PiPa: Pixel- and Patch-wise Self-supervised Learning for Domain Adaptative Semantic Segmentation

Mu Chen
Mu.Chen@student.uts.edu.au
ReLER Lab, AAII
University of Technology Sydney
Sydney, Australia

Zhedong Zheng
zdzheng@nus.edu.sg
Sea-NExT Joint Lab, School of Computing
National University of Singapore
Singapore, Singapore

Yi Yang
yangyics@zju.edu.cn
ReLER Lab, CCAI,
Zhejiang University
Hangzhou, China

Tat-Seng Chua†
chuats@comp.nus.edu.sg
Sea-NExT Joint Lab, School of Computing
National University of Singapore
Singapore, Singapore

ABSTRACT

Unsupervised Domain Adaptation (UDA) aims to enhance the generalization of the learned model to other domains. The domain-invariant knowledge is transferred from the model trained on labeled source domain, *e.g.*, video game, to unlabeled target domains, *e.g.*, real-world scenarios, saving annotation expenses. Existing UDA methods for semantic segmentation usually focus on minimizing the inter-domain discrepancy of various levels, *e.g.*, pixels, features, and predictions, for extracting domain-invariant knowledge. However, the primary intra-domain knowledge, such as context correlation inside an image, remains under-explored. In an attempt to fill this gap, we revisit the current pixel contrast in semantic segmentation and propose a unified pixel- and patch-wise self-supervised learning framework, called **PiPa**, for domain adaptive semantic segmentation that facilitates intra-image pixel-wise correlations and patch-wise semantic consistency against different contexts. The proposed framework exploits the inherent structures of intra-domain images, which: (1) explicitly encourages learning the discriminative pixel-wise features with intra-class compactness and inter-class separability, and (2) motivates the robust feature learning of the identical patch against different contexts or fluctuations. Extensive experiments verify the effectiveness of the proposed method, which obtains competitive accuracy on the two widely-used UDA benchmarks, *i.e.*, 75.6 mIoU on GTA→Cityscapes and 68.2 mIoU on Synthia→Cityscapes. Moreover, our method is compatible with other UDA approaches to further improve the performance without introducing extra parameters.

CCS CONCEPTS

• **Computing methodologies** → **Scene understanding**; **Transfer Learning**.

MM '23, October 29-November 3, 2023, Ottawa, ON, Canada

ACM ISBN 979-8-4007-0108-5/23/10…$15.00
https://doi.org/10.1145/3581783.3611708

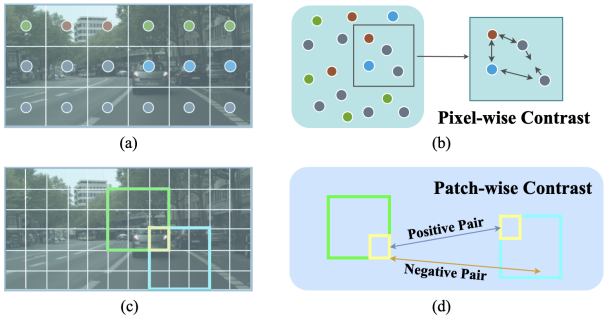

Figure 1: Different from existing works, we focus on mining the intra-domain knowledge, and argue that the contextual structure between pixels and patches can facilitate the model learning the domain-invariant knowledge in a self-supervised manner. In particular, our proposed training framework: (1) motivates intra-class compactness and inter-class dispersion by pulling closer the pixel-wise intra-class features and pushing away inter-class features within the image (see a&b at the top row); and (2) maintains the local patch consistency against different contexts, such as the yellow local patch in the green and the blue patch (see the bottom row c&d).

KEYWORDS

Unsupervised Scene Adaptation, Transfer Learning, Self-supervised Learning

ACM Reference Format:
Mu Chen, Zhedong Zheng, Yi Yang, and Tat-Seng Chua†. 2023. PiPa: Pixel- and Patch-wise Self-supervised Learning for Domain Adaptative Semantic Segmentation. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23), October 29-November 3, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3581783.3611708

1 INTRODUCTION

Prevailing models, *e.g.*, Convolutional Neural Networks (CNNs) [4, 33] and Visual Transformers [32, 84], have achieved significant

†Corresponding author

progress in computer vision applications [11, 66, 76]. But such networks are data-hungry, which usually require large-scale training datasets with pixel-level annotations. The annotation prerequisites are hard to meet in real-world scenarios. To address the shortage in the training data, one straightforward idea is to access the abundant synthetic data and the corresponding pixel-level annotations generated by computer graphics [46, 47]. However, there exist domain gaps between synthetic images and real-world images in terms of illumination, weather, and camera hyper-parameters [9, 62, 62, 69]. To minimize such a gap, researchers resort to unsupervised domain adaptation (UDA) to transfer the knowledge from labeled source-domain data to the unlabeled target-domain environment.

The key idea underpinning UDA is to learn the shared domain-invariant knowledge. One line of works, therefore, investigates techniques to mitigate the discrepancy of data distribution between source domain and target domain at different levels, such as pixel level [15, 29, 69, 77], feature level [19, 35], and prediction level [43, 53, 54, 57]. These inter-domain alignment approaches have achieved significant improvement compared to basic source-only methods, but usually overlook the intra-domain knowledge.

Another potential paradigm to address the lack of training data is self-supervised learning, which mines the visual knowledge from unlabeled data. One common optimization objective is to learn invariant representation against various augmentations, such as rotation [25], colorization [82], mixup [50] and random erasing [88]. Prior UDA works [85, 86] explored self-supervised methods to mine the domain-invariant knowledge, but the pipelines are relatively simple and only consider the prediction consistency against dropout or different network depths. Recent Segmentation and UDA work [63, 72] adopt contrastive learning methods, showing great performance. However, they focus only on pixel-level contrast without a context-aware design. We analyze existing contrastive learning methods and observe that (1) the high-level representation produced by them does not capture enough contextual information which is crucial in segmentation tasks. (2) performing contrastive learning at patch-level could prevent the model from degrading into totally ignoring the contexts. In light of the above observation, we explore the prediction consistency and contrastive learning at different effect regions. The consideration of patch-level has resulted in a larger receptive field, which makes it more suitable for segmentation tasks that require stronger contextual information. Therefore, we introduce a multi-grained Pixel- and Patch-wise self-supervised learning framework.

As the name implies, PiPa explores the pixel-to-pixel and patch-to-patch relation for regularizing the segmentation feature space. Our approach is based on two implicit priors: (1) the feature of the same-class pixels should be kept consistent with the category prototype; and (2) the feature within a patch should maintain robustness against different contexts. As shown in Figure 1, image pixels are mapped into an embedding space (Figure 1 (b) and (d)). For the **pixel-wise contrast**, we explicitly facilitate discriminative feature learning by pulling pixel embeddings of the same category closer while pushing those of different categories away (Figure 1 (b)). Considering the **patch-wise contrast**, we randomly crop two image patches with an overlapping region (the yellow region in Figure 1 (c) and (d)) from an unlabeled image. The overlapping region of the two patches should not lose its spatial information and maintain the prediction consistency even against two different contexts. The proposed method is orthogonal to other existing domain-alignment works. We re-implement two competitive baselines, and show that our framework consistently improves the segmentation accuracy over other existing works.

Our contributions are as follows: (1) Different from existing works on inter-domain alignment, we focus on mining domain-invariant knowledge from the original domain in a self-supervised manner. We propose a unified Pixel- and Patch-wise self-supervised learning framework to harness both pixel- and patch-wise consistency against different contexts, which is well-aligned with the segmentation task. (2) Our self-supervised learning method does not require extra annotations, and is compatible with other existing UDA frameworks. The effectiveness of PiPa has been tested by extensive ablation studies, and it achieves competitive accuracy on two commonly used UDA benchmarks, namely 75.6 mIoU on GTA→Cityscapes and 68.2 mIoU on Synthia→Cityscapes.

2 RELATED WORK

2.1 Unsupervised Domain Adaptation

Pioneering UDA works [15, 68] propose to transfer the visual style of the source domain data to the target domain using CycleGAN [93]. Later UDA methods can mainly be grouped into two categories according to the technical routes: adversarial training [36, 37, 43, 53, 57, 60, 77] and self-training [24, 39, 52, 81, 89, 94, 95]. Adversarial training methods aim to learn domain-invariant knowledge based on adversarial domain alignment. For instance, Tsai *et al.*[53] and Luo *et al.*[37] learn domain-invariant representations based on a min-max adversarial optimization game. However, as shown in [87], unstable adversarial training methods usually lead to suboptimal performance. Another line of work harnesses self-training to create pseudo labels for the target domain data using the model trained by labeled source domain data. Pseudo labels can be pre-computed either offline [77, 94] or generated online [16, 52]. Due to considerable discrepancies in data distributions between two domains, pseudo labels inevitably contains noise. To decrease the influence of faculty labels, Zou *et al.*[94, 95] adopts pseudo labels with high confidence. Taking one step further, Zheng *et al.*[85] conducts the domain alignment to create reliable pseudo labels. Furthermore, some variants leverage specialized sampling [39] and uncertainty [86] to learn from the noisy pseudo labels. In addition to the two mainstream practices mentioned above, researchers also conducted extensive attempts such as entropy minimization [5, 57], image translation [10, 75], Graph Network [65] and combining adversarial training and self-training [29, 59, 87]. Source-free domain adaptation, although a relatively recent concept, has been extensively studied across various fields [21, 28, 78, 79]. Recently, Pan *et al.*[44] minimizes the intra-domain discrepancy by separating the target domain into an easy and hard split using an entropy-based ranking function. Yan *et al.*[74] conducts the inter-domain adaptation between the source and target domain by treating each pixel as an instance. Different from the above-mentioned works, we focus on further mining the domain-invariant knowledge in a self-supervised manner. We harness the pixel- and patch-wise contrast, which is well aligned with the local context-focused semantic segmentation task. The proposed method is orthogonal with

the above-mentioned approaches, and thus is complementary with existing ones to further boost the result.

2.2 Contrastive Learning

Contrastive learning is one of the most prominent unsupervised representation learning methods [6, 7, 13, 42, 70], which contrasts similar (positive) data pairs against dissimilar (negative) pairs, thus learning discriminative feature representations. For instance, Wu *et al.* [70] learn feature representations at the instance level. He *et al.* [13] match encoded features to a dynamic dictionary which is updated with a momentum strategy. Chen *et al.*[6] proposes to engender negative samples from large mini-batches. In the domain adaptative image classification, contrastive learning is utilized to align feature space of different domains [23, 40].

A few recent studies utilize contrastive learning to improve the performance of semantic segmentation task [22, 30, 55, 63, 64, 72]. For example, Wang *et al.*[64] have designed and optimized a self-supervised learning framework for better visual pre-training. Gansbeke *et al.*[55] applies contrastive learning between features from different saliency masks in an unsupervised setting. Recently, Huang *et al.*[20] tackles UDA by considering instance contrastive learning as a dictionary look-up operation, allowing learning of category-discriminative feature representations. Xie *et al.*[71] presents a semantic prototype-based contrastive learning method for fine-grained class alignment. Other works explore contrastive learning either in a fully supervised manner [63, 72] or in a semi-supervised manner [1, 26, 91]. For example, Wang *et al.*[63] uses pixel contrast in a fully supervised manner in semantic segmentation. But most methods above either target image-wise instance separation or tend to learn pixel correspondence alone. Different from existing works, we introduce a multi-grained self-supervised learning framework to formulate pixel- and patch-wise contrast in a similar format but at different effect regions. The unified self-supervised learning on both pixel and patch are complementary to each other, and can mine the domain-invariant context feature.

3 METHODS

We first introduce the problem definition and conventional segmentation losses for semantic segmentation domain adaptation. Then we shed light on the proposed component of our framework PiPa, *i.e.*, Pixel-wise Contrast and Patch-wise Contrast, both of which work on local regions to mine the inherent contextual structures. We finally also raise a discussion on the mechanism of the proposed method.

Problem Definition. As shown in Figure 2, given the source-domain synthetic data $X^S = \{x_u^S\}_{u=1}^U$ labeled by $Y^S = \{y_u^S\}_{u=1}^U$ and the unlabelled target-domain real-world data $X^T = \{x_v^T\}_{v=1}^V$, where U and V are the numbers of images in the source and target domain, respectively. The label Y^S belongs to C categories. Domain adaptive semantic segmentation intends to learn a mapping function that projects the input data X^T to the segmentation prediction Y^T in the target domain.

Basic Segmentation Losses. Similar to existing works [85, 95], we learn the basic source-domain knowledge by adopting the segmentation loss on the source domain as:

$$\mathcal{L}_{ce}^S = \mathbb{E}\left[-p_u^S \log h_{cls}(g_\theta\left(x_u^S\right))\right], \tag{1}$$

where p_u^S is the one-hot vector of the label y_u^S, and the value $p_u^S(c)$ equals to 1 if $c == y_u^S$ otherwise 0. We harness the visual backbone g_θ, and 2-layer multilayer perceptrons (MLPs) h_{cls} for segmentation category prediction.

To mine the knowledge from the target domain, we generate pseudo labels $\bar{Y}^T = \{\bar{y_v}^T\}$ for the target domain data X^T by a teacher network $g_{\bar{\theta}}$ [52, 90], where $\bar{y_v}^T = argmax(h_{cls}g_{\bar{\theta}}(x_v^T))$. In practice, the teacher network $g_{\bar{\theta}}$ is set as the exponential moving average of the weights of the student network g_θ after each training iteration [51, 87]. Considering that there are no labels for the target-domain data, the network g_θ is trained on the pseudo label $\bar{y_v}^T$ generated by the teacher model $g_{\bar{\theta}}$. Therefore, the segmentation loss can be formulated as:

$$\mathcal{L}_{ce}^T = \mathbb{E}\left[-\bar{p_v}^T \log h_{cls}(g_\theta\left(x_v^T\right))\right], \tag{2}$$

where $\bar{p_v}^T$ is the one-hot vector of the pseudo label $\bar{y_v}^T$. We observe that pseudo labels inevitably introduce noise considering the data distribution discrepancy between two domains. Therefore, we set a threshold that only the pixels whose prediction confidence is higher than the threshold are accounted for the loss. In practice, we also follow [16, 52] to mix images from both domains to facilitate stable training. Specifically, the label $\bar{y}^{Mix}$ is generated by copying the random 50% categories in y^S and pasting such class areas to the target-domain pseudo label $\bar{y}^T$. Similarly, we also paste the corresponding pixel area in x^S to the target-domain input x^T as x^{Mix}. Therefore, the target-domain segmentation loss is updated as:

$$\mathcal{L}_{ce}^T = \mathbb{E}\left[-\bar{p_v}^{Mix} \log h_{cls}(g_\theta\left(x_v^{Mix}\right))\right], \tag{3}$$

where $\bar{p_v}^{Mix}$ is the probability vector of the mixed label $\bar{y_v}^{Mix}$. Since we deploy the copy-and-paste strategy instead of the conventional mixup [80], the mixed labels are still one-hot.

Multi-grained Contrast in different effect regions. We note that the above-mentioned segmentation loss does not explicitly consider the inherent context within the image, which is crucial to the local-focused segmentation task. Therefore, we study the feasibility of self-supervised learning in mining intra-domain knowledge for domain adaptive semantic segmentation tasks. In this work, we revisit the current pixel-wise contrast in semantic segmentation [63] and explore the joint training mechanism of contrastive learning on both pixel- and patch-level effect regions. To this end, we introduce a unified multi-grained contrast including patch-wise contrast to enhance the consistency within a local patch.

In the **pixel-wise** effect region, given the labels of each pixel y^S, we regard image pixels of the same class C as positive samples and the rest pixels in x^S belonging to the other classes are the negative samples. The pixel-wise contrastive loss can be derived as:

$$\mathcal{L}_{\text{Pixel}} = -\sum_{C(i)=C(j)} \log \frac{r\left(e_i, e_j\right)}{\sum_{k=1}^{N_{pixel}} r\left(e_i, e_k\right)}, \tag{4}$$

where e is the feature map extracted by the projection head $e = h_{pixel}g_\theta(x)$, and N_{pixel} is the number of pixels. e_i denotes the i-th feature on the feature map e. r denotes the similarity between the two pixel features. In particular, we deploy the exponential cosine similarity $r\left(e_i, e_j\right) = \exp\left(s\left(e_i, e_j\right)/\tau\right)$, where s is cosine similarity between two pixel features e_i and e_j, and τ is the temperature. As

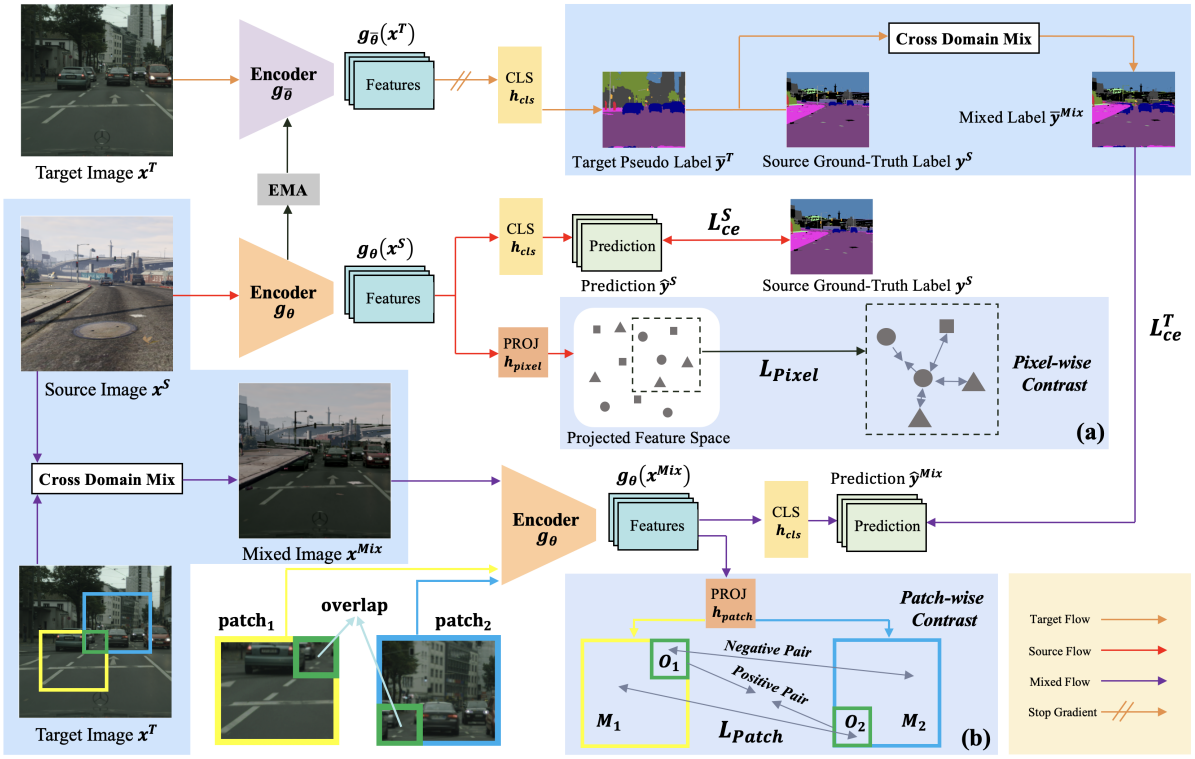

Figure 2: A brief illustration of our unified multi-grained self-supervised learning Framework (PiPa). Given the labeled source data $\left\{\left(x^S, y^S\right)\right\}$, we calculate the segmentation prediction $\hat{y}^S$ with the backbone g_θ and the classification head h_{cls}, supervised by the basic segmentation loss L_{ce}^S. During training, we leverage the moving averaged model $g_{\bar{\theta}}$ to estimate the pseudo label $\bar{y}^T$ to craft the mixed label $\bar{y}^{Mix}$ based on the category. According to the mixed label, we copy the corresponding regions as the mixed data x^{Mix}. We also deploy the model g_θ and the head h_{cls} to obtain the mixed prediction $\hat{y}^{Mix}$ supervised by L_{ce}^T. Except for the above-mentioned basic segmentation losses, we revisit current pixel contrast and propose a unified multi-grained Contrast. In (a), we regularize the pixel embedding space by computing pixel-to-pixel contrast: impelling positive-pair embeddings closer, and pushing away the negative embeddings. In (b), we regularize the patch-wise consistency between projected patch O_1 and O_2. Similarly, we harness the patch-wise contrast, which pulls positive pair, *i.e.*, two features at the same location of O_1 and O_2 closer, while pushing negative pairs apart, *i.e.*, any two features in $M_1 \cup M_2$ at different locations. During inference, we drop the two projection heads h_{patch} and h_{pixel} and only keep g_θ and h_{cls}.

shown in Figure 2, with the guide of pixel-wise contrastive loss, the pixel embeddings of the same class are pulled close and those of the other classes are pushed apart, which promotes intra-class compactness and inter-class separability.

In the **patch-wise** effect region, in particular, given unlabeled target image x^T, we also leverage the network g_θ to extract the feature map of two partially overlapping patches. The cropped examples are shown at the bottom of Figure 2. We deploy an independent head h_{patch} with 2-layer MLPs to further project the output feature maps to the embedding space for comparison. As shown in Figure 2 module (b), overlapping region $\mathbf{O}_1$ and $\mathbf{O}_2$ denote the same green area in the original image. In practice, we first randomly select the region $\mathbf{O}$ and then sample two neighbor patches $\mathbf{M}$ covering $\mathbf{O}$. We use $\mathbf{M}$ to denote the entire patch **including** $\mathbf{O}$. We argue that the output features of the overlapping region should be invariant to the contexts. Therefore, we encourage that each feature in $\mathbf{O}_1$ to be consistent with the corresponding feature of the same location in $\mathbf{O}_2$. Similar to pixel-wise contrast, as shown in Figure 2 module (b), we regard two features at the same position of $\mathbf{O}_1$ and $\mathbf{O}_2$ as positive pair, and any two features in $\mathbf{M}_1$ and $\mathbf{M}_2$ at different positions of the original image are treated as a negative pair. Given a target-domain input x^T, the patch-wise contrast loss can be formulated as:

$$\mathcal{L}_{\text{Patch}} = -\sum_{O_1(i)=O_2(j)} \log \frac{r\left(f_i, f_j\right)}{\sum_{k=1}^{N_{patch}} r\left(f_i, f_k\right)}, \tag{5}$$

where f is the feature map extracted by the projection head $f = h_{patch} g_\theta(x)$, and N_{patch} is the number of pixels in $M_1 \cup M_2$. i is the pixel index in the patch $\mathbf{M}_1$, and j is for $\mathbf{M}_2$. $\mathbf{O}_1(i)$ denotes the location in the overlapping region $\mathbf{O}_1$. $\mathbf{O}_1(i) = \mathbf{O}_2(j)$ denotes i and j are the same pixel (location) in the original image, as shown in Figure 4(b). f_i denotes i-th feature in the map. Similarly, r denotes the exponential function of the cosine similarity as the one in pixel

Algorithm 1 PiPa algorithm

Input: Source-domain data X^S, Source-domain labels Y^S, Target domain data X^T, segmentation network that contains segmentation encoder g_θ, classification head h_{cls}, pixel projection head h_{pixel}, patch projection head h_{patch}, the total iteration number T_{total}.

Initialize network parameter θ with ImageNet pre-trained parameters. Initialize teacher network $\bar{\theta}$ randomly
for iteration = 1 to T_{total} **do**
  $x^S, y^S \sim U$.
  $x^T \sim V$.
  $\bar{y}^T \leftarrow argmax\left(h_{cls}\left(g_{\bar{\theta}}\left(x^T\right)\right)\right)$.
  x^{Mix}, $\bar{y}^{Mix} \leftarrow$ Augmentation and pseudo label from mixing x^S, y^S, x^T and $\bar{y}^T$.
  Compute predictions
  $\hat{y}^S \leftarrow argmax\left(h_{cls}\left(g_\theta\left(x^S\right)\right)\right)$,
  $\hat{y}^{Mix} \leftarrow argmax\left(h_{cls}\left(g_\theta\left(x^{Mix}\right)\right)\right)$.
  Compute loss for the mini-batch:
  $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{ce}} + \mathcal{L}_{\text{Pixel}} + \mathcal{L}_{\text{Patch}}$.
  Compute $\nabla_\theta \mathcal{L}_{\text{total}}$ by backpropagation.
  Perform stochastic gradient descent.
  Update teacher network $\bar{\theta}$ with θ.
end for
return student network g_θ and classification head h_{cls}.

contrast. It is worth noting that we also enlarge the negative sample pool. In practice, the rest feature f_k not only comes from the union set $M_1 \cup M_2$, but also from other training images within the current batch.

Total Loss. The overall training objective is the combination of pixel-level cross-entropy loss and the proposed PiPa:

$$\mathcal{L}_{total} = \mathcal{L}_{\text{ce}}^S + \mathcal{L}_{\text{ce}}^T + \alpha \mathcal{L}_{\text{Pixel}} + \beta \mathcal{L}_{\text{Patch}}, \tag{6}$$

where α and β are the weights for pixel-wise contrast $\mathcal{L}_{\text{Pixel}}$ and patch-wise contrast $\mathcal{L}_{\text{Patch}}$, respectively. We summarize the pipeline of PiPa in Algorithm 1.

Discussion. 1. Correlation between Pixel and Patch Contrast. Both pixel and patch contrast are derived from instance-level contrastive learning and share a common underlying idea, *i.e.*, contrast, but they work at different effect regions, *i.e.*, pixel-wise and patch-wise. The pixel contrast explores the pixel-to-pixel category correlation over the whole image, while patch-wise contrast imposes regularization on the semantic patches from a local perspective. Therefore, the two kinds of contrast are complementary and can work in a unified way to mine the intra-domain inherent context within the data. **2. What is the advantage of the proposed framework?** Traditional UDA methods focus on learning shared inter-domain knowledge. Differently, we are motivated by the objectives of UDA semantic segmentation in a bottom-up manner, and thus leverage rich pixel correlations in the training data to facilitate intra-domain knowledge learning. By explicitly regularizing the feature space via PiPa, we enable the model to explore the inherent intra-domain context in a self-supervised setting, *i.e.*, pixel-wise and patch-wise, without extra parameters or annotations. Therefore, PiPa could be effortlessly incorporated into existing UDA approaches to achieve better results without extra overhead during testing. **3. Difference from conventional contrastive learning.** Conventional contrastive learning methods typically tend to perform contrast in the instance or pixel level alone [20, 63, 70]. We formulate pixel- and patch-wise contrast in a similar format but focus on the local effect regions within the images, which is well aligned with the local-focused segmentation task. We show that the proposed local contrast, *i.e.*, pixel- and patch-wise contrasts, regularizes the domain adaptation training and guides the model to shed more light on the intra-domain context. Our experiment also verifies this point that pixel- and patch-wise contrast facilitates smooth edges between different categories and yields a higher accuracy on small objects.

4 EXPERIMENT

4.1 Implementation Details

Datasets. We evaluate the proposed method on GTA → Cityscapes and SYNTHIA → Cityscapes, following common UDA protocols [2, 16, 17, 52, 61]. The target dataset Cityscapes, collected from the real-world street-view images, contains 2,975 unlabeled images for training, 500 images for validation, and 1525 images for testing. We report the results on Cityscapes validation set for comparisons.

Structure Details. Following recent SOTA UDA setting [16, 72, 90], our network consists of a SegFormer MiT-B5 backbone [16, 73] pretrained on ImageNet-1k [8] and several MLP-based heads, *i.e.*, h_{cls} , h_{pixel} and h_{patch} , which contains two fully-connected (fc) layers and ReLU activation between two fc layers. Note that the self-supervised projection heads h_{pixel} and h_{patch} are only applied at training time and are removed during inference, which does not introduce extra computational costs in deployment.

Implementation details. We train the network with batch size 2 for 60k iterations with a single NVIDIA RTX 6000 GPU. We adopt AdamW [34] as the optimizer, a learning rate of 6×10^{-5}, a linear learning rate warmup of 1.5k iterations and the weight decay of 0.01. Following [72, 90], the input image is resized to 1280×720 for GTA and 1280×760 for SYNTHIA, with a random crop size of 640×640. For the patch-wise contrast, we randomly resize the input images by a ratio between 0.5 and 2, and then randomly crop two patches of the size 720×720 from the resized image and ensure the Intersection-over-Union(IoU) value of the two patches between 0.1 and 1. We utilize the same data augmentation *e.g.*, color jitter, Gaussian blur and ClassMix [41] and empirically set pseudo labels threshold 0.968 following [52]. The exponential moving average parameter of the teacher network is 0.999. The hyperparameters of the loss function are chosen empirically $\alpha = \beta = 0.1$. **Reproducibility.** The code is based on Pytorch [45]. The code is available at "https://github.com/chen742/PiPa".

4.2 Comparisons with State-of-the-art Methods

GTA → Cityscapes. Generally, our PiPa yields a significant improvement over the transformer-based models DAFormer[16] and HRDA[17]. Particularly, PiPa achieves 71.7 mIoU, which outperforms DAFormer by a considerable margin of +3.4 mIoU. Additionally, when applying PiPa to HRDA, which is a strong baseline that adopts high-resolution crops, we increase +1.8 mIoU and achieve

Table 1: Quantitative comparison with previous UDA methods on GTA → Cityscapes. We present pre-class IoU and mIoU. The best accuracy in every column is in bold.

Method	Road	SW	Build	Wall	Fence	Pole	TL	TS	Veg.	Terrain	Sky	PR	Rider	Car	Truck	Bus	Train	Motor	Bike	mIoU
CyCADA [15]	86.7	35.6	80.1	19.8	17.5	38.0	39.9	41.5	82.7	27.9	73.6	64.9	19.0	65.0	12.0	28.6	4.5	31.1	42.0	42.7
CLAN [37]	87.0	27.1	79.6	27.3	23.3	28.3	35.5	24.2	83.6	27.4	74.2	58.6	28.0	76.2	33.1	36.7	6.7	31.9	31.4	43.2
ASA [92]	89.2	27.8	81.3	25.3	22.7	28.7	36.5	19.6	83.8	31.4	77.1	59.2	29.8	84.3	33.2	45.6	16.9	34.5	30.8	45.1
SPCL [71]	90.3	50.3	85.7	45.3	28.4	36.8	42.2	22.3	85.1	43.6	87.2	62.8	39.0	87.8	41.3	53.9	17.7	35.9	33.8	52.1
DACS [52]	89.9	39.7	87.9	30.7	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
BAPA [31]	94.4	61.0	88.0	26.8	39.9	38.3	46.1	55.3	87.8	46.1	89.4	68.8	40.0	90.2	60.4	59.0	0.0	45.1	54.2	57.4
CaCo [20]	93.8	64.1	85.7	43.7	42.2	46.1	50.1	54.0	88.7	47.0	86.5	68.1	2.9	88.0	43.4	60.1	31.5	46.1	60.9	58.0
PiPa (CNN)	95.1	71.3	87.7	44.2	42.0	43.5	52.1	63.3	87.8	44.0	87.5	72.3	44.2	89.3	59.9	59.4	2.1	47.2	48.9	60.1
DAFormer [16]	95.7	70.2	89.4	53.5	48.1	49.6	55.8	59.4	89.9	47.9	92.5	72.2	44.7	92.3	74.5	78.2	65.1	55.9	61.8	68.3
CAMix [90]	96.0	73.1	89.5	53.9	50.8	51.7	58.7	64.9	90.0	51.2	92.2	71.8	44.0	92.8	78.7	82.3	70.9	54.1	64.3	70.0
DAFormer [16] + PiPa	96.1	72.0	90.3	56.6	52.0	55.1	61.8	63.7	90.8	52.6	93.6	74.3	43.6	93.5	78.4	84.2	77.3	59.9	66.7	71.7
HRDA [17]	96.4	74.4	91.0	61.6	51.5	57.1	63.9	69.3	91.3	48.4	94.2	79.0	52.9	93.9	84.1	85.7	75.9	63.9	67.5	73.8
CLUDA [56]	**97.1**	**78.0**	91.0	60.3	55.3	56.3	64.3	71.5	91.2	51.1	94.7	78.4	52.9	**94.5**	82.8	86.5	73.0	**64.2**	**69.7**	74.4
HRDA [17] + PiPa	96.8	76.3	**91.6**	**63.0**	**57.7**	**60.0**	**65.4**	**72.6**	**91.7**	**51.8**	**94.8**	**79.7**	**56.4**	94.4	**85.9**	**88.4**	**78.9**	63.5	67.2	**75.6**

Table 2: Quantitative comparison with previous UDA methods on SYNTHIA → Cityscapes. We present pre-class IoU, mIoU and mIoU*. mIoU and mIoU* are averaged over 16 and 13 categories, respectively. The best accuracy in every column is in bold.

Method	Road	SW	Build	Wall*	Fence*	Pole*	TL	TS	Veg.	Sky	PR	Rider	Car	Bus	Motor	Bike	mIoU*	mIoU
CLAN [37]	81.3	37.0	80.1	–	–	–	16.1	13.7	78.2	81.5	53.4	21.2	73.0	32.9	22.6	30.7	47.8	–
SP-Adv [49]	84.8	35.8	78.6	–	–	–	6.2	15.6	80.5	82.0	66.5	22.7	74.3	34.1	19.2	27.3	48.3	–
ASA [92]	**91.2**	48.5	80.4	3.7	0.3	21.7	5.5	5.2	79.5	83.6	56.4	21.0	80.3	36.2	20.0	32.9	49.3	41.7
DADA [58]	89.2	44.8	81.4	6.8	0.3	26.2	8.6	11.1	81.8	84.0	54.7	19.3	79.7	40.7	14.0	38.8	49.8	42.6
CCM [27]	79.6	36.4	80.6	13.3	0.3	25.5	22.4	14.9	81.8	77.4	56.8	25.9	80.7	45.3	29.9	52.0	52.9	45.2
BL [29]	86.0	46.7	80.3	–	–	–	14.1	11.6	79.2	81.3	54.1	27.9	73.7	42.2	25.7	45.3	51.4	–
DAFormer [16]	84.5	40.7	88.4	41.5	6.5	50.0	55.0	54.6	86.0	89.8	73.2	48.2	87.2	53.2	53.9	61.7	67.4	60.9
CAMix [90]	87.4	47.5	88.8	–	–	–	55.2	55.4	87.0	91.7	72.0	49.3	86.9	57.0	57.5	63.6	69.2	–
DAFormer [16] + PiPa	87.9	48.9	88.7	45.1	4.5	53.1	59.1	58.8	87.8	92.2	75.7	49.6	88.8	53.5	58.0	62.8	70.1	63.4
HRDA [17]	85.2	47.7	88.8	49.5	4.8	57.2	65.7	60.9	85.3	92.9	79.4	52.8	89.0	64.7	63.9	64.9	72.4	65.8
CLUDA [56]	87.7	46.9	**90.2**	49.0	**7.9**	**59.5**	66.9	58.5	88.3	94.6	**80.1**	57.1	89.8	68.2	**65.5**	65.8	73.8	67.2
HRDA [17] + PiPa	**88.6**	**50.1**	90.0	**53.8**	7.7	58.1	**67.2**	**63.1**	**88.5**	**94.5**	79.7	**57.6**	**90.8**	**70.2**	65.1	**66.9**	**74.8**	**68.2**

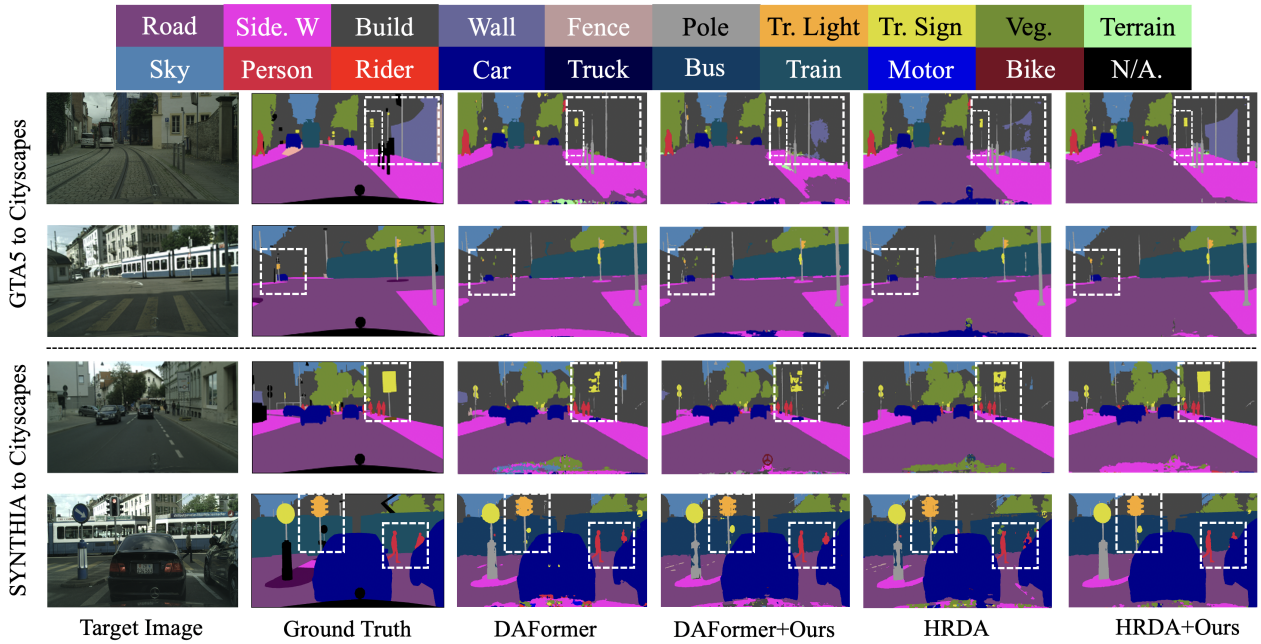

Figure 3: Qualitative results on GTA → Cityscapes and SYNTHIA → Cityscapes. From left to right: Target Image, Ground Truth, the visual results predicted by DAFormer, DAFormer + Ours (PiPa), HRDA, HRDA + Ours (PiPa). We deploy the white dash boxes to highlight different prediction parts.

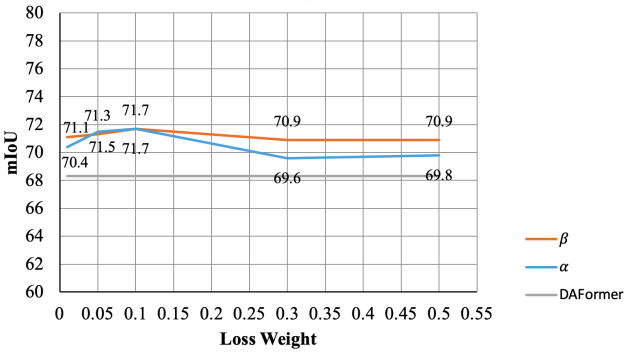

Figure 4: Ablation study on Loss Weights α and β.

the state-of-the-art performance of 75.6 mIoU, verifying the effectiveness of the proposed method that introduces a unified and multi-grained self-supervised learning algorithm in UDA task. Furthermore, PiPa achieves leading IoU of almost all classes on GTA → Cityscapes, including several small-scale objectives such as Fence, Pole, Wall and Training Sign. Particularly, we increase the IoU of the Fence by +6.2 from 51.5 to 57.7 IoU. The IoU performance of PiPa verifies our motivation that the exploration of the inherent structures of intra-domain images indeed helps category recognition, especially for challenging small objectives.

SYNTHIA → Cityscapes. As revealed in Table 2, PiPa also achieves remarkable mIoU and mIoU* (13 most common categories) performance on SYNTHIA → Cityscapes, increasing +2.5 and +2.4 mIoU compared with DAFormer [16] and HRDA [17], respectively.

Qualitative results. In Figure 3, we visualize the segmentation results and the comparison with previous strong methods DAFormer [16], HRDA [17], and the ground truth on both GTA → Cityscapes and SYNTHIA → Cityscapes benchmarks. The results highlighted by white dash boxes show that PiPa is capable of segmenting minor categories such as 'wall', 'traffic sign' and 'traffic light'. It is also noticeable that PiPa predicts smoother edges between different categories, *e.g.*, 'person' in the fourth row of Figure 3. We think it is because the proposed method explicitly encourages patch-wise consistency against different contexts, which facilitates the prediction robustness on edges.

4.3 Ablation Studies and Further Analysis

Effect of Pixel-wise Contrast and Patch-wise Contrast. We evaluate the effectiveness of the two primary components, *i.e.*, Pixel-wise Contrast and Patch-wise Contrast in the proposed PiPa and investigate how the combination of two contrasts contributes to the final performance on GTA → Cityscapes. For a fair comparison, we apply the same experimental settings and hyperparameters. We first reproduce the baseline DAFormer [16], which yields a competitive mIoU of 68.4. As shown in the Table 3, we could observe: (1) Both Patch Contrast and Pixel Contrast individually could lead to +1.4 mIoU and +2.3 mIoU improvement respectively, verifying the effectiveness of exploring the inherent contextual knowledge. (2) The two kinds of contrasts are complementary to each other. The proposed method successfully mines the multi-level knowledge by combining the two kinds of contrast. When applying both losses, our PiPa further improves the network performance to 71.7 mIoU, surpassing the model that deploys only one kind of contrast by a clear margin. The second baseline model is HRDA [17]. The observation is consistent with DAFormer. Using either pixel or patch loss could increase the performance, but jointly training them in a unified framework leads to the best results. Since HRDA introduces High Resolution (HR) and Low Resolution (LR) features, to effectively introduce Pixel-wise contrast and Patch-wise contrast in HRDA [17], we conducted experiments on both HR and LR features as shown in Table 4. It is shown that training with HR features results in higher performance.

Table 3: Ablation study on the effect of Pixel-wise Contrast and Patch-wise Contrast on GTA → Cityscapes based on two competitive baselines DAFormer[16] and HRDA[17].

Method	$\mathcal{L}_{\text{Pixel}}$	$\mathcal{L}_{\text{Patch}}$	mIoU	ΔmIoU
DAFormer[16]			68.4	–
Patch Contrast		✓	69.8	+1.4
Pixel Contrast	✓		70.7	+2.3
PiPa	✓	✓	71.7	+3.3
HRDA[17]			73.8	–
Patch Contrast		✓	74.7	+0.9
Pixel Contrast	✓		74.9	+1.1
PiPa	✓	✓	75.6	+1.8

Effect of the loss weight. We conduct loss weight sensitivity analysis on GTA → Cityscapes. Specifically, we change the weights α and β of the two kinds of contrasts in Eq 6, respectively. As shown in Figure 4, we can observe that both pixel-wise and patch-wise contrast are not sensitive to the relative weight. PiPa keeps outperforming the competitive DAFormer baseline of 68.3 mIoU in all compositions of loss weights. When applying the proposed method to an unseen environment, $\alpha = 0.1, \beta = 0.1$ can be a good initial weight to start.

Effect of the patch crop size. For the patch contrast, the size of the patch also affects the number of negative pixels and training difficulty. As shown in Table 5, we gradually increase the patch size. We observe that larger patch generally obtain better performance since it contains more diverse contexts. There are two main advantages when increasing the patch size: (1) In larger patches, we could include more "hard negative" pixels for contrastive learning; (2) In larger patches, we have a larger receptive field, which could include contextual cues for bigger objects, such as trains. It is also worth noting that if the patch size is too large (like 960), the overlapping area can be larger than the non-overlapping area, which also may compromise the training.

Table 4: Effect of different crop types in HRDA [17].

Method	mIoU
LR Crops	75.1
HR Crops	75.6

Table 5: Effect of the patch crop size.

Crop Size	mIoU
480 × 480	70.4
600 × 600	71.0
720 × 720	71.7
900 × 900	70.9

Sensitivity of the pseudo label threshold. Since the target annotation is not available in unsupervised domain adaptation, a hard threshold beta is used to eliminate low-confidence pixel predictions from the predicted label. We conducted additional experiments on the threshold and found that within the range of 0.9-0.99, the

Table 6: Sensitivity analysis of the pseudo label threshold.

Threshold	0.6	0.7	0.8	0.9	0.95	0.968	0.99
mIoU	66.3	68.9	69.4	70.8	71.2	71.7	71.4

Table 7: Results on GTA5 + SYNTHIA → Cityscapes.

Base	Multi Src.	Multi Src + PiPa
52.1	54.2	56.1

Table 8: Quantitative comparison with previous UDA methods on Cityscapes → ACDC. The performance is provided as mIoU in % and the best result is in bold.

Method	Architecture	mIoU
ADVENT [57]	DeepLabv2	32.7
AdaptSegNet [53]	DeepLabv2	32.7
BDL [29]	DeepLabv2	37.7
CLAN [37]	DeepLabv2	39.0
FDA [77]	DeepLabv2	45.7
MGCDA [48]	DeepLabv2	48.7
DANNet [67]	DeepLabv2	50.0
DAFormer [16]	Transformer	55.4
DAFormer [16] + PiPa	Transformer	**58.6 (+3.2)**
MIC [18]	Transformer	59.2
MIC [18] + PiPa	Transformer	**61.1 (+1.9)**
Refign [3]	Transformer	65.5
Refign [3] + PiPa	Transformer	**66.4 (+0.9)**

DAFormer + PiPa results were not sensitive to the beta in Table 6. We set the threshold to 0.968 to obtain optimal results following previous self-training works [16, 52].

Multi source domain setting. By incorporating multi-source domain data, the model can be trained to be more robust to the unlabelled target environment [12, 83]. We first adopt previous work MADAN [83] as our baseline, which reaches 41.4 mIoU on GTA5 + SYNTHIA → Cityscapes. MADAN + PiPa increases the performance to 44.1 mIoU. Then we adopt a self-training baseline DACS [52], which achieves a mIoU of 52.1 (Only GTA) as shown in Table 7. By incorporating additional source-domain data, the model's performance improves to 54.2 mIoU. Our proposed method further improves the model's performance, increasing the mIoU from 54.2 to 56.1 mIoU, demonstrating consistent improvement over various baselines.

Ablation study on Normal-to-Adverse setting. ACDC is a large dataset with 4,006 images containing four common adverse conditions: fog, nighttime, rain and snow. In Cityscapes → ACDC, the knowledge is transferred from the source domain under normal visual conditions, *i.e.*, at daytime and in clear weather to adverse visual conditions. The quantitative comparisons are shown in Table 8. We can observe that our PiPa yields a significant improvement over the previous methods. Particularly, PiPa achieves 58.6 mIoU, which outperforms DAFormer by +3.2 mIoU, which demonstrates the competitive generalization ability of PiPa in adverse visual conditions. When plugging on recent works MIC [18] and Refign [3], PiPa shows consistent improvement.

Oxford RobotCar dataset [38] contains 894 training images with 9 classes and is collected during rainy and cloudy weather conditions, presenting a challenge due to the noisy variants introduced by such illumination conditions. We observe that the proposed method also has achieved the competitive results on Cityscapes → Oxford-Robot based on MRNet [85] and Uncertainty [86], reaching 1.8 and 2.1 mIoU increase respectively.

Table 9: Quantitative Results on Cityscapes → Oxford-Robot [38]. The performance is provided as mIoU in % and the best result is in bold.

Method	road	sidewalk	building	light	sign	sky	person	automobile	two-wheel	mIoU
MRNet [85]	95.9	73.5	86.2	69.3	31.9	87.3	57.9	88.8	61.5	72.5
MRNet + PiPa	**96.9**	75.1	88.0	69.9	36.5	88.8	61.5	89.1	**63.1**	74.3
Uncertainty [86]	95.9	73.7	87.4	72.8	**43.1**	88.6	61.7	89.6	57.0	74.4
Uncertainty + PiPa	96.0	**76.2**	**93.3**	**73.3**	42.5	**90.9**	**65.4**	**91.1**	59.5	**76.5**

Table 10: Quantitative result on a CNN-based architecture. The performance is provided as mIoU in %.

Src-Only	Baseline	Baseline+PiPa
34.3	54.2	60.1 (+5.9)

Table 11: Further study on advanced architecture. The performance is provided as mIoU in %.

Dataset	GTA-Cityscapes	SYNTHIA-Cityscapes
MIC [18]	75.9	67.3
MIC [18] + PiPa	77.3	68.9

Ablation study on CNN-based architectures. In addition to Vision Transformer-based DA architectures, we also evaluate our PiPa on the DeepLabV2 [4] baseline with ResNet-101 backbone [14]. We do not pursue the SOTA performance here, but to demonstrate the relative improvement by plugging PiPa. Therefore, we do not search optimal hyper-parameters but follow the common setting. In Table 10, we show the adaptation performance of the baseline and our PiPa on GTA5 → Cityscapes. We also provide the performance of the DeepLabV2 trained merely on the source domain data, *i.e.*, Src-Only. It can be observed that PiPa improves the UDA baseline performance of DeepLabV2 by a large margin from 54.2 mIoU to 60.1 mIoU accuracy, still remains competitive.

Further experimental results on advanced architecture. We then apply our PiPa on the advanced method MIC [18]. MIC + PiPa achieves 77.3 mIoU (1.4 higher than MIC) on GTA-Cityscapes and 68.9 mIoU (1.6 higher than MIC) on SYNTHIA-Cityscapes, showing consistent improvement. The results are shown in Table 11.

5 CONCLUSION

In this work, we focus on the exploration of intra-domain knowledge, such as context correlation inside an image for the semantic segmentation domain adaptation. We target to learn a feature space that enables discriminative pixel-wise features and the robust feature learning of the overlapping patch against variant contexts. To this end, we propose PiPa, a unified pixel- and patch-wise self-supervised learning framework, which introduces pixel-level and patch-level contrast learning to UDA. PiPa encourages the model to mine the inherent contextual feature, which is domain invariant. Experiments show that PiPa outperforms the state-of-the-art approaches and yields a competitive 75.6 mIoU on GTA→Cityscapes and 67.4 mIoU on Synthia→Cityscapes. Since PiPa does not introduce extra parameters or annotations, it can be combined with other existing methods to further facilitate the intra-domain knowledge learning. In the future, we will continue to study the proposed PiPa on relevant tasks, such as domain adaptive video segmentation and open-set adaptation *etc.*

REFERENCES

[1] Inigo Alonso, Alberto Sabater, David Ferstl, Luis Montesano, and Ana C Murillo. 2021. Semi-supervised semantic segmentation with pixel-level contrastive learning from a class-wise memory bank. In *ICCV*.

[2] Nikita Araslanov and Stefan Roth. 2021. Self-supervised augmentation consistency for adapting semantic segmentation. In *CVPR*.

[3] David Brüggemann, Christos Sakaridis, Prune Truong, and Luc Van Gool. 2023. Refign: Align and refine for adaptation of semantic segmentation to adverse conditions. In *WACV*.

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

[5] Minghao Chen, Hongyang Xue, and Deng Cai. 2019. Domain adaptation for semantic segmentation with maximum squares loss. In *ICCV*.

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *ICML*.

[7] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. 2020. Improved baselines with momentum contrastive learning. *arXiv:2003.04297* (2020).

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*.

[9] Jianwu Fang, Fan Wang, Peining Shen, Zhedong Zheng, Jianru Xue, and Tat-seng Chua. 2022. Behavioral Intention Prediction in Driving Scenes: A Survey. *arXiv:2211.00385* (2022).

[10] Shaohua Guo, Qianyu Zhou, Ye Zhou, Qiqi Gu, Junshu Tang, Zhengyang Feng, and Lizhuang Ma. 2021. Label-free regional consistency for image-to-image translation. In *ICME*.

[11] Liang Han, Pichao Wang, Zhaozheng Yin, Fan Wang, and Hao Li. 2020. Exploiting better feature aggregation for video object detection. In *ACM Multimedia*.

[12] Jianzhong He, Xu Jia, Shuaijun Chen, and Jianzhuang Liu. 2021. Multi-source domain adaptation with collaborative learning for semantic segmentation. In *CVPR*.

[13] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *CVPR*.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*.

[15] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. 2018. Cycada: Cycle-consistent adversarial domain adaptation. In *ICML*.

[16] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. 2022. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *CVPR*.

[17] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. 2022. HRDA: Context-Aware High-Resolution Domain-Adaptive Semantic Segmentation. In *ECCV*.

[18] Lukas Hoyer, Dengxin Dai, Haoran Wang, and Luc Van Gool. 2023. MIC: Masked image consistency for context-enhanced domain adaptation. In *CVPR*.

[19] Haoshuo Huang, Qixing Huang, and Philipp Krahenbuhl. 2018. Domain transfer through deep activation matching. In *ECCV*.

[20] Jiaxing Huang, Dayan Guan, Aoran Xiao, Shijian Lu, and Ling Shao. 2022. Category contrast for unsupervised domain adaptation in visual tasks. In *CVPR*.

[21] Yi Huang, Xiaoshan Yang, Ji Zhang, and Changsheng Xu. 2022. Relative alignment network for source-free multimodal video domain adaptation. In *ACM Multimedia*.

[22] Zhengkai Jiang, Yuxi Li, Ceyuan Yang, Peng Gao, Yabiao Wang, Ying Tai, and Chengjie Wang. 2022. Prototypical Contrast Adaptation for Domain Adaptive Segmentation. In *ECCV*.

[23] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. 2019. Contrastive adaptation network for unsupervised domain adaptation. In *CVPR*.

[24] Guoliang Kang, Yunchao Wei, Yi Yang, and Alex Hauptmann. 2020. Pixel-Level Cycle Association: A New Perspective for Domain Adaptive Semantic Segmentation. In *NeurIPS*.

[25] Nikos Komodakis and Spyros Gidaris. 2018. Unsupervised representation learning by predicting image rotations. In *ICLR*.

[26] Xin Lai, Zhuotao Tian, Li Jiang, Shu Liu, Hengshuang Zhao, Liwei Wang, and Jiaya Jia. 2021. Semi-supervised semantic segmentation with directional context-aware consistency. In *CVPR*.

[27] Guangrui Li, Guoliang Kang, Wu Liu, Yunchao Wei, and Yi Yang. 2020. Content-Consistent Matching for Domain Adaptive Semantic Segmentation. In *ECCV*.

[28] Xinhao Li, Jingjing Li, Lei Zhu, Guoqing Wang, and Zi Huang. 2021. Imbalanced source-free domain adaptation. In *ACM Multimedia*.

[29] Yunsheng Li, Lu Yuan, and Nuno Vasconcelos. 2019. Bidirectional learning for domain adaptation of semantic segmentation. In *CVPR*.

[30] Weizhe Liu, David Ferstl, Samuel Schulter, Lukas Zebedin, Pascal Fua, and Christian Leistner. 2021. Domain adaptation for semantic segmentation via patch-wise contrastive learning. *arXiv:2104.11056* (2021).

[31] Yahao Liu, Jinhong Deng, Xinchen Gao, Wen Li, and Lixin Duan. 2021. Bapanet: Boundary adaptation and prototype alignment for cross-domain semantic segmentation. In *ICCV*.

[32] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*.

[33] Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully convolutional networks for semantic segmentation. In *CVPR*.

[34] Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv:1711.05101* (2017).

[35] Yawei Luo, Ping Liu, Tao Guan, Junqing Yu, and Yi Yang. 2019. Significance-aware Information Bottleneck for Domain Adaptive Semantic Segmentation. In *ICCV*.

[36] Yawei Luo, Ping Liu, Liang Zheng, Tao Guan, Junqing Yu, and Yi Yang. 2021. Category-level adversarial adaptation for semantic segmentation using purified features. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021).

[37] Yawei Luo, Liang Zheng, Tao Guan, Junqing Yu, and Yi Yang. 2019. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *CVPR*.

[38] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 2017. 1 year, 1000 km: The Oxford RobotCar dataset. *The International Journal of Robotics Research* 36, 1 (2017), 3–15.

[39] Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. 2020. Instance adaptive self-training for unsupervised domain adaptation. In *ECCV*.

[40] Saeid Motiian, Marco Piccirilli, Donald A Adjeroh, and Gianfranco Doretto. 2017. Unified deep supervised domain adaptation and generalization. In *ICCV*.

[41] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. 2021. Classmix: Segmentation-based data augmentation for semi-supervised learning. In *WACV*.

[42] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv:1807.03748* (2018).

[43] Fei Pan, Inkyu Shin, Francois Rameau, Seokju Lee, and In So Kweon. 2020. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In *CVPR*.

[44] Fei Pan, Inkyu Shin, Francois Rameau, Seokju Lee, and In So Kweon. 2020. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In *CVPR*.

[45] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. (2017).

[46] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. 2016. Playing for data: Ground truth from computer games. In *ECCV*.

[47] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. 2016. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *CVPR*.

[48] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. 2020. Map-guided curriculum domain adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020).

[49] Yuhu Shan, Chee Meng Chew, and Wen Feng Lu. 2020. Semantic-aware short path adversarial training for cross-domain semantic segmentation. *Neurocomputing* 380 (2020), 125–132. https://doi.org/10.1016/j.neucom.2019.11.008

[50] Chao Sun, Zhedong Zheng, Xiaohan Wang, Mingliang Xu, and Yi Yang. 2022. Self-supervised Point Cloud Representation Learning via Separating Mixed Shapes. *IEEE Transactions on Multimedia* (2022).

[51] Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*.

[52] Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. 2021. Dacs: Domain adaptation via cross-domain mixed sampling. In *WACV Workshop*.

[53] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. 2018. Learning to adapt structured output space for semantic segmentation. In *CVPR*.

[54] Yi-Hsuan Tsai, Kihyuk Sohn, Samuel Schulter, and Manmohan Chandraker. 2019. Domain adaptation for structured output via discriminative patch representations. In *ICCV*.

[55] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. 2021. Unsupervised semantic segmentation by contrasting object mask proposals. In *ICCV*.

[56] Midhun Vayyat, Jaswin Kasi, Anuraag Bhattacharya, Shuaib Ahmed, and Rahul Tallamraju. 2022. Cluda: Contrastive learning in unsupervised domain adaptation for semantic segmentation. *arXiv:2208.14227* (2022).

[57] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. 2019. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *CVPR*.

[58] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. 2019. Dada: Depth-aware domain adaptation in semantic segmentation. In *ICCV*.

[59] Haoran Wang, Tong Shen, Wei Zhang, Ling-Yu Duan, and Tao Mei. 2020. Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation. In *ECCV*.

[60] Mengzhu Wang, Wei Wang, Baopu Li, Xiang Zhang, Long Lan, Huibin Tan, Tianyi Liang, Wei Yu, and Zhigang Luo. 2021. Interbn: Channel fusion for adversarial

unsupervised domain adaptation. In *ACM Multimedia*.
[61] Qin Wang, Dengxin Dai, Lukas Hoyer, Luc Van Gool, and Olga Fink. 2021. Domain adaptive semantic segmentation with self-supervised depth estimation. In *ICCV*.
[62] Tingyu Wang, Zhedong Zheng, Yaoqi Sun, Tat-Seng Chua, Yi Yang, and Chenggang Yan. 2022. Multiple-environment Self-adaptive Network for Aerial-view Geo-localization. *arXiv preprint arXiv:2204.08381* (2022).
[63] Wenguan Wang, Tianfei Zhou, Fisher Yu, Jifeng Dai, Ender Konukoglu, and Luc Van Gool. 2021. Exploring Cross-Image Pixel Contrast for Semantic Segmentation. In *ICCV*.
[64] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. 2021. Dense contrastive learning for self-supervised visual pre-training. In *CVPR*.
[65] Zijian Wang, Yadan Luo, Zi Huang, and Mahsa Baktashmotlagh. 2020. Prototype-matching graph network for heterogeneous domain adaptation. In *ACM Multimedia*.
[66] Zhonghao Wang, Mo Yu, Yunchao Wei, Rogerio Feris, Jinjun Xiong, Wen-mei Hwu, Thomas S Huang, and Honghui Shi. 2020. Differential treatment for stuff and things: A simple unsupervised domain adaptation method for semantic segmentation. In *CVPR*.
[67] Xinyi Wu, Zhenyao Wu, Hao Guo, Lili Ju, and Song Wang. 2021. Dannet: A one-stage domain adaptation network for unsupervised nighttime semantic segmentation. In *CVPR*.
[68] Zuxuan Wu, Xintong Han, Yen-Liang Lin, Mustafa Gokhan Uzunbas, Tom Goldstein, Ser Nam Lim, and Larry S Davis. 2018. Dcan: Dual channel-wise alignment networks for unsupervised scene adaptation. In *ECCV*.
[69] Zuxuan Wu, Xin Wang, Joseph E Gonzalez, Tom Goldstein, and Larry S Davis. 2019. Ace: Adapting to changing environments for semantic segmentation. In *ICCV*.
[70] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. 2018. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*.
[71] Binhui Xie, Mingjia Li, and Shuang Li. 2021. Spcl: A new framework for domain adaptive semantic segmentation via semantic prototype-based contrastive learning. *arXiv:2111.12358* (2021).
[72] Binhui Xie, Shuang Li, Mingjia Li, Chi Harold Liu, Gao Huang, and Guoren Wang. 2022. SePiCo: Semantic-Guided Pixel Contrast for Domain Adaptive Semantic Segmentation. *arXiv:2204.08808* (2022).
[73] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. 2021. SegFormer: Simple and efficient design for semantic segmentation with transformers. *NeurIPS* (2021).
[74] Zizheng Yan, Xianggang Yu, Yipeng Qin, Yushuang Wu, Xiaoguang Han, and Shuguang Cui. 2021. Pixel-level intra-domain adaptation for semantic segmentation. In *ACM Multimedia*.
[75] Jinyu Yang, Weizhi An, Sheng Wang, Xinliang Zhu, Chaochao Yan, and Junzhou Huang. 2020. Label-driven reconstruction for domain adaptation in semantic segmentation. In *ECCV*.
[76] Shuyu Yang, Yinan Zhou, Yaxiong Wang, Yujiao Wu, Li Zhu, and Zhedong Zheng. 2023. Towards Unified Text-based Person Retrieval: A Large-scale Multi-Attribute and Language Search Benchmark. *arXiv:2306.02898* (2023).
[77] Yanchao Yang and Stefano Soatto. 2020. Fda: Fourier domain adaptation for semantic segmentation. In *CVPR*.
[78] Yalan Ye, Ziqi Liu, Yangwuyong Zhang, Jingjing Li, and Hengtao Shen. 2022. Alleviating Style Sensitivity then Adapting: Source-free Domain Adaptation for Medical Image Segmentation. In *ACM Multimedia*.
[79] Fuming You, Jingjing Li, Lei Zhu, Zhi Chen, and Zi Huang. 2021. Domain adaptive semantic segmentation without source data. In *ACM Multimedia*.
[80] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. 2017. mixup: Beyond empirical risk minimization. *arXiv:1710.09412* (2017).
[81] Pan Zhang, Bo Zhang, Ting Zhang, Dong Chen, Yong Wang, and Fang Wen. 2021. Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation. In *CVPR*.
[82] Richard Zhang, Phillip Isola, and Alexei A Efros. 2016. Colorful image colorization. In *ECCV*.
[83] Sicheng Zhao, Bo Li, Xiangyu Yue, Yang Gu, Pengfei Xu, Runbo Hu, Hua Chai, and Kurt Keutzer. 2019. Multi-source domain adaptation for semantic segmentation. *NeurIPS* (2019).
[84] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. 2021. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *CVPR*.
[85] Zhedong Zheng and Yi Yang. 2020. Unsupervised Scene Adaptation with Memory Regularization in vivo. In *IJCAI*.
[86] Zhedong Zheng and Yi Yang. 2021. Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation. *International Journal of Computer Vision* 129, 4 (2021), 1106–1120.
[87] Zhedong Zheng and Yi Yang. 2022. Adaptive Boosting for Domain Adaptation: Toward Robust Predictions in Scene Segmentation. *IEEE Transactions on Image Processing* 31 (2022), 5371–5382. https://doi.org/10.1109/TIP.2022.3195642
[88] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. 2020. Random erasing data augmentation. In *AAAI*.
[89] Qianyu Zhou, Zhengyang Feng, Qiqi Gu, Guangliang Cheng, Xuequan Lu, Jianping Shi, and Lizhuang Ma. 2022. Uncertainty-aware consistency regularization for cross-domain semantic segmentation. *Computer Vision and Image Understanding* (2022), 103448.
[90] Qianyu Zhou, Zhengyang Feng, Qiqi Gu, Jiangmiao Pang, Guangliang Cheng, Xuequan Lu, Jianping Shi, and Lizhuang Ma. 2022. Context-aware mixup for domain adaptive semantic segmentation. *IEEE Transactions on Circuits and Systems for Video Technology* (2022).
[91] Qianyu Zhou, Chuyun Zhuang, Ran Yi, Xuequan Lu, and Lizhuang Ma. 2022. Domain Adaptive Semantic Segmentation via Regional Contrastive Consistency Regularization. In *ICME*.
[92] Wei Zhou, Yukang Wang, Jiajia Chu, Jiehua Yang, Xiang Bai, and Yongchao Xu. 2020. Affinity space adaptation for semantic segmentation across domains. *IEEE Transactions on Image Processing* 30 (2020), 2549–2561.
[93] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*.
[94] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. 2018. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*.
[95] Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. 2019. Confidence regularized self-training. In *ICCV*.