Domain Adaptation for Semantic Segmentation via Patch-Wise Contrastive Learning

Weizhe Liu\textsuperscript{1} David Ferstl\textsuperscript{2} Samuel Schulter\textsuperscript{3} Lukas Zebedin\textsuperscript{3} Pascal Fua\textsuperscript{1} Christian Leistner\textsuperscript{3}
\textsuperscript{1} CVLab, EPFL \textsuperscript{2} Magic Leap \textsuperscript{3} Amazon

Abstract

We introduce a novel approach to unsupervised and semi-supervised domain adaptation for semantic segmentation. Unlike many earlier methods that rely on adversarial learning for feature alignment, we leverage contrastive learning to bridge the domain gap by aligning the features of structurally similar label patches across domains. As a result, the networks are easier to train and deliver better performance. Our approach consistently outperforms state-of-the-art unsupervised and semi-supervised methods on two challenging domain adaptive segmentation tasks, particularly with a small number of target domain annotations. It can also be naturally extended to weakly-supervised domain adaptation, where only a minor drop in accuracy can save up to 75\% of annotation cost.

1. Introduction

Given large amounts of annotated training data, current fully-supervised semantic segmentation algorithms deliver outstanding results. Because annotating many images at the pixel-level is expensive, a common practice is to generate synthetic data and to rely on unsupervised domain adaptation to bridge the gap from the synthetic source to the real-world target domain.

In this paper, we introduce a novel approach to aligning cross-domain features for both unsupervised and semi-supervised domain adaptation. In the first case, no target domain labels are given, whereas in the second one only a small amount of annotated data is available in the target domain. In practice, this second scenario is important because even a handful of target domain labels can boost the performance significantly.

Our key insight is that patches from both domains that are structurally similar in label space should also have similar distributions in feature space. To enforce this, which goes beyond what can be done using ordinary pixel-wise similarity, we introduce a patch-wise metric. It measures label disparity at several levels of resolution along with a contrastive loss [5] that, when minimized, aligns the feature distribution closer for patches with similar structures in label space and pushes them apart otherwise, see Fig. 1.

To perform unsupervised or semi-supervised training, we incorporate unlabeled pixels into training by using pseudo labels that we iteratively generate using the output of the partially trained network. At each training iteration, we use our patch-wise metric to decrease the feature space disparity of patches that are structurally similar and to increase it for those that are not. This renders a more straightforward approach than adversarial learning, previously often used for cross-domain feature alignment. We do not require an extra discriminator network and therefore eliminate the sometimes substantial difficulty of having to train it.

Our experiments show that our approach delivers a performance increase over state-of-the-art methods in the unsupervised regime [43, 49] and an even larger boost in the semi-supervised one [45, 20]. In a practical setting, we believe the latter to be particularly significant because, while it is very tedious to get huge amounts of annotated frames, it is almost always possible to supply a few annotated frames.
Our contribution is therefore a new contrastive learning approach to aligning features across different domains for semantic segmentation. It relies on structural label disparity instead of adversarial learning and outperforms state-of-the-art methods for unsupervised domain adaptation and even more for semi-supervised domain adaptation. We also show how our approach can be extended to weakly-supervised domain adaptation, where only a minor drop in accuracy can save up to 75% of annotation cost.

2. Related Work

Most recent works on domain adaptation focus on unsupervised methods, with only very few incorporating the limited amount of supervision we advocate. We briefly review these two classes of approaches and then discuss contrastive learning, which is a central component of our approach.

Unsupervised Domain Adaptation (UDA) for Semantic Segmentation aims to align the source and target domain feature distributions given annotated data only in the source domain. A popular approach is to leverage adversarial learning to generate domain-invariant features. This trend started with [15] and was extended to different levels of representation, including feature space [6, 16, 32, 38] and label space [7, 17, 30, 42, 43, 44]. Notable extensions are [4, 7, 32] which enforce class-wise alignment to narrow the distributions to be matched. However, all these adversarial learning approaches rely on extra discriminator networks which are complicated and hard to train jointly with the generator network.

Other widely used UDA approaches to semantic segmentation include generating realistic-looking synthetic images [18, 39, 46, 48, 49, 51], using pseudo labels for self-training [24, 26, 40, 49, 53], and leveraging weak labels [28, 32]. Among these methods, FDA [49] is a simple approach which achieves state-of-the-art performance by generating realistic-looking synthetic images directly in Fourier space and self-training.

Our approach builds on FDA and incorporates a novel domain-wise contrastive loss, without the reliance on complex adversarial learning.

Semi-Supervised Domain Adaptation for Semantic Segmentation (SSDA) assumes that a handful of target domain labels are available. In the context of semantic segmentation it has not received as much attention as UDA. For example, [37] achieves adaptation by alternately maximizing the conditional entropy of unlabeled target data with respect to the classifier and minimizing it with respect to the feature encoder. In [36], a two-stream architecture is proposed, where one stream operates in the source and the other in the target domain. In contrast to others, the weights in corresponding layers are related but not shared and optimized to deliver good performance in both domains. In [22], target domain samples are perturbed to reduce intra-domain discrepancy using adversarial learning.

However, none of these works have been demonstrated for semantic segmentation, nor do they leverage contrastive learning for SSDA. The only ones that do are [48] and [20]. In [45], class-wise adversarial learning is used to promote the similarity of pixel-level feature representations for the same classes. In [20], a pixel-level entropy regularization scheme is introduced to favor feature alignment among multiple domains. Therefore, domain alignment is only enforced on the pixel-level, whereas ours is done at a more semantic level.

Contrastive Learning (CL) aims to learn visual representations by leveraging both similar and dissimilar samples. Early work [12] showcased improved visual representation by contrasting positive pairs against negative ones. Deep CL has been used extensively [10, 47, 52, 41, 13, 29, 5, 25, 9, 50, 19, 21] for applications such as image classification, image-to-image translation [31] or phrase grounding [11].

In the context of semantic segmentation, CL has also been used for intra-domain model pre-training [2]. The approach is specifically designed for use in conjunction with a U-Net [34] backbone and does not generalize well to other architectures. In our experiments, we show that using CL directly as loss term achieves superior results compared to CL only used for model pre-training.

3. Approach

We first formalize the problem of domain adaptation in Sec. 3.1. We then give an overview of our proposed model in Sec. 3.2 and explain how contrastive learning is used for domain adaptation in Sec. 3.3 and 3.4.

3.1. Problem Formulation

Let $D^s = \{ (x_i^s, y_i^s) \}_{i=1}^{N_s}$ be a source-domain dataset, where $x^s$ denotes a color image and $y^s$ the corresponding semantic map. The target-domain dataset is split into two sets. The first is a labeled set $D^l = \{ (x_i^l, y_i^l) \}_{i=1}^{N_l}$ with ground truth semantic maps. The second is the unlabeled set $D^u = \{ x_i^u \}_{i=1}^{N_u}$ without ground truth semantic labels. We have $N_l = 0$ in the Unsupervised Domain Adaptation (UDA) setting and $N_l > 0$ for Semi-Supervised Domain Adaptation (SSDA). In most real-world scenarios, we have $N_u \gg N_l$ and $N_u \gg N_l$. Given the three data sets $D^s$, $D^l$ and $D^u$, the task is to learn a single model that performs well on previously unseen target domain data.
3.2. Overview of Our Approach

Our model consists of an encoder $E$ that maps an input image $x$ to a list of feature vectors $f = E(x)$ and two decoders $D_{la}$ and $D_{se}$ that map these intermediate features to a list of latent vectors $z = D_{la}(f)$ and a semantic map $y = D_{se}(f)$. Let $F_{se} = D_{se} \circ E$ and $F_{la} = D_{la} \circ E$ be the networks that take as input an image and return a semantic map and a latent vector respectively. Fig. 2 gives an overview of our network architecture and the loss functions, which we describe below.

Patch-wise representation: In practice, the functions $E$ and $D_{se}$ are convolutional neural networks, but for the sake of describing our contrastive loss in Sec. 3.3, a patch-wise representation is useful. Therefore, each image $x$ is decoded into $N_p$ latent vectors, corresponding to $N_p$ rectangular patches $x_i$ for $1 \leq i \leq N_p$ and the encoder preserves the association between features and patches so that we can write

$$f = E(x) = E([\hat{x}_1, \ldots, \hat{x}_{N_p}]) = [\hat{f}_1, \ldots, \hat{f}_{N_p}]$$

and

$$z = F_{la}(x) = D_{la}(f) = D_{la}( [\hat{f}_1, \ldots, \hat{f}_{N_p}] )
= [\hat{z}_1, \ldots, \hat{z}_{N_p}]$$

where $\hat{f}_i$ and $\hat{z}_i$ are local features and latent vectors associated to patch $\hat{x}_i$. In the remainder of the paper, we will denote by $\hat{x}^s$, $\hat{x}^l$ and $\hat{x}^u$ patches of the source, labeled and unlabeled target images.

Loss functions for classification: Training all parameters of our model involves minimizing several loss functions, including the novel contrastive loss that we describe in Sec. 3.3. For the per-pixel classification task, we also use supervised cross-entropy losses for the source domain images and, if available, labeled target domain images, respectively, which are defined as

$$L_{sup}^s = -\sum_i \langle y^s_i, log(F_{se}(x^s_i)) \rangle,$$

$$L_{sup}^l = -\sum_i \langle y^l_i, log(F_{se}(x^l_i)) \rangle,$$

Additionally, we employ a regularization loss

$$L_{ent} = \sum_i \rho(-\langle F_{se}(x^u_i), log(F_{se}(x^u_i)) \rangle),$$

where $\rho(x) = (x^2 + 0.001^2)^{\eta}$ is the Charbonnier penalty function [1]. As in [49], $L_{ent}$ penalizes uncertainty in the predictions for the unlabeled samples and encourages one label to dominate over the others.

We summarize these three losses into our base loss

$$L_{base} = L_{sup}^s + L_{sup}^l + \lambda_{ent} L_{ent},$$

with $\lambda_{ent}$ being a small scaling factor, usually $< 1$.

Pseudo Labels: After some initial training steps, we can use $F_{se}$ to assign pseudo labels to unlabeled images and to compute an additional cross-entropy loss term

$$L_{self} = -\sum_i \langle y^u_i, log(F_{se}(x^u_i)) \rangle,$$

where $y^u_i = \arg\max\{F_{se}(x^u_i)\}$ are the pseudo labels. In Sec. 4.5, we demonstrate that pseudo labeling becomes more effective in the SSDA setting, as compared to an unsupervised setting, because pseudo labels are more reliable.
3.3. Contrastive Learning for Domain Adaptation

The main contribution of this paper is to leverage contrastive learning for domain adaptation. Instead of relying on an adversarial training scheme, as most prior works do [45, 43], we use a contrastive loss on pairs of patches from different domains. The goal is to bring the representation of positive pairs closer together, while pushing negative pairs apart. A benefit of our approach is that optimizing this loss is relatively simpler, compared to adversarial learning, which involves a min-max optimization scheme. However, the main challenge is to define positive and negative pairs of patches across domains for the contrastive loss.

3.3.1 Matching of Patches for Contrastive Learning

The key idea of our approach is that if two patches (one from the source domain and the other from the target domain) are semantically similar, then their embedding in latent space \( \hat{z}_i = F_{la}(\hat{x}_i) \) should also be similar. Conversely, if two patches are semantically dissimilar, their embeddings should also have a large distance.

To find such pairs, let us assume a semantic disparity function \( D \), which we define formally in the following section. Patches with high semantic similarity have low disparity values \( D \), and vice-versa. We sample a patch in an image of one domain and compute the disparity to all patches in an image from the other domain. Pairs of patches with low disparity score are considered positive (\( D < \alpha \)) and pairs with high disparity negative (\( D > \beta \)). Pairs with disparity values in-between \( \alpha \) and \( \beta \) are simply ignored. Fig. 3 gives an example of positive and negative pairs.

To define a contrastive loss with the discovered pairs of patches, let us define the query patch \( \hat{x}^\dagger \) sampled from one domain, the positive patch \( \hat{x}^\star \) and \( k \) negative patches \( \{\hat{x}_i^-\}_{i=1}^k \) sampled from the other domain. Let \( \hat{z}^\dagger, \hat{z}^\star \) and \( \hat{z}_i^- \) be the corresponding latent vectors. We can then define the contrastive loss as

\[
L_{cont} = - \log \left[ \frac{\text{sim}(\hat{z}^\dagger, \hat{z}^\star)}{\text{sim}(\hat{z}^\dagger, \hat{z}^\star) + \sum_{i=1}^k \text{sim}(\hat{z}^\dagger, \hat{z}_i^-)} \right],
\]

with \( \text{sim}(u, v) = \exp \left( \frac{u^Tv}{\|u\|\|v\|\tau} \right) \) being the similarity between any two vectors, defined as the exponential of the cosine similarity normalized by temperature parameter \( \tau \).

3.3.2 Patch-Wise Semantic Disparity

We need a measure of patch-wise semantic disparity in label space to define positive and negative pairs as described in Sec. 3.3.1. Given a pair of patches, one from source and the other from the target domain, we define a metric accounting for both semantic and structural disparity in label space.

Let us consider patch \( \hat{x} \) and semantic map \( \hat{y} \) from either domain. Let \( f_k^{xc} \) be the proportion of pixels that have label \( c \) for \( 1 \leq c \leq N_c \). We take

\[
V_{\hat{x}} = [f_k^{xc}, f_k^{x2}, ..., f_k^{xN_c}] \in \mathbb{R}^{N_c}
\]

to be the semantic vector of \( \hat{x} \), containing the overall semantic information without any spatial layout information.

To also encode rough spatial information and allow for robust matching, we adopt a simplified version of spatial pyramid matching [23] in label space. We compute the semantic vector via Eq. (8) on three spatial levels, as shown in Fig. 4. We define the patch-wise semantic disparity between \( \hat{x}_s \) and \( \hat{x}_t \), from source and target domains, respectively, as

\[
D(\hat{x}_s, \hat{x}_t) = \sum_{m=0}^{2} N_2-m D_m (\hat{x}_s, \hat{x}_t),
\]

with \( D_m(\hat{x}_s, \hat{x}_t) = \sum_{i=1}^{N_m} \|V_{\hat{x}_i}^s - V_{\hat{x}_i}^t\|^2 \). That is, we measure \( \ell_2 \) distance between semantic vectors at 3 pyramid levels, each with \( N_m \) patches. \( \hat{x}_i^s, \hat{x}_i^t \) denote the \( i \)-th sub-patch of \( \hat{x}^s \) and \( \hat{x}^t \) at level \( m \). The first spatial level covers the whole patch, hence \( N_0=1 \).

At the second and third levels, we split the patch into 4 and 16 sub-patches, respectively, and set \( N_1 = 4 \) and \( N_2 = 16 \). The coefficient \( N_2-m \) in Eq. (9) ensures equal contribution from all levels.

3.4. Training Strategy

We can now put the individual losses Eq. (5), (6) and (7) together to define our overall training objective

\[
L_{all} = L_{base} + \lambda_{self} L_{self} + \lambda_{cont} L_{cont},
\]

which we minimize with respect to the network parameters. We introduce weighting factors \( \lambda_{self} \) and \( \lambda_{cont} \) to balance the impact of individual loss terms.
We first train a network only with $L_{base}$, which we use to estimate pseudo labels. Then, our network is re-initialized and re-trained with the full loss $L_{all}$. The contrastive loss operates on both labeled target and pseudo-labeled target data, where we use a lower weight $\lambda_{cont}$ for the one operating on pseudo-labeled data. Note again that our training approach does not require adversarial learning objectives for domain adaptation.

### 3.5. Implementation Details

We use DeepLabV2 [3] as our semantic segmentation network $F_{sc}$, with the encoder $E$ being a ResNet101 [14] backbone. This choice enables a fair comparison with prior works [45, 43, 49]. The decoder network $D_{la}$, which extracts the latent variables for our contrastive loss, is more sophisticated. It uses an average pooling layer followed by a two-layer perceptron to project the feature patches $f_e$ of the encoder $E$ (see Eq. (1)) into the latent space $\hat{z}$ in which the contrastive loss Eq. (7) is computed. We add decoder $D_{la}$ to multiple intermediate layers as suggested by previous work [31]. Note that $D_{la}$ is only required at training time.

To improve our overall adaptation quality, we employ the recently proposed Fourier Domain Adaptation (FDA) method [49]. It translates source domain images to the target domain by swapping the low-frequency component of the spectrum of the source image with that of a randomly selected target one. The strength of this approach is that the translation is very simple as it happens directly in the image space without any deep neural network. Thus, the source domain images we use in Eq. (2) are translated using FDA.

We use the following hyper-parameters: As in [49], we set $\lambda_{ent}$ to 0.005 and $\lambda_{self}$ to 1.0. We then tested different values of $\lambda_{cont}$ in Eq. (10). We set it to $1e^{-3}$ for target labels sampled from ground truth annotation and to $1e^{-4}$ for pseudo labels as mentioned in Sec. 3.4.

The temperature parameter $\tau$ in Eq. (7) is set to 0.07 during training and the patch size for the contrastive loss is set to $64 \times 32$ pixels in image space. We use $\alpha = 3$ and $\beta = 70$ as thresholds to define positive and negative patch pairs. Our model is trained using SGD with initial learning rate $2.5e^{-4}$ and adjusted according to the ‘poly’ learning rate scheduler with a power of 0.9 and weight decay 0.0005, following [49].

### 4. Experiments

#### 4.1. Baselines

We evaluate our proposed method on both unsupervised (UDA) and semi-supervised (SSDA) domain adaptation. ASS [45] is the only semantic segmentation work we know of that operates in the SSDA setting, i.e., assumes full annotation in the source domain and partial annotations in target domain. For a fair comparison against other state-of-the-art methods, we extend the following baselines to handle both UDA and SSDA setups:

- FDA [49] translates source domain images to the target style by swapping low-frequency values in Fourier space and leveraging self-training to refine the estimation.

- MinEnt [43] minimizes an entropy loss to penalize low-confidence predictions in the target domain.

- AdvEnt [43] minimizes the same entropy loss as MinEnt and also performs structure adaptation from source to target domain in an adversarial setting.

- Universal [20] introduces a pixel-level entropy regularization scheme to perform feature alignment among multiple domains.

The first three were designed for UDA while the fourth performs SSDA but assumes partial annotations in both domains.

To also test FDA, MinEnt, and AdvEnt in a semi-supervised context, we modified them to also leverage annotated target domain data. To this end, we train them by minimizing their original objective function along with the loss from Eq. (3) for supervised target domain data. For Universal, we used all the source domain annotations, replaced the original network by the same DeepLabV2 [3] network as for all the other baselines and kept the rest of the method as in the original work, which boosted its performance and makes the comparison fair.

#### 4.2. Datasets

To compare these baselines against our approach, we use one real-world dataset and two synthetic ones and follow the same protocols as in these earlier methods for a fair comparison. They are:

- **CityScapes** [8] is a real-world benchmark dataset for semantic segmentation featuring driving scenes and dense manual annotations. We use the 2,975 images from the training set as the target domain for training. We test on the 500 validation images and resize the images to $1024 \times 512$ without any random cropping, as in [48, 49].

- **GTA5** [33] contains 24,966 synthetic images captured from a video game. As in [48, 49], we resize the images to $1280 \times 720$ and randomly crop them to $1024 \times 512$ during training. The original dataset features 33 different categories of pixel-wise semantic labels but we only use the 19 classes that are shared with CityScapes as on our baselines [43, 45, 49].
SYNTHIA [35] is also a synthetic dataset. As others [43, 45, 49], we use the SYNTHIA-RAND-CITYSCAPES subset which comprises 9, 400 annotated images. During training, we also randomly crop the image to $1024 \times 512$ and evaluate with 13 classes to follow the standard protocol.

We use CityScapes as the target domain and either GTA5 or SYNTHIA as the source domain. We refer to the two resulting tasks as GTA5→CityScapes and SYNTHIA→CityScapes. We use all the labels from the source domain and either none for the target domain or only a handful, which we define as the unsupervised and semi-supervised cases, respectively.

4.3. Comparative Results

In this section, we compare our results against the baselines and report the results in Tables 1 and 2 as a function of the number of annotated images in the target domain. 0 corresponds to the UDA setting while 50, 100, 200, 500 and 1000 denote SSDA as in [45]. We provide qualitative results in Fig. 5.

Our approach outperforms others in overall mIoU and in most individual categories. ASS [45] delivers the best performance in some categories but still does worse than our proposed method overall. Note that the performance gap is highest in the SSDA setting where we use only a small number of annotated target domain samples, such as 50. This is significant for many practical applications: Annotating 50 images is typically possible and therefore worth doing given the boost it provides.

Moreover, the ASS approach is complementary to ours and both could be used jointly. The adversarial loss of ASS could be used as an extra loss term in our approach, which is something worth exploring in future work.

4.4. Further Weakening the Annotations

To further drive down the annotation cost and increase its practical appeal, we not only restrict the number of annotated images in the target domain but also perform only partial annotations within these images.

To this end, we split the 200 labeled target domain images into $10 \times 10$ pixel blocks as in [27] and, instead of annotating all of them, we randomly annotate only a subset and fill the others with pseudo labels.
### Table 1: Results on GTA5→CityScapes
We evaluate on the 19 common classes shared between these two dataset. Our method is consistently better compared to baselines, particularly for few number of labeled target domain images.

| Method  | 0  | 50  | 100 | 200 | 500 | 1000 |
|---------|----|-----|-----|-----|-----|------|
| 
| 50  | 60.43 | 73.50 | 80.95 | 82.75 | 84.95 | 86.95 |
| 100 | 60.50 | 73.65 | 80.95 | 82.75 | 84.95 | 86.95 |
| 500 | 60.60 | 73.80 | 81.00 | 82.80 | 85.00 | 87.00 |

### Table 2: Results on SYNTHIA→CityScapes
We report mIoU over 13 classes, following the standard protocol. Note that our method consistently outperforms others. Our results can be found in supplemental material.

| Method  | 0  | 50  | 100 | 200 | 500 | 1000 |
|---------|----|-----|-----|-----|-----|------|
| MinEnt [43] | 44.20 | 53.50 | 61.50 | 62.50 | 63.50 | 64.50 |
| AdvEnt [47] | 47.60 | 54.10 | 61.60 | 62.60 | 63.60 | 64.60 |
| FDA [49] | 52.50 | 58.40 | 62.00 | 64.40 | 66.80 | 70.17 |
| ASS [45] | - | 60.70 | 62.10 | 64.80 | 69.80 | 73.00 |
| Universal [20] | - | 53.63 | 57.05 | 60.20 | 63.41 | 68.47 |
| Ours | 53.26 | 61.18 | 63.39 | 65.23 | 70.26 | 73.09 |

### 4.5. Ablation Study
#### Quality of Pseudo Labels
To analyze how the quality of the pseudo labels affects our contrastive loss, we evaluate it using pseudo labels of different quality. Let the pseudo labels generated by the model trained using 100, 200, and 500 annotated target domain images be the low, median, and high quality ones, respectively. We then use these labels to compute the contrastive loss but compute the other losses as high quality ones, respectively. We then use these labels to compute the contrastive loss but compute the other losses as

#### Training Stability
To highlight that our approach is easier to train than methods based on adversarial learning, we
compare the training loss between OURS and AdvEnt. As shown in Fig. 6, our training curve is much better behaved.

Hyper-Parameters and Design Choices. Finally, we demonstrate the impact of various hyper-parameter and design choices on our model with 50 labeled real images.

Tables 3c, 3d and 3e show the influence of hyper-parameters specific to the contrastive loss: The temperature parameter \(\tau\), the thresholds \(\alpha\) and \(\beta\) that define positive and negatives pairs of patches, and the patch size itself.

In Tab. 3f, we compare the pyramid matching algorithm of Section 3.3.2 against a simplified Exact matching scheme that computes the Hamming distance between the two flattened label patches. As expected, our more sophisticated scheme delivers better results.

In Tab. 3g, we analyze the weighting of the contrastive loss terms applied on real and pseudo ground truth, respectively. Even though both ground truth and pseudo-labels are useful, using higher weights \(\lambda_{\text{GT,cont}}\) of Eq. 10 for the ground truth labels than \(\lambda_{\text{PSE,cont}}\) for the pseudo labels is advisable.

We analyze the impact of the individual loss terms of our model in Tab. 3h with and without using FDA-adapted input images. All the proposed loss terms improve the performance in both cases. Interestingly, the gain of using FDA becomes smaller when adding the contrastive loss, indicating its potential for domain adaptation.

Finally, we compare our training strategy and use of the contrastive loss against a pre-training strategy akin to that of [2]. In Tab. 3i, OURS-PRE is similar to OURS but uses the contrastive loss only for model pre-training as opposed to using it jointly with the semantic segmentation loss term during the training phrase as we normally do. As the table shows, this is less effective.

5. Conclusion

We introduced a new domain adaptation algorithm for semantic segmentation. Our main contribution is a novel patch-wise contrastive loss that aligns image sub-regions across domains when they exhibit similar structures in label space. It enables our algorithm to outperform state-of-the-art methods both in unsupervised and semi-supervised scenarios.

We have shown that our approach naturally extends to the weakly-supervised case in which we annotate the images only partially. In future work, we will leverage this ability to implement an active learning scheme in which the image blocks to be annotated are chosen automatically. This should result in an even lower-cost and highly practical semi-supervised approach.

Acknowledgments This work was completed during an internship at Amazon Prime Air and supported in part by the Swiss National Science Foundation.
References

[1] A. Bruhn and J. Weickert. Towards Ultimate Motion Estimation: Combining Highest Accuracy with Real-Time Performance. In International Conference on Computer Vision, 2005.

[2] K. Chaitanya, E. Erdil, N. Karani, and E. Konukoglu. Contrastive learning of global and local features for medical image segmentation with limited annotations. In Advances in Neural Information Processing Systems, 2020.

[3] 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, 2017.

[4] M. Chen, H. Xue, and D. Cai. Domain Adaptation for Semantic Segmentation with Maximum Squares Loss. In International Conference on Computer Vision, 2019.

[5] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A Simple Framework for Contrastive Learning of Visual Representations. In International Conference on Learning Representations, 2020.

[6] Y. Chen, W. Li, and L. Van Gool. ROAD: Reality Oriented Adaptation for Semantic Segmentation of Urban Scenes. In Conference on Computer Vision and Pattern Recognition, 2018.

[7] Y.-H. Chen, W.-Y. Chen, Y.-T. Chen, B.-C. Tsai, Y.-C. F. Wang, and M. Sun. No More Discrimination: Cross City Adaptation of Road Scene Segmenters. In International Conference on Computer Vision, 2017.

[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 Conference on Computer Vision and Pattern Recognition, 2016.

[9] C. Doersch and A. Zisserman. Multi-task Self-Supervised Visual Learning. In International Conference on Computer Vision, 2017.

[10] A. Dosovitskiy, J.T. Springenberg, M. Riedmiller, and T. Brox. Discriminative Unsupervised Feature Learning with Convolutional Neural Networks. In Advances in Neural Information Processing Systems, 2014.

[11] T. Gupta, A. Vahdat, G. Chechik, X. Yang, J. Kautz, and D. Hoiem. Contrastive Learning for Weakly Supervised Phrase Grounding. In European Conference on Computer Vision, 2020.

[12] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality Reduction by Learning an Invariant Mapping. In Conference on Computer Vision and Pattern Recognition, 2006.

[13] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. In Conference on Computer Vision and Pattern Recognition, 2020.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In Conference on Computer Vision and Pattern Recognition, 2016.

[15] J. Hoffman, D. Wang, F. Yu, and T. Darrel. FCNs in the Wild: Pixel-level Adversarial and Constraint-based Adaptation. In arXiv:1612.02649, 2016.

[16] W. Hong, Z. Wang, M. Yang, and J. Yuan. Conditional Generative Adversarial Network for Structured Domain Adaptation. In Conference on Computer Vision and Pattern Recognition, 2018.

[17] J. Huang, S. Lu, D. Guan, and X. Zhang. Contextual-Relation Consistent Domain Adaptation for Semantic Segmentation. In European Conference on Computer Vision, 2020.

[18] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. Efros, and T. Darrell. CyCADA: Cycle-Consistent Adversarial Domain Adaptation. In International Conference on Machine Learning, 2018.

[19] X. Ji, J.F. Henriques, and A. Vedaldi. Invariant Information Clustering for Unsupervised Image Classification and Segmentation. In International Conference on Computer Vision, 2019.

[20] T. Kalluri, G. Varma, M. Chandraker, and C. V. Jawahar. Universal Semi-Supervised Semantic Segmentation. In International Conference on Computer Vision, 2019.

[21] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning, 2020.

[22] T. Kim and C. Kim. Attract, Perturb, and Explore: Learning a Feature Alignment Network for Semi-supervised Domain Adaptation. In European Conference on Computer Vision, 2020.

[23] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06), volume 2, pages 2169–2178, 2006.

[24] G. Li, G. Kang, W. Liu, Y. Wei, and Y. Yang. Content-Consistent Matching for Domain Adaptive Semantic Segmentation. In European Conference on Computer Vision, 2020.

[25] Z. Li, Q. Tran, L. Mai, Z. Lin, and Alan Yuille. Context-Aware Group Captioning via Self-Attention and Contrastive Features. In Conference on Computer Vision and Pattern Recognition, 2020.

[26] Q. Lian, F. Lv, L. Duan, and B. Gong. Constructing Self-motivated Pyramid Curriculums for Cross-Domain Semantic Segmentation: A Non-Adversarial Approach. In International Conference on Computer Vision, 2019.

[27] H. Lin, P. Upchurch, and K. Bala. Block annotation: Better image annotation with sub-image decomposition. In 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pages 5289–5299, 2019.

[28] F. Lv, T. Liang, X. Chen, and G. Lin. Cross-Domain Semantic Segmentation via Domain-Invariant Interactive Relation Transfer. In Conference on Computer Vision and Pattern Recognition, 2020.

[29] I. Misra and L. Van Der Maaten. Self-Supervised Learning of Prettext-Invariant Representations. In Conference on Computer Vision and Pattern Recognition, 2020.
[30] F. Pan, I. Shin, F. Rameau, S. Lee, and I. So. Kweon. Unsupervised Intra-domain Adaptation for Semantic Segmentation through Self-Supervision. In Conference on Computer Vision and Pattern Recognition, 2020. 2
[31] T. Park, A. A. Efros, R. Zhang, and J.-Y. Zhu. Contrastive Learning for Unpaired Image-to-Image Translation. In European Conference on Computer Vision, 2020. 2, 5
[32] S. Paul, Y.-H. Tsai, S. Schulter, A. K. Roy-Chowdhury, and M. Chandraker. Domain Adaptive Semantic Segmentation Using Weak Labels. In European Conference on Computer Vision, 2020. 2
[33] S. R. Richter, V. Vineet, S. Roth, and V. Koltun. Playing for Data: Ground Truth from Computer Games. In European Conference on Computer Vision, 2016. 5
[34] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Conference on Medical Image Computing and Computer Assisted Intervention, pages 234–241, 2015. 2
[35] 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 Conference on Computer Vision and Pattern Recognition, 2016. 6
[36] A. Rozansve, M. Salzmann, and P. Fua. Beyond Sharing Weights for Deep Domain Adaptation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 41(4):801–814, 2019. 2
[37] K. Saito, D. Kim, S. Sclaroff, T. Darrell, and Kate Saenko. Semi-supervised Domain Adaptation via Minimax Entropy. In International Conference on Computer Vision and Pattern Recognition, 2019. 2
[38] K. Saito, Y. Ushiku, T. Harada, and K. Saenko. Adversarial Dropout Regularization. In International Conference on Learning Representations, 2018. 2
[39] S. Sankaranarayanan, Y. Balaji, A. Jain, S. Nam Lim, and R. Chellappa. Learning from Synthetic Data: Addressing Domain Shift for Semantic Segmentation. In Conference on Computer Vision and Pattern Recognition, 2018. 2
[40] M. N. Subhani and M. Ali. Learning from Scale-Invariant Examples for Domain Adaptation in Semantic Segmentation. In European Conference on Computer Vision, 2020. 2
[41] Y. Tian, D. Krishnan, and P. Isola. Contrastive Multiview Coding. In European Conference on Computer Vision, 2020. 2
[42] Y.-H. Tsai, W.-C. Hung, S. Schulter, K. Sohn, M.-H. Yang, and M. Chandraker. Learning to Adapt Structured Output Space for Semantic Segmentation. In Conference on Computer Vision and Pattern Recognition, 2018. 2
[43] T.-H. Vu, H. Jain, M. Bucher, M. Cord, and P. Perez. ADVENT: Adversarial Entropy Minimization for Domain Adaptation in Semantic Segmentation. In Conference on Computer Vision and Pattern Recognition, 2019. 1, 2, 4, 5, 6, 7, 8
[44] T.-H. Vu, H. Jain, M. Bucher, M. Cord, and P. Perez. DADA: Depth-Aware Domain Adaptation in Semantic Segmentation. In International Conference on Computer Vision, 2019. 2
[45] Z. Wang, Y. Wei, R. Feris, J. Xiong, W.-M. Hwu, T. S. Huang, and H. Shi. Alleviating Semantic-level Shift: A Semi-supervised Domain Adaptation Method for Semantic Segmentation. In Conference on Computer Vision and Pattern Recognition Workshops, 2020. 1, 2, 4, 5, 6, 7
[46] Z. Wu, X. Han, Y.-L. Lin, M. Gokhan Uzunbas, T. Goldstein, S. Nam Lim, and L. S. Davis. DCAN: Dual Channel-wise Alignment Networks for Unsupervised Scene Adaptation. In European Conference on Computer Vision, 2018. 2
[47] Z. Wu, Y. Xiong, S.X. Yu, and D. Lin. Unsupervised Feature Learning via Non-Parametric Instance-level Discrimination. In Conference on Computer Vision and Pattern Recognition, 2018. 2
[48] Y. Yang, D. Lao, G. Sundaramoorthi, and S. Soatto. Phase Consistent Ecological Domain Adaptation. In Conference on Computer Vision and Pattern Recognition, 2020. 2, 5
[49] Y. Yang and S. Soatto. FDA: Fourier Domain Adaptation for Semantic Segmentation. In Conference on Computer Vision and Pattern Recognition, 2020. 1, 2, 3, 5, 6, 7
[50] M. Ye, X. Zhang, P. C. Yuen, and S.-F. Chang. Unsupervised Embedding Learning via Invariant and Spreading Instance Feature. In Conference on Computer Vision and Pattern Recognition, 2019. 2
[51] X. Zhu, H. Zhou, C. Yang, J. Shi, and D. Lin. Penalizing Top Performers: Conservative Loss for Semantic Segmentation Adaptation. In European Conference on Computer Vision, 2018. 2
[52] C. Zhuang, A. L. Zhai, and D. Yamins. Local Aggregation for Unsupervised Learning of Visual Embeddings. In Conference on Computer Vision and Pattern Recognition, 2019. 2
[53] Y. Zou, Z. Yu, B. V. Kumar, and J. Wang. Domain Adaptation for Semantic Segmentation via Class-Balanced Self-Training. In European Conference on Computer Vision, 2018. 2