **Top:** The goal is to transfer a segmentation model trained in the source domain to the target domain, but some semantically similar classes (see the left of (a) for examples of *bike* and *motorbike*) are difficult to distinguish due to domain shift. **Bottom:** segmentation results of the upper-right image without and with DAP, where (b) shows incorrect segmentation (*bike*→*motorbike* and *sidewalk*→*road*) of the baseline (DACS [49]), and (c) shows how DAP improves segmentation.

at the pixel level can be easily affected by the change of data distribution. Note that this factor can bring in the risk that pseudo labels are inaccurate and thus deteriorate transfer performance, and the risk becomes even higher when the target domain lacks sufficient data for specific classes or class pairs. Fig 1 shows an example that such approaches [64, 49] are difficult in distinguishing semantically similar classes (*e.g.*, *motorbike* vs. *bike*, *road* vs. *sidewalk*) from each other.

To alleviate the above issue, we first offer a hypothesis about the confusion – the proportion of similar categories in

the two domains varies too much or they often appear adjacent to each other and the border is difficult to find (there are limited pixels around the boundary). Consequently, it is difficult for the deep network to learn the discrimination boundary based on the transferred image features. To compensate, we propose to add a **domain-agnostic prior** (DAP) to force the features from the source and target domains to align with an auxiliary space that is individual to both domains. According to the Bayesian theory, a properly designed prior relieves the instability of likelihood (provided by the limited training data with similar classes co-occurring) and leads to more accurate inference.

We implement our algorithm upon DACS, a recent approach built upon an advanced data augmentation named ClassMix [37]. The training procedure of DACS involves sampling a pair of source and target images and making use of pseudo labels to generate a mixed image with partly-pseudo labels, and feeding the source and mixed images into the deep network. We introduce DAP into the framework by defining a high-dimensional, domain-agnostic embedding vector for each class, and force the features extracted from both the source and mixed images to align with the embedding vectors through an auxiliary module. We investigate two types of domain-agnostic embedding, namely, one-hot vectors and the word2vec features [35], and show that both of them bring consistent gain in transferring. As a side note, DAP is efficient to carry out – the auxiliary module is lightweight that requires around 7% extra training computation and is removed during the inference stage.

We evaluate DAP on two standard UDA segmentation benchmarks, in which the source domain is defined by a synthesized dataset (*i.e.*, GTAv [39] or SYNTHIA [40]) while the target domain involves Cityscapes [10], a dataset captured from the real world. With the word2vec features as prior, DAP achieves segmentation mIOU scores of 55.0% and 50.2% from GTAv and SYNTHIA, respectively, with absolute gains of 2.9% and 1.3% over the DACS baseline, setting the new state-of-the-art among single-model, single-round approaches for UDA segmentation.

The main contribution of this work lies in the proposal and implementation of domain-agnostic prior for UDA segmentation. With such a simple and effective approach, we reveal that much room is left behind UDA. We expect more sophisticated priors and/or more effective constraints to be explored in the future.

2. Related Work

Unsupervised domain adaptation (UDA) aims to transfer models trained in a known (labeled) source domain to an unknown (unlabeled) target domain [15, 32, 64]. It differs from semi-supervised learning [63, 27] mainly in the potentially significant difference between the labeled and unlabeled data. The past years have witnessed a fast development of UDA , extending the studied task from image classification [42, 5] to fine-scaled visual recognition including detection [8, 25], segmentation [50, 4], person re-identification [14, 45], *etc*. Specifically, segmentation is a good testbed for UDA for at least two reasons. First, compared to classification, pixel-level segmentation requires more sophisticated manipulation of domain transfer. Second, UDA can reduce the annotation cost for segmentation which is often expensive. The recent approaches of UDA segmentation is roughly categorized into three parts, adversarial learning, self-training, and data generation.

- **Adversarial learning** is an important tool for domain adaptation [52, 2]. When it was applied to UDA segmentation [50, 48], the segmentation modules are regarded as the generator that produce prediction, and an auxiliary discriminator is trained to judge which domain the inputs are from. By confusing the discriminator, the purpose of domain adaptation is achieved. Extensions beyond this idea include [24] that facilitated the domain alignment from the input image and transferred the low-level representation (*e.g.* the texture and brightness) from the target domain to the source domain, [56] that applied a Fourier transform to narrow the low-frequency gap between the two domains, and others.
- The idea of **self-training** was borrowed from semi-supervised learning [47, 41, 21] and few-shot learning [28, 31], and was applied well to domain adaptation [44]. The key to self-training is to produce high-quality pseudo labels [27], but prediction errors are inevitable especially when the domain gap is significant. Many efforts have been made to alleviate the inaccuracy, including entropy minimization [65, 54, 7] that tried to increase the confidence on unlabeled data, class-balance regulation [64] that improved the prediction probability for hard categories and thus alleviated the burden that minor categories may be suppressed, and [17, 60] that reduced the uncertainty of prediction to rectify pseudo labels.
- The **data generation** branch tried to integrate the advantages of adversarial learning and self-training. The idea is to bridge the gap between the source and target domains using a generated domain in which source and target data are mixed. The images can be generated either by performing GAN [62, 19] or pixel-level mixing [37, 57]. Our approach follows the second path which is verified powerful in UDA segmentation [49, 16, 61], and we introduce a domain-agnostic prior to constrain representation learning on the target domain.

There have been studies of combining visual data with other types of information, in particular, with text data [23, 36, 26]. Powered by pre-training on a large amount of image and text data, either paired [38] or unpaired [22], deep neural networks gain the ability of aligning visual and linguistic features in a shared space, hence facilitating

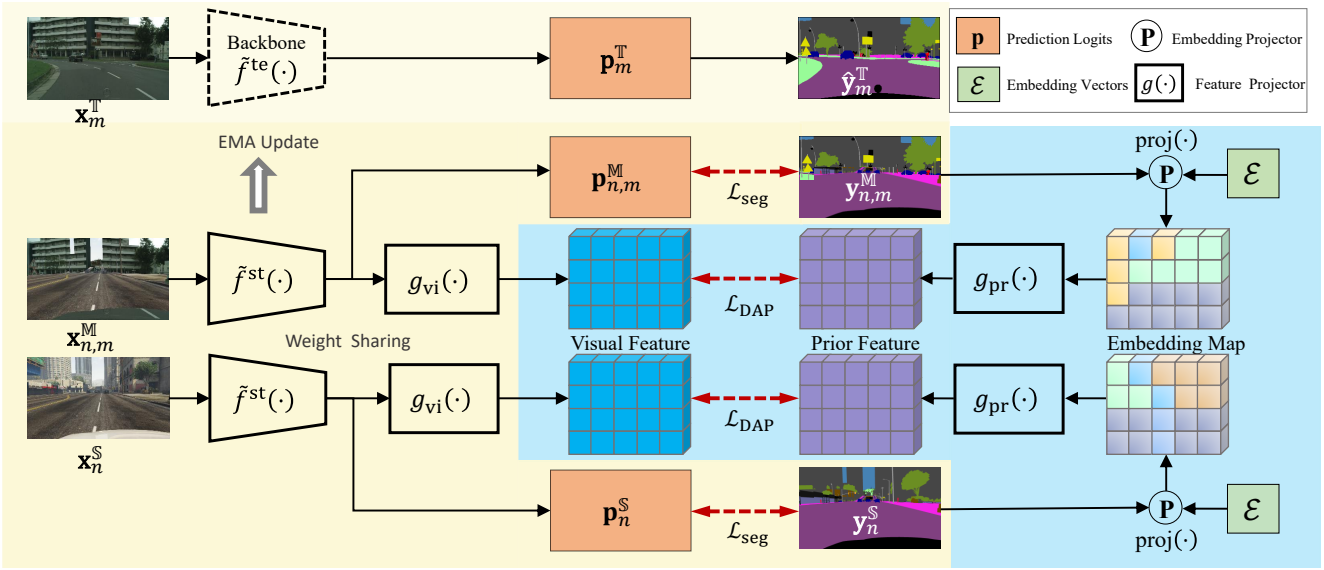

Figure 2. The proposed framework of that involves building DAP (the blue-shaded region) upon DACS [49] (the yellow-shaded region). We omit the illustration of using ClassMix [37] to generate $\mathbf{x}_{n,m}^{\mathbb{T}}$. The details of producing the embedding map (*i.e.*, the $\text{proj}(\cdot)$ function) is shown in Fig 3. *This figure is best viewed in color.*

cross-modal understanding [46, 29] and retrieval [55, 13]. Recently, the rapid development of Transformers [53, 12] brings the possibility of unifying image and text data within one framework. Transferring the learned knowledge from one modality to another may cause the setting somewhat similar to zero-shot learning [3, 59]. There are also some discussions on safe and/or efficient cross-modal learning, including [20] that introduced knowledge distillation into cross-modal retrieval, and [1] that introduced boundary-aware regression and semantic consistency constraint and improved the discrimination for the unlabeled classes.

3. Our Approach

3.1. Problem Setting and Baselines

Unsupervised domain adaptation (UDA) starts with defining two domains corresponding to different data distributions. In our setting for semantic segmentation, complete pixel-level annotations are available for the source domain, $\mathbb{S}$, but unavailable for the target domain, $\mathbb{T}$, yet we hold an assumption that the class sets on $\mathbb{S}$ and $\mathbb{T}$ are identical. From the data perspective, let the training samples on $\mathbb{S}$ from a set of $\mathcal{D}^{\mathbb{S}} = \{(\mathbf{x}_n^{\mathbb{S}}, \mathbf{y}_n^{\mathbb{S}})\}_{n=1}^{N}$ where both $\mathbf{x}_n^{\mathbb{S}}$ is a high-resolution image and $\mathbf{y}_n^{\mathbb{S}}$ of the same size offers pixel-wise semantic labels. Similarly, we have a training set on $\mathbb{T}$ being $\mathcal{D}^{\mathbb{T}} = \{(\mathbf{x}_m^{\mathbb{T}})\}_{m=1}^{M}$ where ground-truth labels are not provided. The goal is to train a model $\mathbf{y} = f(\mathbf{x}; \boldsymbol{\theta})$ on both $\mathcal{D}^{\mathbb{S}}$ and $\mathcal{D}^{\mathbb{T}}$ so that it works well on a hidden test set $\mathcal{D}^{\mathbb{T}\prime}$ which is also sampled from $\mathbb{T}$.

The major difficulty of UDA comes from the domain gap between $\mathbb{S}$ and $\mathbb{T}$, *e.g.*, in our testing environment, $\mathbb{S}$ corresponds to synthesized data [39, 40] but $\mathbb{T}$ corresponds to real-world data [10]. The potential differences in lighting, object styles, *etc.*, can downgrade prediction confidence as well as quality on the target domain. One of the popular solutions to bridge the domain gap is to use the source model to generate pseudo labels $\hat{\mathbf{y}}_m^{\mathbb{T}}$ for $\mathbf{x}_m^{\mathbb{T}}$ by a mean-teacher model [47], and uses both $\mathcal{D}^{\mathbb{S}}$ and $\hat{\mathcal{D}}^{\mathbb{T}} = \{(\mathbf{x}_m^{\mathbb{T}}, \hat{\mathbf{y}}_m^{\mathbb{T}})\}_{m=1}^{M}$, the extended target set, for updating the online student model. We denote the teacher and student models as $f^{\text{te}}(\mathbf{x}; \boldsymbol{\theta}^{\text{te}})$ and $f^{\text{st}}(\mathbf{x}; \boldsymbol{\theta}^{\text{st}})$, respectively, where in the mean-teacher algorithm, $f^{\text{te}}(\cdot)$ is the moving average of $f^{\text{st}}(\cdot)$.

The above mechanism, though simple and elegant, suffers the unreliability of $\hat{\mathbf{y}}_m^{\mathbb{T}}$. To alleviate this issue, in a recently published work named DACS [49], researchers proposed to replace the training data from the target domain with that from a mixed domain, $\mathbb{M}$, where the samples are generated by mixing images at the pixel level guided by class labels [37]. In each training iteration, a pair of source and target images with (true or pseudo) labels are sampled and cropped into the same resolution, denoted as $(\mathbf{x}_n^{\mathbb{S}}, \mathbf{y}_n^{\mathbb{S}}, \mathbf{x}_m^{\mathbb{T}}, \hat{\mathbf{y}}_m^{\mathbb{T}})$. Next, a subset of classes is randomly chosen from $\mathbf{y}_n^{\mathbb{S}}$ and a binary mask $\mathbf{M}_{n,m}$ of the same size as $\mathbf{x}_n^{\mathbb{S}}$ is made, with all pixels that $\mathbf{y}_n^{\mathbb{S}}$ belongs to the subset being 1 and otherwise 0. Upon $\mathbf{M}_{n,m}$, a mixed image with its label is defined:

$$\begin{cases} \mathbf{x}_{n,m}^{\mathbb{M}} = \mathbf{x}_n^{\mathbb{S}} \odot \mathbf{M}_{n,m} + \mathbf{x}_m^{\mathbb{T}} \odot (\mathbf{1} - \mathbf{M}_{n,m}) \\ \mathbf{y}_{n,m}^{\mathbb{M}} = \mathbf{y}_n^{\mathbb{S}} \odot \mathbf{M}_{n,m} + \hat{\mathbf{y}}_m^{\mathbb{T}} \odot (\mathbf{1} - \mathbf{M}_{n,m}) \end{cases}, \quad (1)$$

where $\odot$ denotes element-wise multiplication. The student

		A: *motor*, B: *bike*		A: *road*, B: *sidewalk*	
Methods	Data	$\cos\langle\boldsymbol{\mu}_{\mathrm{A}}^{\cdot}, \boldsymbol{\mu}_{\mathrm{B}}^{\cdot}\rangle$	IOU (%)	$\cos\langle\boldsymbol{\mu}_{\mathrm{A}}^{\cdot}, \boldsymbol{\mu}_{\mathrm{B}}^{\cdot}\rangle$	IOU (%)
Source	$\mathbb{S}$	*0.42*	*6.36*	*0.35*	*1.43*
Only	$\mathbb{T}$	0.60	27.76	0.43	5.53
DACS	$\mathbb{T}$	0.60	26.62	0.38	1.95
DAP	$\mathbb{T}$	**0.56**	**22.19**	**0.23**	**1.29**

Table 1. Statistics of features extracted two pairs of semantically similar classes, where $\mathbb{S}$ and $\mathbb{T}$ corresponds to GTAv and Cityscapes, respectively. Please refer to the main text for details.

model, $f^{\mathrm{st}}(\mathbf{x};\boldsymbol{\theta}^{\mathrm{st}})$, is trained with $(\mathbf{x}_n^{\mathbb{S}}, \mathbf{y}_n^{\mathbb{S}})$ and $(\mathbf{x}_n^{\mathbb{M}}, \mathbf{y}_n^{\mathbb{M}})$:

$$\mathcal{L}_{\mathrm{seg}} = \mathcal{L}_{\mathrm{CE}}(f^{\mathrm{st}}(\mathbf{x}_n^{\mathbb{S}};\boldsymbol{\theta}^{\mathrm{st}}), \mathbf{y}_n^{\mathbb{S}}) + \mathcal{L}_{\mathrm{CE}}(f^{\mathrm{st}}(\mathbf{x}_{n,m}^{\mathbb{M}};\boldsymbol{\theta}^{\mathrm{st}}), \mathbf{y}_{n,m}^{\mathbb{M}}), \tag{2}$$

where $\mathcal{L}_{\mathrm{CE}}(\cdot,\cdot)$ is the pixel-wise cross-entropy loss. The teacher model, $f^{\mathrm{te}}(\mathbf{x};\boldsymbol{\theta}^{\mathrm{te}})$, is updated with the student model using the exponential moving average (EMA) mechanism, namely, $\boldsymbol{\theta}^{\mathrm{te}} \leftarrow \boldsymbol{\theta}^{\mathrm{te}} \cdot \lambda + \boldsymbol{\theta}^{\mathrm{st}} \cdot (1-\lambda)$, where λ controls the window of EMA and is often close to 1.0. The entire flowchart of DACS is illustrated in the yellow-shaded part of Fig 2.

3.2. Confusion of Semantically Similar Classes

Despite the effectiveness in stabilizing self-learning, DACS still has difficulties in distinguishing semantically similar classes, especially when these classes do not appear frequently in the target domain, *e.g.*, *motorbike* accounts for only 0.1% of the total number of pixels. Fig 1 shows an example that the class pair of *motorbike* and *bike* can easily confuse the model, and so can the pair of *road* and *sidewalk*. According to experimental results, such confusion contributes significantly to segmentation error, *e.g.*, 20.8% mis-classification of *bike* pixels goes to *motorbike*.

We offer a hypothesis for the above phenomenon. Since data from the target domain, say $\mathbf{x}_m^{\mathbb{T}}$, are not labeled, the semantic correspondence is learned by mapping $\mathbf{x}_m^{\mathbb{T}}$ to the source domain, *e.g.*, by image-level style transfer for GAN-based approaches [62, 19] and by label-level simulation for DACS. Mathematically, this is to learn a transfer function (which transfers visual features from the target domain to the source domain) in a weakly-supervised manner. This causes approximation of visual representation and consequently incurs inaccuracy of recognition.

In addition, we consider two semantically similar classes denoted as A and B, and the corresponding features extracted from these classes form two sets of $\mathcal{F}_{\mathrm{A}}^{\cdot}$ and $\mathcal{F}_{\mathrm{B}}^{\cdot}$, respectively, where the superscript can be either $\mathbb{S}$ or $\mathbb{T}$. Let us assume that each feature set follows a multi-variate Gaussian distribution denoted by $\mathcal{N}(\boldsymbol{\mu}_{\mathrm{A}}^{\cdot}, \boldsymbol{\Sigma}_{\mathrm{A}}^{\cdot})$ and $\mathcal{N}(\boldsymbol{\mu}_{\mathrm{B}}^{\cdot}, \boldsymbol{\Sigma}_{\mathrm{B}}^{\cdot})$, abbreviated as $\mathcal{N}_{\mathrm{A}}^{\cdot}$ and $\mathcal{N}_{\mathrm{B}}^{\cdot}$, respectively. Since A and B are semantically similar, we assume that $\boldsymbol{\mu}_{\mathrm{A}}^{\mathbb{S}}$ and $\boldsymbol{\mu}_{\mathrm{B}}^{\mathbb{S}}$ are close in the feature space, and the source model learns to distinguish A from B by reducing $\cos\langle\boldsymbol{\mu}_{\mathrm{A}}^{\mathbb{S}}, \boldsymbol{\mu}_{\mathrm{B}}^{\mathbb{S}}\rangle$ and hence the IOU between $\mathcal{N}_{\mathrm{A}}^{\mathbb{S}}$ and $\mathcal{N}_{\mathrm{B}}^{\mathbb{S}}$ [1]. However, in the target domain, these conditions are not necessarily satisfied since no strong supervision is present – the distance between $\boldsymbol{\mu}_{\mathrm{A}}^{\mathbb{T}}$ and $\boldsymbol{\mu}_{\mathrm{B}}^{\mathbb{T}}$ gets smaller (*i.e.*, $\cos\langle\boldsymbol{\mu}_{\mathrm{A}}^{\mathbb{T}}, \boldsymbol{\mu}_{\mathrm{B}}^{\mathbb{T}}\rangle$ gets larger), and, consequently, the IOU between $\mathcal{N}_{\mathrm{A}}^{\mathbb{T}}$ and $\mathcal{N}_{\mathrm{B}}^{\mathbb{T}}$ is larger. Tab 1 offers quantitative results from transferring a segmentation model from GTAv to Cityscapes. The features of two semantically similar class pairs, namely, *motorbike* vs. *bike*, and *road* vs. *sidewalk*, are both made easier to confuse the network.

3.3. Domain-Agnostic Prior for UDA Segmentation

To obtain more accurate estimation for $\mathcal{N}_{\mathrm{A}}^{\mathbb{T}}$ and $\mathcal{N}_{\mathrm{B}}^{\mathbb{T}}$, we refer to the Bayesian theory that the posterior distribution is composed of a prior and a likelihood. In our setting, the likelihood comes from the target dataset, $\mathcal{D}^{\mathbb{T}}$, where there are insufficient data to guarantee accurate estimation. The solution lies in introducing an informative prior for the elements, namely, $\mathbf{z}_{\mathrm{A}}^{\mathbb{T}} \sim \mathcal{N}_{\mathrm{A}}^{\mathbb{T}}$ and $\mathbf{z}_{\mathrm{B}}^{\mathbb{T}} \sim \mathcal{N}_{\mathrm{B}}^{\mathbb{T}}$. This involves defining constraints as follows:

$$\mathbf{z}_{\mathrm{A}}^{\mathbb{T}} = g(\mathbf{e}_{\mathrm{A}}), \quad \mathbf{z}_{\mathrm{B}}^{\mathbb{T}} = g(\mathbf{e}_{\mathrm{B}}), \tag{3}$$

where $\mathbf{e}_{\mathrm{A}}$ and $\mathbf{e}_{\mathrm{B}}$ are **not** related to any specific domain – we name them **domain-agnostic priors** (DAP), and $g(\cdot)$ is a learnable function that projects the priors to the semantic space. In practice, it is less likely to strictly satisfy Eqn (3), so we implement it as a loss term to minimize.

We instantiate DAP into two examples. The first one is a set of one-hot vectors, *i.e.*, for a target domain with C classes, the c-th class is encoded into $\mathbf{I}_c$, a C-dimensional vector in which all entries are 0 except for the c-th dimension being 1. The second one is borrowed from word2vec [35], a pre-trained language model that represents each word using a 300-dimensional vector. Note that both cases are completely independent from the vision domain, *i.e.*, reflecting the principle of being domain-agnostic. We denote the set of embedding vectors as $\mathcal{E} = \{\mathbf{e}_c\}_{c=1}^{C}$, where $\mathbf{e}_c$ is the embedding vector of the c-th class.

Back to the main story, DAP and the embedded vectors are easily integrated into DACS, the baseline framework. Besides the segmentation loss $\mathcal{L}_{\mathrm{seg}}$ defined in Eqn (2), we introduce another loss term $\mathcal{L}_{\mathrm{DAP}}$ that measures the distance between the embedded domain-agnostic priors and visual features extracted from training images,

$$\begin{aligned}\mathcal{L}_{\mathrm{DAP}} =& \|g_{\mathrm{vi}}(\tilde{f}^{\mathrm{st}}(\mathbf{x}_n^{\mathbb{S}};\boldsymbol{\theta}^{\mathrm{st}})) - g_{\mathrm{pr}}(\mathrm{proj}(\tilde{\mathbf{y}}_n^{\mathbb{S}};\mathcal{E}))\|_2^2 + \\ & \|g_{\mathrm{vi}}(\tilde{f}^{\mathrm{st}}(\mathbf{x}_{n,m}^{\mathbb{M}};\boldsymbol{\theta}^{\mathrm{st}})) - g_{\mathrm{pr}}(\mathrm{proj}(\tilde{\mathbf{y}}_{n,m}^{\mathbb{M}};\mathcal{E}))\|_2^2,\end{aligned} \tag{4}$$

where $\tilde{f}^{\mathrm{st}}(\cdot)$ is the backbone part of $f^{\mathrm{st}}(\cdot)$ that extracts mid-resolution features (*i.e.*, visual features), $\tilde{\mathbf{y}}_n^{\mathbb{S}}$ and $\tilde{\mathbf{y}}_{n,m}^{\mathbb{M}}$ de-

[1] To calculate the IOU between $\mathcal{N}_{\mathrm{A}}^{\mathbb{S}}$ and $\mathcal{N}_{\mathrm{B}}^{\mathbb{S}}$, we sample an equal amount of points from $\mathcal{N}_{\mathrm{A}}^{\mathbb{S}}$ and $\mathcal{N}_{\mathrm{B}}^{\mathbb{S}}$, and calculate r_{A} as the probability that a point sampled from $\mathcal{N}_{\mathrm{A}}^{\mathbb{S}}$ actually has a higher density at $\mathcal{N}_{\mathrm{B}}^{\mathbb{S}}$, and r_{B} vice versa. The IOU is then approximated as $(r_{\mathrm{A}} + r_{\mathrm{B}})/(2 - r_{\mathrm{A}} - r_{\mathrm{B}})$.

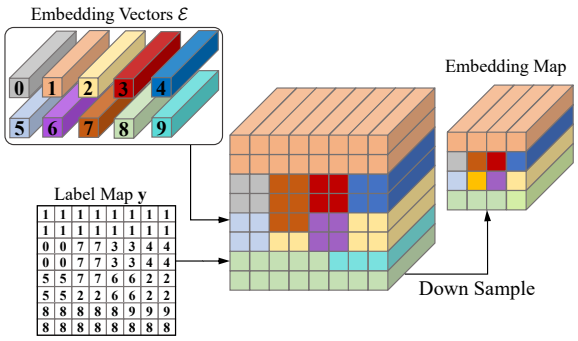

Figure 3. An illustration of constructing the domain-agnostic embedding map with prior vectors. Each number in the label map stands for a class ID. The total number of classes can be arbitrary.

note the supervisions adjusted to the same size of the backbone outputs, $\mathrm{proj}(\cdot)$ projects the domain-agnostic embedding vectors to the image plane based on the labels (to be detailed later), and $g_{\mathrm{vi}}(\cdot)$ and $g_{\mathrm{pr}}(\cdot)$ are each a learnable convolution layer that maps initial features from the vision and prior spaces to the common space.

We elaborate the construction of $\mathrm{proj}(\cdot)$ in Fig 3. The computation is simple: for each position on the image plane, we refer to the ground-truth label $\tilde{\mathbf{y}}_n^{\mathbb{S}}$ or partly-pseudo label $\tilde{\mathbf{y}}_{n,m}^{\mathbb{M}}$, and directly paste the class-wise embedding vectors to the corresponding positions. Since $\mathcal{E}$ is fixed, this procedure does not need any gradient back-propagation. We are aware that advanced versions of $\mathrm{proj}(\cdot)$ (*e.g.*, performing local smoothing) are available, and we leave these properties to be learned by $g_{\mathrm{vi}}(\cdot)$ and $g_{\mathrm{pr}}(\cdot)$.

Finally, the overall loss function is written as

$$\mathcal{L}_{\mathrm{overall}} = \mathcal{L}_{\mathrm{seg}} + \alpha \cdot \mathcal{L}_{\mathrm{DAP}}, \tag{5}$$

where α is the balancing coefficient which is by default set to be 1.0 – an analysis of the effect of α is provided in the experimental part (see Tab 6).

Back to Tab 1, one can observe how DAP reduces the IOU between $\mathcal{N}_{\mathrm{A}}^{\mathbb{T}}$ and $\mathcal{N}_{\mathrm{B}}^{\mathbb{T}}$, and thus alleviate the confusion between the semantically similar classes. Interestingly, the gains of two class pairs are consistent and significant, even though the training samples of *road* vs. *sidewalk* being more abundant. In experiments, we shall see how the ability of discriminating these class pairs is improved.

3.4. Discussions

To the best of our knowledge, this is the first work that integrates text embedding into UDA segmentation and producing considerable accuracy gain, which demonstrating the effectiveness of linguistic cues assisting visual recognition. However, it is yet a preliminary solution, and some possible directions can be discovered.

- **Enhancing text embedding.** The currently used word2vec features it does not consider different words that correspond to the same semantics (*e.g.*, *person* can be *pedestrian*). Interestingly, we tried to enhance the prior by searching for semantically similar words, but obtained little accuracy gain. This may call for a complicated mechanism of exploring the text world.

- **Constructing domain-agnostic yet vision-aware priors.** This is to answer the question: what kind of image data is considered to offer domain-free information? The answer may lie in generalized datasets like ImageNet [11] or Conceptual Captioning [43], or even the pre-trained image-text models such as CLIP [38] that absorbed 400 million image-text pairs. Note that it is a major challenge to disentangle domain-related information to avoid over-fitting, and we will continue exploring the possibility in the future.

4. Experiments

4.1. Datasets and Implementation Details

- **Datasets.** We evaluate our method on a popular scenario transferring the information from a synthesis domain to a real domain. We use GTAv [39] and SYNTHIA [40] as composite domain datasets and Cityscapes [10] as the real domain. GTAv [39] is a synthetic dataset extracted from the game of Grand Theft Auto V. There are 24,966 images with pixel-level semantic segmentation ground truth. The resolution of these images is 1914×1052 and we resize them into 1280×720 in our experiments. GTAv shares 19 common classes with Cityscapes. SYNTHIA [40] contains 9,400 virtual European-style urban images whose resolution is 1280×760 and we keep the original size in our experiments. We evaluate two settings (13 and 16 categories) in SYNTHIA. Cityscapes is a large-scale dataset with a resolution of 2048×1024. There are 2,975 and 500 images in the training and validation sets, respectively.

- **Implementation Details.** To be consistent with other methods, we use the Deeplabv2 [6] framework with a RseNet101 [18] backbone as our image encoder and an ASPP classifier as segmentation head. The output map is up-sampled and operated by a softmax layer to match the size of the inputs. The pre-trained model on ImageNet [11] and MSCOCO [30] is applied to initialize the backbone. The visual and prior feature projectors, $g_{\mathrm{vi}}(\cdot)$ and $g_{\mathrm{pr}}(\cdot)$, are both convolution layers with a 1×1 kernel to adjust the channel to 256. The batch size of one GPU is set to 2. We use SGD with Nesterov acceleration as the optimizer which is decreased based on a polynomial decay policy with exponent 0.9. The initial learning rate of backbone and feature projectors is 2.5×10^{-4}, that of segmentation head is $10\times$ larger. The momentum and weight decay of the optimizer are 0.9 and 5×10^{-4}. During the training process, we apply color jittering and Gaussian blurring on the mixed data and only resize operation on source and target data. Teacher model is updated with an EMA decay and λ equals 0.99.

GTAv→Cityscapes																				
Method	Road	Sidewalk	Building	Wall	Fence	Pole	Light	Sign	Veg	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motor	Bike	**mIOU**
AdaptSegNe [50]	86.5	36.0	79.9	23.4	23.3	23.9	35.2	14.8	83.4	33.3	75.6	58.5	27.6	73.7	32.5	35.4	3.9	30.1	28.1	42.4
PatchAligne [51]	92.3	51.9	82.1	29.2	25.1	24.5	33.8	33.0	82.4	32.8	82.2	58.6	27.2	84.3	33.4	46.3	2.2	29.5	32.3	46.5
LTIR [24]	92.9	55.0	85.3	34.2	31.1	34.9	40.7	34.0	85.2	40.1	87.1	61.0	31.1	82.5	32.3	42.9	0.3	36.4	46.1	50.2
PIT [33]	87.5	43.4	78.8	31.2	30.2	36.3	39.9	42.0	79.2	37.1	79.3	65.4	**37.5**	83.2	46.0	45.6	**25.7**	23.5	49.9	50.6
FDA [56]	92.5	53.3	82.4	26.5	27.6	36.4	40.6	38.9	82.3	39.8	78.0	62.6	34.4	84.9	34.1	**53.1**	16.9	27.7	46.4	50.5
MetaCorrect [17]	92.8	58.1	86.2	39.7	33.1	36.3	42.0	38.6	85.5	37.8	87.6	62.8	31.7	84.8	35.7	50.3	2.0	36.8	48.0	52.1
DACS [49]	89.9	39.7	**87.9**	39.7	**39.5**	38.5	46.4	**52.8**	**88.0**	44.0	**88.8**	67.2	35.8	84.5	45.7	50.2	0.0	27.3	34.0	52.1
IAST [34]	**94.1**	**58.8**	85.4	39.7	29.2	25.1	43.1	34.2	84.8	34.6	88.7	62.7	30.3	87.6	42.3	50.3	24.7	35.2	40.2	52.2
DPL [9]	92.8	54.4	86.2	**41.6**	32.7	36.4	**49.0**	34.0	85.8	41.3	86.0	63.2	34.2	87.2	39.3	44.5	18.7	**42.6**	43.1	53.3
Source only	75.6	17.1	69.8	10.7	16.1	21.1	27.0	10.6	77.3	15.1	71.1	53.4	20.5	73.9	28.6	31.1	1.62	32.4	21.5	35.5
DACS (rep)	93.1	48.1	87.3	36.7	35.1	38.7	42.5	49.3	87.5	41.9	87.9	64.8	30.7	88.3	40.2	51.0	0.0	25.1	42.6	52.1
DACS+DAP	93.5	53.9	87.5	30.0	36.4	**39.0**	43.9	49.5	87.5	**45.4**	**88.8**	**66.6**	36.8	**89.4**	**49.1**	51.4	0.0	42.2	**53.1**	**55.0**
ProDA [58]	87.8	56.0	79.7	46.3	44.8	45.6	53.5	53.5	88.6	45.2	82.1	70.7	39.2	88.8	45.5	50.4	1.0	48.9	56.4	57.5
Chao *et al.* [4]	94.4	60.9	88.1	39.5	41.8	43.2	49.1	56.0	88.0	45.8	87.8	67.6	38.1	90.1	57.6	51.9	0.0	46.6	55.3	58.0
DAP + ProDA	**94.5**	**63.1**	**89.1**	29.8	**47.5**	**50.4**	**56.7**	**58.7**	**89.5**	**50.2**	87.0	**73.6**	38.6	**91.3**	50.2	**52.9**	0.0	**50.2**	**63.5**	**59.8**

Table 2. Segmentation accuracy (IOU, %) of different UDA approaches from GTAv to Cityscapes. DACS (rep) indicates our re-implementation of DACS. For each class, we mark the highest number with **bold** and the second highest with underline. The top and bottom parts are achieved without and with multi-stage training or multi-model fusion.

SYNTHIA→Cityscapes																		
Method	Road	Sidewalk	Building	Wall*	Fence*	Pole*	Light	Sign	Veg	Sky	Person	Rider	Car	Bus	Motor	Bike	mIOU	mIOU*
PatchAlign [51]	82.4	38.0	78.6	8.7	0.6	26.0	3.9	11.1	75.5	84.6	53.5	21.6	71.4	32.6	19.3	31.7	40.0	46.5
AdaptSegNe [50]	84.3	42.7	77.5	–	–	–	4.7	7.0	77.9	82.5	54.3	21.0	72.3	32.2	18.9	32.3	–	46.7
FDA [56]	79.3	35.0	73.2	–	–	–	19.9	24.0	61.7	82.6	61.4	31.1	83.9	40.8	**38.4**	51.1	–	52.5
LTIR [24]	**92.6**	**53.2**	79.2	–	–	–	1.6	7.5	78.6	84.4	52.6	20.0	82.1	34.8	14.6	39.4	–	49.3
PIT [33]	83.1	27.6	81.5	8.9	0.3	21.8	26.4	**33.8**	76.4	78.8	64.2	27.6	79.6	31.2	31.0	31.3	44.0	51.8
MetaCorrect [17]	92.6	52.7	81.3	8.9	2.4	28.1	13.0	7.3	83.5	85.0	60.1	19.7	84.8	37.2	21.5	43.9	45.1	52.5
DPL [9]	87.5	45.7	82.8	13.3	0.6	33.2	22.0	20.1	83.1	86.0	56.6	21.9	83.1	40.3	29.8	45.7	47.0	54.2
DACS [49]	80.6	25.1	81.9	21.5	2.9	**37.2**	22.7	24.0	83.7	**90.8**	**67.6**	**38.3**	82.9	38.9	28.5	47.6	48.3	54.8
IAST [34]	81.9	41.5	**83.3**	17.7	**4.6**	32.3	30.9	28.8	83.4	85.0	65.6	30.8	**86.5**	38.2	33.1	**52.7**	49.8	57.0
Source only	24.7	12.1	75.4	11.3	0.1	22.4	7.5	16.6	71.7	78.3	52.9	10.1	56.6	23.3	4.0	13.0	30.0	34.3
DACS (rep)	82.1	31.0	82.4	22.1	1.2	33.1	**32.7**	25.1	**84.4**	88.2	65.2	34.3	83.4	42.9	24.1	50.8	48.9	55.9
DACS+DAP	83.9	33.3	80.2	**24.1**	1.2	33.4	30.8	**33.8**	84.3	88.5	65.7	36.2	84.3	**43.3**	33.3	46.3	**50.2**	**57.2**
Chao *et al.* [4]	88.7	46.7	83.8	22.7	4.1	35.0	35.9	36.1	82.8	81.4	61.6	32.1	87.9	52.8	32.0	57.7	52.6	60.0
ProDA [58]	87.8	45.7	84.6	37.1	0.6	44.0	54.6	37.0	88.1	84.4	74.2	24.3	88.2	51.1	40.5	45.6	55.5	62.0
DAP + ProDA	84.2	**46.5**	82.5	35.1	0.2	**46.7**	53.6	**45.7**	**89.3**	**87.5**	**75.7**	**34.6**	**91.7**	**73.5**	**49.4**	**60.5**	**59.8**	**64.3**

Table 3. Segmentation accuracy (IOU, %) of different UDA approaches from SYNTHIA to Cityscapes. DACS (rep) indicates our re-implementation of DACS. For each class, we mark the highest number with **bold** and the second highest with underline. The top and bottom parts are achieved without and with multi-stage training or multi-model fusion. mIOU* denotes the average of 13 classes, with the classes marked with * not computed.

We train the model for 250K iterations on a single NVIDIA Tesla-V100 GPU and adopt an early stop setting. The background and invalid categories are ignored during training. The weight of $\mathcal{L}_{\text{DAP}}$, *i.e.* α, is set to be 1.0. We adapt bilinear interpolation to down-sample the embedding map from the input size to that of the visual feature map – this is an important step for DAP, which is ablated in Section 4.3.

4.2. Quantitative Results and Visualization

We first evaluate DAP on the transfer segmentation task from GTAv to Cityscapes. The comparison against recent approaches are shown in Tab 2. To ensure reliability, we run DACS (the baseline) and DAP three times and report the average accuracy. DAP achieves a mean IOU of 55.0% over 19 classes, which claims a 2.9% gain beyond the baseline and also outperforms all other competitors except for Chao *et al.* [4] and ProDA [58]. Specifically, Chao *et al.* [4] used ensemble learning to integrate the prediction from four complementarily-trained models, including DACS, but DAP used a single model; ProDA [58] improved the segmentation accuracy significantly via multi-stage training, yet its first stage reported a 53.7% mIOU.What is more, the results on transferring SYNTHIA to Cityscapes, as shown in Tab 3, demonstrate the similar trend – DAP outperforms

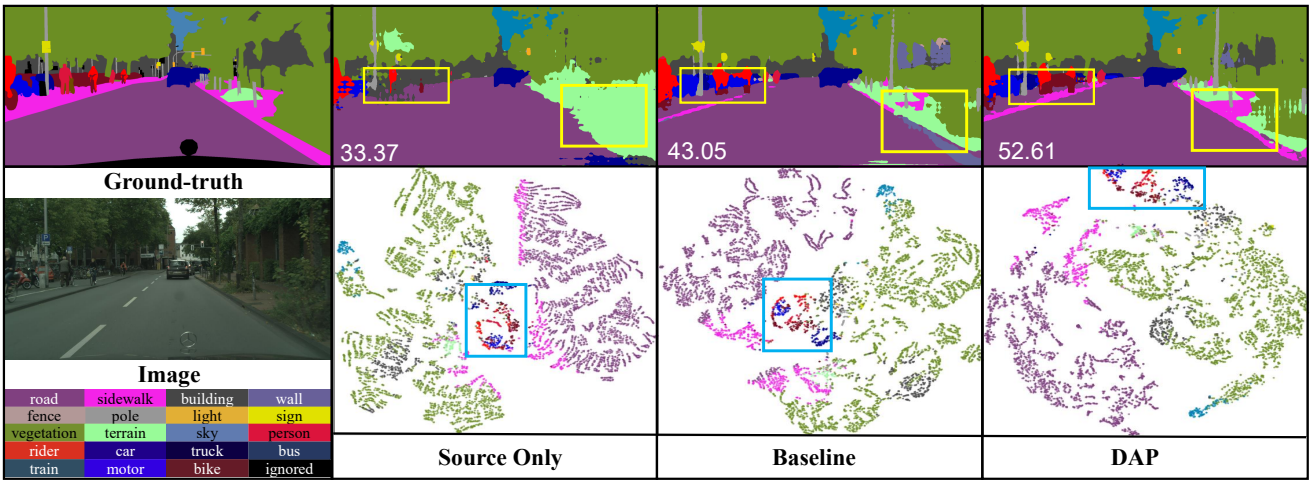

Figure 4. An example of transfer segmentation from GTAv to Cityscapes. The top row shows the ground-truth and three segmentation results, while the input and legend is in the bottom row. The yellow boxes indicate the regions that the segmentation quality is largely improved, and the number corresponds to single-image mIOU (over the existing classes). The bottom row also shows the t-SNE of visual features colored by the predicted class. The blue boxes locate the features of *bike* and *motorbike*, in which DAP achieves a favorable ability to distinguish them. *This figure is best viewed in color and we suggest the reader to zoom in for details.*

all the competitors, except for ProDA and RED, in terms of either 13-class or 16-class mIOU. To show that DAP offers complementary benefits, we feed the output of DAP as the pseudo labels to the 1st stage of ProDA, and the 2nd and 3rd stages remain unchanged. As shown in Tables 2 and 3, the segmentation mIOUs of ProDA in GTAv→Cityscapes and SYNTHIA→Cityscapes are improved by 2.3% and 4.3%, respectively, setting new records in these two scenarios.

Next, we investigate the ability of DAP in distinguishing semantically similar classes. From GTAv to Cityscapes, the segmentation mIOUs of *bike* and *motorbike* are improved from 42.6% and 25.1% to 53.1% and 42.2%, with absolute gains of 10.5% and 17.1%, respectively. From SYNTHIA to Cityscapes, the mIOU of *bike* drops by 4.5% and that of *motorbike* increases by 9.2%, achieving an average improvement of 2.4%. We visualize an example of segmentation in Fig 4. Besides a qualitative observation on the improvement of distinguishing *bike* from *motorbike* and *road* from *sidewalk*, we also notice the reason behind the improvement being a scattered feature distribution of these similar classes. This aligns with the statistics shown in Tab 1, indicating that DAP reduces the IOU between the estimated distributions of *bike* and *motorbike* as well as that between *road* from *sidewalk*.

From another perspective, we study how the language-based prior assists visual recognition. In Fig 5, we show the relationship matrix of the features learned from the source (GTAv) and target (Cityscapes) domains as well as the word2vec features. There is an interesting example that *person* and *rider* are semantically similar in both GTAv and Cityscapes but not so correlated according to word2vec.

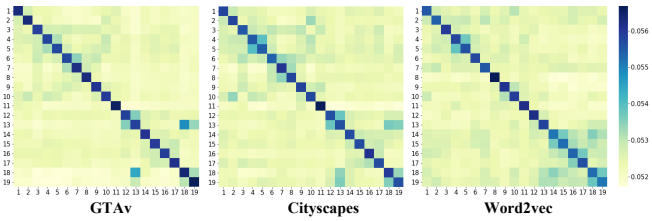

Figure 5. The relationship matrix of all 19 classes, where the left and middle are generated by the class-averaged feature vectors extracted from the models directly trained on GTAv and Cityscapes, and the right uses the word2vec features. Each cell is the inner product of two normalized features. The order of class is identical to that in Tab 2 – for easier reference: 1 for *road*, 2 for *sidewalk*, 12 for *person*, 13 for *rider*, *18* for *motorbike*, 19 for *bike*.

Source Dataset	GTAv			SYNTHIA		
	mIOU (%)	gain	std(%)	mIOU (%)	gain	std(%)
Baseline	52.1	–	1.6	48.9	–	0.3
w/ random vectors	52.3	0.2↑	2.3	48.0	0.9↓	0.8
w/ one-hot vectors	53.0	0.9↑	0.8	49.8	0.9↑	0.4
w/ CLIP [38]	54.6	2.5↑	0.6	**50.6**	1.7↑	0.5
w/ word2vec [35]	**55.0**	2.9↑	0.5	50.2	1.3↑	0.4

Table 4. The results of different vectors choices include the adaptation from GTAv and SYNTHIA

This may cause a strong yet harmful correlation of these two feature sets, leading to confusion in segmentation. The relatively weaker correlation of word2vec features alleviates the bias and improves the IOU of both classes (refer to Tab 2). Similar phenomena are also observed for the class pairs of *road* vs. *sidewalk*, and *bike* vs. *motorbike*.

4.3. Diagnostic Studies

• **The choice of domain-agnostic prior.** We investigate three other options except for word2vec embedding [35], namely, (1) that generating a 300-dimensional, normalized **random vector** for each class, (2) directly creating a **one-hot vector** for each class, where the dimensionality equals to the number of classes, (3) classes embeddings generated by the language branch of **CLIP** [38]. Similar to the main experiments, we run each option three times and report the averaged accuracy. Results are summarized in Tab 4.

One can see that random vectors achieve a slight 0.2% accuracy gain on GTAv→Cityscapes, but incurs a 0.9% accuracy drop on SYNTHIA→Cityscapes, implying the instability. Throughout the three individual runs on GTAv→Cityscapes, the best run achieves a 54.4% mIOU, just 0.6% lower than using word2vec, but the worst one reports 49.2% which is even significantly lower than the baseline. Regarding the one-hot vectors, the results over three runs are less diversified, and the average improvement is consistent, (*i.e.*, 0.9% on both GTAv→Cityscapes and SYNTHIA→Cityscapes), though smaller than that brought by word2vec embedding. The prior from CLIP [38] reports 54.6% and 50.6% mIOUs on the GTAv and SYNTHIA experiments, respectively. Despite the fact that CLIP is stronger than word2vec, the mIOUs are just comparable to that using word2vec (55.0% and 50.2%). From these results, we learn the lesson that (1) even a naive prior alleviates the inter-class confusion caused by domain shift, however, (2) it would be better if the inter-class relationship is better captured so that the model is aware of semantically similar classes – text embedding offers a safe and effective option. (3) The limited number and diversity of target categories may have diminished the advantages of a stronger language model (*e.g.*, CLIP), and we still pursue for a vision-aware yet domain-agnostic embedding method.

• **Different Backbones.** To verify the effectiveness of DAP on different network structures, we replace the convolution backbone (ResNet101) with a transformer network (ViT-Base [12]). The numbers of the block layers, token size, and heads are $12, 768$ and 12 respectively in the transformer encoder. The input size of the training data is set 768×768. And we initiate the ViT encoder with a model pre-trained on ImageNet-21k then fine-tune the segmentation network with a base learning rate of 0.01 adopted with the 'poly' learning rate decay and use SGD as the optimizer. The g_{pr} is one layer transformer structure and weight of $\mathcal{L}_{\text{DAP}}$ is 0.25. On GTAv, source-only, DACS, DAP report 49.4%, 58.4%, 61.1% mIOUs. As for the transferring from SYNTHIA, the numbers of the three settings are 42.7%, 53.2%, 59.1% when evaluating on 16 classes and 48.1%, 60.9%, 66.1% on 13 classes. We can see that DAP still obtains consistent accuracy gain. To the best of our knowledge, the DAP numbers are **SOTA**.

Setting	source only	DACS	DAP w/o interp	DAP w/ interp
GTAv	35.5	52.1	53.8	**55.0**
SYNTHIA	30.0	48.9	49.6	**50.2**

Table 5. Ablation on the contribution of each module in both GTAv→Cityscapes and SYNTHIA→Cityscapes experiments.

α	0.50	0.75	1.00	1.25	1.50
GTAv	54.5	54.1	**55.0**	54.3	54.7
SYNTHIA	50.0	49.4	**50.2**	49.6	49.1

Table 6. Impact of tuning the balancing coefficient, α. All numbers are segmentation mIOU (%).

• **The importance of feature interpolation.** From the model that only uses the source domain, we gradually add mixed data (by DACS [49]), introduce DAP (using the word2vec embedding), and perform feature interpolation to down-sample the embedding map. As shown in Tab 5, for the both two transferring scenarios, feature interpolation contributes nearly half accuracy gain of DAP over DACS. Intuitively, feature interpolation enables the features on small-area classes to be preserved, yet nearest-neighbor down-sampling can cause these features to be ignored.

• **Parameter Analysis.** Lastly, we study the impact of the coefficient α that balances between $\mathcal{L}_{\text{seg}}$ and $\mathcal{L}_{\text{DAP}}$. The results shown in Tab 6 suggest that $\alpha = 1.0$ is the best option. In addition, increasing α causes a larger accuracy drop compared to decreasing it, which may indicate that $\mathcal{L}_{\text{seg}}$ is the essential goal and $\mathcal{L}_{\text{DAP}}$ serves as an auxiliary term.

5. Conclusions

In this paper, we investigate the UDA segmentation problem and observe that semantically similar classes are easily confused during the transfer procedure. We formulate this problem using the Bayesian theory and owe such confusion to the weakness of likelihood, *e.g.*, insufficient training data. To alleviate the issue, we introduce domain-agnostic priors to compensate the likelihood. Experiments on two standard benchmarks for UDA segmentation in urban scenes verify its effectiveness both quantitatively and qualitatively

Limitations of this work. Currently, the best option of DAP is to leverage the text embedding vectors. We are looking forward to more powerful priors, *e.g.*, from the cross-modal pre-trained models [38, 22]. This may call for more sophisticated designs of prior embedding, projection, alignment, *etc.*, which we leave for future work.

Acknowledge This work was supported in part by the National Natural Science Foundation of China under Contract U20A20183 and 62021001, and in part by the Youth Innovation Promotion Association CAS under Grant 2018497. It was also supported by the GPU cluster built by MCC Lab of Information Science and Technology Institution, USTC.

References

[1] Donghyeon Baek, Youngmin Oh, and Bumsub Ham. Exploiting a joint embedding space for generalized zero-shot semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9536–9545, 2021. 3

[2] Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, and Dilip Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3722–3731, 2017. 2

[3] Maxime Bucher, Tuan-Hung Vu, Matthieu Cord, and Patrick Pérez. Zero-shot semantic segmentation. *Advances in Neural Information Processing Systems*, 32:468–479, 2019. 3

[4] Chen-Hao Chao, Bo-Wun Cheng, and Chun-Yi Lee. Rethinking ensemble-distillation for semantic segmentation based unsupervised domain adaption. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2610–2620, 2021. 2, 6

[5] Chaoqi Chen, Weiping Xie, Wenbing Huang, Yu Rong, Xinghao Ding, Yue Huang, Tingyang Xu, and Junzhou Huang. Progressive feature alignment for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 627–636, 2019. 2

[6] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 5

[7] Minghao Chen, Hongyang Xue, and Deng Cai. Domain adaptation for semantic segmentation with maximum squares loss. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2090–2099, 2019. 2

[8] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3339–3348, 2018. 2

[9] Yiting Cheng, Fangyun Wei, Jianmin Bao, Dong Chen, Fang Wen, and Wenqiang Zhang. Dual path learning for domain adaptation of semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9082–9091, 2021. 6

[10] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 2, 3, 5

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3, 8

[13] Fangxiang Feng, Xiaojie Wang, and Ruifan Li. Cross-modal retrieval with correspondence autoencoder. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 7–16, 2014. 3

[14] Yang Fu, Yunchao Wei, Guanshuo Wang, Yuqian Zhou, Honghui Shi, and Thomas S Huang. Self-similarity grouping: A simple unsupervised cross domain adaptation approach for person re-identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6112–6121, 2019. 2

[15] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015. 2

[16] Li Gao, Jing Zhang, Lefei Zhang, and Dacheng Tao. Dsp: Dual soft-paste for unsupervised domain adaptive semantic segmentation. *arXiv preprint arXiv:2107.09600*, 2021. 2

[17] Xiaoqing Guo, Chen Yang, Baopu Li, and Yixuan Yuan. Metacorrection: Domain-aware meta loss correction for unsupervised domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3927–3936, 2021. 2, 6

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[19] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pages 1989–1998. PMLR, 2018. 2, 4

[20] Hengtong Hu, Lingxi Xie, Richang Hong, and Qi Tian. Creating something from nothing: Unsupervised knowledge distillation for cross-modal hashing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3123–3132, 2020. 3

[21] Xinyue Huo, Lingxi Xie, Jianzhong He, Zijie Yang, Wengang Zhou, Houqiang Li, and Qi Tian. Atso: Asynchronous teacher-student optimization for semi-supervised image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1235–1244, 2021. 2

[22] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. *arXiv preprint arXiv:2102.05918*, 2021. 2, 8

[23] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1780–1790, 2021. 2

[24] Myeongjin Kim and Hyeran Byun. Learning texture invariant representation for domain adaptation of semantic seg-

mentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12975–12984, 2020. 2, 6

[25] Seunghyeon Kim, Jaehoon Choi, Taekyung Kim, and Changick Kim. Self-training and adversarial background regularization for unsupervised domain adaptive one-stage object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6092–6101, 2019. 2

[26] Hugo Larochelle, Dumitru Erhan, and Yoshua Bengio. Zero-data learning of new tasks. In *AAAI*, volume 1, page 3, 2008. 2

[27] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896, 2013. 2

[28] Xinzhe Li, Qianru Sun, Yaoyao Liu, Qin Zhou, Shibao Zheng, Tat-Seng Chua, and Bernt Schiele. Learning to self-train for semi-supervised few-shot classification. *Advances in Neural Information Processing Systems*, 32:10276–10286, 2019. 2

[29] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer, 2020. 3

[30] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 5

[31] Jinlu Liu, Liang Song, and Yongqiang Qin. Prototype rectification for few-shot learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 741–756. Springer, 2020. 2

[32] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *International conference on machine learning*, pages 97–105. PMLR, 2015. 2

[33] Fengmao Lv, Tao Liang, Xiang Chen, and Guosheng Lin. Cross-domain semantic segmentation via domain-invariant interactive relation transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4334–4343, 2020. 6

[34] Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. Instance adaptive self-training for unsupervised domain adaptation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*, pages 415–430. Springer, 2020. 6

[35] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013. 2, 4, 7, 8

[36] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 891–898, 2014. 2

[37] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. Classmix: Segmentation-based data augmentation for semi-supervised learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1369–1378, 2021. 2, 3

[38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021. 2, 5, 7, 8

[39] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *European conference on computer vision*, pages 102–118. Springer, 2016. 2, 3, 5

[40] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio 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*, pages 3234–3243, 2016. 2, 3, 5

[41] Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. Semi-supervised self-training of object detection models. 2005. 2

[42] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3723–3732, 2018. 2

[43] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. 5

[44] Inkyu Shin, Sanghyun Woo, Fei Pan, and In So Kweon. Two-phase pseudo label densification for self-training based domain adaptation. In *European conference on computer vision*, pages 532–548. Springer, 2020. 2

[45] Liangchen Song, Cheng Wang, Lefei Zhang, Bo Du, Qian Zhang, Chang Huang, and Xinggang Wang. Unsupervised domain adaptive re-identification: Theory and practice. *Pattern Recognition*, 102:107173, 2020. 2

[46] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. 3

[47] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *arXiv preprint arXiv:1703.01780*, 2017. 2, 3

[48] Marco Toldo, Umberto Michieli, Gianluca Agresti, and Pietro Zanuttigh. Unsupervised domain adaptation for mobile semantic segmentation based on cycle consistency and feature alignment. *Image and Vision Computing*, 95:103889, 2020. 2

[49] Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. Dacs: Domain adaptation via cross-domain mixed sampling. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1379–1389, 2021. 1, 2, 3, 6, 8

[50] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7472–7481, 2018. 2, 6

[51] Yi-Hsuan Tsai, Kihyuk Sohn, Samuel Schulter, and Manmohan Chandraker. Domain adaptation for structured output via discriminative patch representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1456–1465, 2019. 6

[52] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017. 2

[53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 3

[54] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2517–2526, 2019. 2

[55] Bokun Wang, Yang Yang, Xing Xu, Alan Hanjalic, and Heng Tao Shen. Adversarial cross-modal retrieval. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 154–162, 2017. 3

[56] Yanchao Yang and Stefano Soatto. Fda: Fourier domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4085–4095, 2020. 2, 6

[57] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 2

[58] Pan Zhang, Bo Zhang, Ting Zhang, Dong Chen, Yong Wang, and Fang Wen. Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12414–12424, 2021. 6

[59] Hang Zhao, Xavier Puig, Bolei Zhou, Sanja Fidler, and Antonio Torralba. Open vocabulary scene parsing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2002–2010, 2017. 3

[60] Qianyu Zhou, Zhengyang Feng, Qiqi Gu, Guangliang Cheng, Xuequan Lu, Jianping Shi, and Lizhuang Ma. Uncertainty-aware consistency regularization for cross-domain semantic segmentation. *arXiv preprint arXiv:2004.08878*, 2020. 2

[61] Qianyu Zhou, Zhengyang Feng, Qiqi Gu, Jiangmiao Pang, Guangliang Cheng, Xuequan Lu, Jianping Shi, and Lizhuang Ma. Context-aware mixup for domain adaptive semantic segmentation. *arXiv preprint arXiv:2108.03557*, 2021. 2

[62] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 2, 4

[63] Xiaojin Jerry Zhu. Semi-supervised learning literature survey. 2005. 2

[64] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European conference on computer vision (ECCV)*, pages 289–305, 2018. 1, 2

[65] Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. Confidence regularized self-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5982–5991, 2019. 2