Leverage Depth and Copy-Paste to Boost Contrastive Learning for Urban-Scene Segmentation

Liang Zeng
L.Zeng-1@student.tudelft.nl
Attila Lengyel
A.Lengyel@tudelft.nl
Nergis Tömen
N.Tomen@tudelft.nl
Jan van Gemert
J.C.vanGemert@tudelft.nl

Delft University of Technology
Delft, The Netherlands

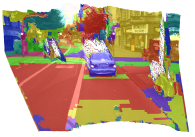
Abstract

In this work, we leverage estimated depth to boost self-supervised contrastive learning for segmentation on urban scenes, where videos are usually available for training self-supervised depth estimation. We argue that the semantics of a group of adjacent pixels in 3D space is self-contained and invariant across different contexts. We group these pixels given the estimated depth and use copy-paste to build cross-context correspondences. The model is optimized to retrieve the correspondences using contrastive learning. Our method allows directly learning superior visual representations from urban scenes. For unsupervised semantic segmentation, our method surpasses the previous state-of-the-art baseline by +7.14% in mIoU on Cityscapes and +6.65% on KITTI. For fine-tuning on Cityscapes and KITTI segmentation, our method is competitive with existing models that are pre-trained on the larger ImageNet or COCO with more GPUs. Our code is available at https://github.com/LeungTsang/MSc-Project.

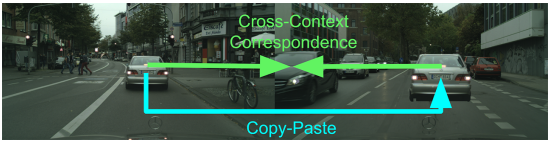
1 Introduction

Insufficiency of labeled data has become the main bottleneck for computer vision due to the cost of annotating [11, 35]. To this end, researchers have established the "self-supervised pre-training then fine-tuning" paradigm which utilizes a mass of unlabeled data. One of the most successful approaches is self-supervised contrastive learning [4, 5, 7, 21, 24]. The learned visual representations surpass the long dominant ImageNet[35] pre-training in many downstream tasks.

In this work, we explore how to use depth to aid self-supervised contrastive learning for urban-scene segmentation [11, 16]. We notice that how to perform contrastive learning on compelx urban scenes is still a vaccuum while self-supervised depth estimation on urban scenes is well-addressed [20, 50, 52], which can provide valuable clues. Recalling



(a) Adjacent Pixels in 3D



(b) Cross-Context Correspondence

Figure 1: We argue that adjacent pixels in 3D determine their semantics and the semantics are self-contained regardless of different contexts. (a)Pixels are grouped into regions as a prior. (b)We copy and paste the resulted regions to build cross-context correspondences. The representations of correspondences are pulled together by contrastive learning

the experience in real life, the semantic of a point can be mostly determined by scanning its adjacent region. We aim at approximating such observation in self-supervised learning. First of all, depth can reveal the true adjacency in 3D world. Given the estimated depth, we design a heuristic algorithm to group the pixels adjacent in 3D space. Next, to learn pixel representations only dependent on their adjacent regions, we copy and paste [18] these regions on different images to exclude irrelevant backgrounds and build cross-context correspondences. We optimize the network to retrieve the cross-context correspondences using contrastive learning.

The effectiveness of cross-context correspondences is two-fold. On one hand, as the group of adjacent pixels is sufficient to determine their semantics, the model should learn the invariance when these pixels co-occur in another context. On the other hand, a principle of self-supervised learning is avoiding shortcuts [17, 30] that can fulfill the defined tasks. Without the destruction by copy-paste, the model may overfit the constant irrelevant background pixels to find the correspondences. Our experiments show that copy-paste is a vital augmentation for learning object-specific representations in complex scenes.

(a) Cars in different contexts (b) w/o copy-paste (c)with copy-paste

Figure 2: Feature maps are visualized as RGB images by PCA reduction[38]. Copy-paste is vital for learn object-specific representations that invariant in different contexts

Our contributions are summarized as follows.

- We argue that the semantics of a group of adjacent pixels in 3D space is self-contained and invariant across different contexts. Based on such an argument, we propose a novel contrastive learning framework aided by depth estimation and copy-paste for urban-scene segmentation.
- For unsupervised semantic segmentation, our method outperforms the previous state-of-the-art baseline by a significant +7.14% in mIoU on Cityscapes [11] and +6.65% on KITTI [16].
- Our approach can effectively perform pre-training on urban-scene data for downstream tasks on urban scenes. When pre-training and fine-tuning on Cityscapes [11] and

KITTI [16], our method achieves competitive performance with single GPU to existing models that are pre-trained on ImageNet [35] or COCO [28] with 8 GPUs.

2 Related Works

2.1 Self-supervised Contrastive Learning

To alleviate the shortage of labeled data, self-supervised pre-training is gaining more attention in computer vision. Many self-supervised learning tasks have been explored [13, 19, 25, 30, 31, 39, 48]. Among them, contrastive learning [21] is one of the most promising methods.

The design of contrastive learning, especially the definition of positive and negative samples, depends on task and data at hand. Understanding the data and task then introducing certain priors in the learning process is important yet challenging. Pioneering works [3, 4, 5, 6, 7, 8, 24, 46] are mostly based on instance discrimination on ImageNet [35]. The same image under different transformation are positive, while other images are negative. For dense prediction [29], VADeR [32], DenseCL [40], PixPro [43] learn dense visual representation do instance discrimination in pixel-level. For detection on complex scenes [33], object-level instance discrimination is possible if we can localize objects beforehand [36, 42]. For segmentation on complex scenes, additional similarity can be defined among certain regions on images, such as classic hierarchical pixel grouping [49], auxiliary label [47], salient object estimation [15].

We adopt SwAV [4] to perform contrastive learning as it only need positive samples. We extend it to learn dense visual representation as our goal is segmentation. The positive samples can be defined at pixel-level or region-level. We compute precise pixel correspondences with respect to the geometric transformations similarly as in PixPro [43] and VADeR [32] but our geometric transformations involve copy-paste in addition. Region-level positive samples are sampled from pixel grouping results as recent works [15, 47, 49] but our pixel grouping is based on the novel 3D adjacency.

2.2 Copy-Paste Data Augmentation

Copy-Paste is a well-studied augmentation in supervised learning [12, 18, 45]. Copy-paste-like augmentation can also be involved in self-supervised contrastive learning frameworks but we must know what to copy-paste beforehand. DiLo [51] swaps salient objects from images while RegionCL [44] swaps random patches as augmentation. However, their assumption that the extracted objects or patches are semantically equivalent to the original images is only valid on simple object-centric datasets [35].

Our work utilizes depth to derive natural and abundant regions for copy-paste. The semantics of regions is less ambiguous, which helps the learning of dense visual representation on complex urban scenes.

2.3 Self-Supervised Depth Estimation

The recent-developed self-supervised depth estimation on videos has achieved impressive success [20, 50, 52] while Video clips are usually available in urban-scene datasets [11, 16]. Depth estimation may help supervised segmentation in different ways, including jointly training, data selection, data augmentation [26]

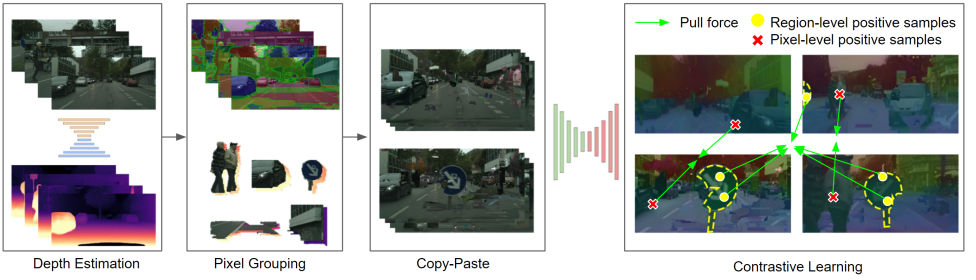


Figure 3: Our method consists of four steps. 1. Training a depth estimator on video clips by self-supervision. 2. Grouping pixels adjacent in 3D space given the depth. 3. Building cross-context correspondences by copy-paste. 4. Pulling together the representations of corresponding pixels and regions using contrastive learning.

In our work, depth serves as a prior for generating abundant and realistic augmented views for self-supervised learning. DepthMix [26] is adopted to enhance copy-paste augmentation.

3 Method

We describe the pipeline of our approach, which consists of four steps in Figure 3.

3.1 Self-Supervised Depth Estimation

To obtain depth, we take advantage of the recent success of self-supervised depth estimation on monocular videos[20, 50, 52]. In our framework, we adopt Monodepth2[20] in virtue of its simplicity and effectiveness.

3.2 Adjacent Pixels Grouping

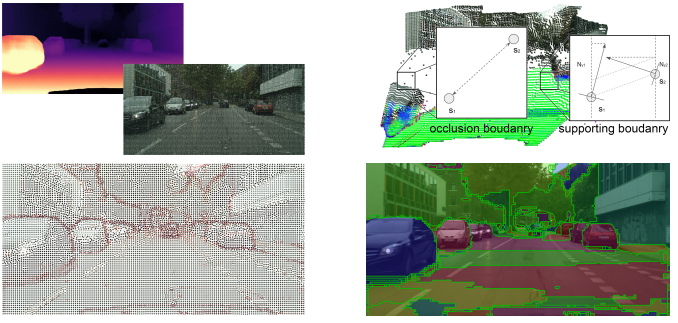
We crafted a heuristic algorithm to group pixels adjacent in 3D space based on the estimated depth and horizontal pattern of urban-scene images.

For each image, we compute the normal map from the depth D and take the component N_y perpendicular to the ground. Let $[u,v,1]$ denote the homogeneous coordinates of a pixel and $\mathcal{K}$ denote the camera intrinsics matrix. We can obtain 3D coordinates of the pixel $[x,y,z]$ under the camera coordinate by Eqn. 1 and normalize the coordinates by standard deviation.

$$[x,y,z]^\top = D(u,v)\,\mathcal{K}^{-1}\,[u,v,1]^\top \tag{1}$$

We use SLIC superpixels [1] to downsample the image and build the region adjacency graph from the superpixels. Each superpixel holds the average 3d coordinate $[x,y,z]$, upward normal N_y. With these attributes, we weight the edges by handcrafted formulas, which indicates how likely the edges lie on spatial boundaries.

We pinpoint two types of boundaries regarding occlusion and support relation [37] as Figure 4(b). We assume there are two adjacent superpixels $\mathbf{S}_1$ and $\mathbf{S}_2$, and $\mathbf{S}_1$ has lower height y_1. To detect occlusion boundaries, where the foreground occludes the background, euclidean distance is computed between $\mathbf{S}_1$ and $\mathbf{S}_2$. Noted that, the 3D scale of superpixels increases proportionally to depth so the distance should be normalized by their sum of depth



(c) Spatial boundary graph (d) Result of Adjacency Grouping

Figure 4: Decomposition of the proposed adjacent pixels grouping algorithm based on depth.

as Eqn.2. To detect supporting boundaries, where the objects stand on the ground, we assume that the lower $\mathbf{S}_1$ is the ground and $\mathbf{S}_2$ is an object. The pattern of urban-scene images implies that the ground surfaces should be level, which is measured by the normal in the upward direction n_{y1}. The ground should also be low so we have the square of y_1 when $y_1 < 0$. The objects on the ground should be upright, measured by the difference of n_{y1} and n_{y2}. The three terms are multiplied and the final equation is constructed as Eqn.3.

$$D_{ocln} = \frac{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}}{z_1 + z_2} \tag{2}$$

$$D_{sup} = \begin{cases} \max(n_{y1}, 0)(\max(n_{y1} - n_{y2}, 0))y_1^2 & y_1 < 0 \\ 0 & y_1 \geq 0 \end{cases} \tag{3}$$

The edge weight W between $\mathbf{S}_1$ and $\mathbf{S}_2$ is calculated by Eqn.4, the activation of a sigmoid function by the combination of the two distances and a bias term.

$$W = Sigmoid(w_{ocln} D_{ocln} + w_{sup} D_{sup} + b) \tag{4}$$

With the weighted graph as Figure4(c), the desired grouping of adjacent pixels becomes the nodes enclosed by possible edges so we turn the segmentation into a community detection problem. We use the classic InfoMap algorithm [34] to fulfill the task. Details are provided in supplementary materials.

3.3 Copy-Paste

With the 3D adjacency regions, we use copy-paste to build cross-context correspondences. We sample M images and extract all regions above a certain size. Those images also act as backgrounds. We perform copy-paste in two rounds. All regions will be pasted e times on every unmodified background in a round. In the first round, we set a small e to display more parts of the original images. In the second round, e is large to allow more regions present in new contexts. The generated $2M$ images form a training item. Details are provided in supplementary materials.

Along with copy-paste, random resize, horizontal flip, color jitter, and gaussian blur are applied. We adopt DepthMix [26] as depth is handy. The pixels where the depth is larger than the background will not be pasted. The position to paste is within a threshold h_t from

the original height of the regions since height is an important prior [10]. All geometric transformations are recorded as transform matrices so that the correspondence can be traced back during training.

3.4 Contrastive Learning and Positive Samples Sampling

3.4.1 SwAV contrastive learning framework

We adopt the clustering-based contrastive learning framework introduced by SwAV [3]. Intuitively, treating all different samples as negative samples like SimCLR [4] and MoCo [24] and explicitly maximizing their representation distance could be problematic in urban scenes. Samples are frequently sampled from the same major classes such as roads, buildings, and vehicles. SwAV is more robust because it only needs positive samples.

We recap SwAV with slight modification in the swap prediction loss since the number of positive samples is inconstant in each group. With a set of representations $\mathbf{Z} = [\mathbf{z}_0, \mathbf{z}_1, ..., \mathbf{z}_N]$, we compute their assignments $\mathbf{P} = [\mathbf{p}_0, \mathbf{p}_1, ..., \mathbf{p}_N]$ to K prototypes $\mathbf{C} = [\mathbf{c}_0, \mathbf{c}_1, ..., \mathbf{c}_K]$ by Eqn.5 with a temperature τ.

$$\mathbf{p}_n^{(k)} = \frac{\exp\left(\frac{1}{\tau}\mathbf{z}_n^\top \mathbf{c}_k\right)}{\sum_{k'} \exp\left(\frac{1}{\tau}\mathbf{z}_n^\top \mathbf{c}_{k'}\right)} \tag{5}$$

Directly minimizing the cross-entropy for the assignments of positive samples will lead to a trivial solution. Codes $\mathbf{Q}$ are computed by Sinkhorn-Knopp algorithm[12] which is the balanced assignments with the smallest distance to $\mathbf{P}$. We average the codes within each group of positive samples by Eqn.6.

$$\overline{\mathbf{q}}_n^{(k)} = \frac{1}{|\mathbf{G}^n|} \sum_{i \sim \mathbf{G}^n} \mathbf{q}_i^{(k)} \quad n \sim \mathbf{G}^n \tag{6}$$

The loss for $\mathbf{Z}$ is given by the mean of cross-entropy of all assignments and their corresponding codes as Eqn. 7

$$\mathcal{L}_\mathbf{Z} = -\frac{1}{N} \sum_{n=1}^{N} \sum_{k=1}^{K} \overline{\mathbf{q}}_n^{(k)} \log \mathbf{p}_n^{(k)} \tag{7}$$

3.4.2 Pixel-level positive samples

As a simple extension from image-level instance discrimination to pixel-level, the same pixels under different transformations are positive samples. We randomly sample a certain number of initial coordinates on feature maps. For each initial coordinate, we inspect the recorded geometric transformations to localize all its positive counterparts in a batch. We then perform bilinear interpolation to obtain the features at those coordinates.

3.4.3 Region-level positive samples

For segmentation tasks, we may assume the pixels inside the same region share similarity. We randomly sample a certain number of coordinates on feature maps and perform bilinear interpolation to obtain the features. Coordinates from identical source regions are positive samples.

3.4.4 Loss weight

Let $\mathcal{L}_{pixel}$ and $\mathcal{L}_{region}$ denote the loss computed among the pixel-level positive samples and region-level positive samples. We balance the two types of loss with a weight λ as Eqn. 8.

$$\mathcal{L} = \lambda \mathcal{L}_{pixel} + (1-\lambda)\mathcal{L}_{region} \tag{8}$$

3.5 Clustering

For unsupervised semantic segmentation, we need to cluster the learned representation. A fine-grained clustering is already done by SwAV[3] and we can further cluster the prototypes to the target number of classes. Empirically, we choose agglomerative clustering with cosine distance measurement and average linkage criterion.Assuming that the clustering result over the prototypes $\mathbf{C}$ is $\hat{\mathbf{C}} = [\hat{c_0}, \hat{c_1}, ..., \hat{c_K}]$, the pseudo-class of feature $\mathbf{z}_n$ is given by Eqn. 9

$$Class_n = \hat{\mathbf{C}}(\arg\max_k(\mathbf{z}_n^\top \mathbf{c}_k)) \tag{9}$$

4 Experiments

4.1 Implementation details

Datasets We performed our experiments on Cityscapes[11] and KITTI[16]. Cityscapes has 2975 30-frame stereo videos with one frame annotated in *train* set and 500 images in *val* set. KITTI has 71 stereo videos from the categories city, road, residential, and campus. Its *train* set has 200 annotated frames. We used the left images, resulting in 89250 images for Cityscapes and 42382 for KITTI. We resized Cityscapes images to 384×768 and KITTI images to 384×1280. Baseline models were mainly trained on ImageNet[35] or COCO[28].

Hyperparameters For depth estimation, we used the default setting in Monodepth2[20]. For 3D adjacency grouping, the number of superpixels was 10000 and w_{ocln}, w_{sup}, b as defined in Eqn. 4 are 48.0, 200.0, -4.0, respectively. We used the default parameters of infomap[34]. We mixed all images mutually in a batch by copy-paste. The expectation e is set to 1 in the first round and 2 in the second round.

Network architecture We used semantic FPN[27] with ResNet-50[22] as the feature extractor. The final classifier was replaced by a 1-layer MLP projection head with 128 output channels. We held 1000 prototypes and set temperature τ to 0.1 in SwAV[3].

Training We trained the model for 40 epochs on a single 16GB V100. The batch size was 8, bringing 16 images to input. To speed up training we generated synthesized data equivalent to 2 epochs beforehand and train on them repeatedly. During training, resized crop, horizontal flip, and color augmentation are applied to full images. The model was updated by the SGD optimizer with momentum 0.9 and weight decay $1e^{-4}$. The initial learning rate was warmed up to $1e^{-1}$ in the first 2 epochs and then decayed to $1e^{-5}$ via cosine annealing. We sampled around 288k features for contrasting per iteration and the budget was allocated to pixel-level and region-level equally if $1 > \lambda > 0$.

4.2 Unsupervised Semantic Segmentation

Evaluation setting Seldom works try to address the challenging unsupervised urban-scene semantic segmentation. We compare our results to the recently introduced PiCIE[9], which achieved the previous state-of-art in this scenario. Noted that, in the original paper, PiCIE is trained with resnet-18 initialized by ImageNet pre-trained model and their results are evaluated on raw 27 classes instead of the conventional 19 evaluated classes on Cityscapes[11]. We retrained PiCIE with the equivalent setting to ours, e.g., architecture, and batch size. We report the performance given the optimal match between the clustering results and ground truth in Table 1.

Unsupervised semantic segmentation With ImageNet initialization[35], Our method surpasses PiCIE[9] by +7.14% and 6.65% in mIoU on Cityscapes[11] and KITTI[16] respectively but fall behind PiCIE in accuracy. With random initialization, our method degrades little and outperforms PiCIE by a huge margin. We also observe training PiCIE has to preserve the ImageNet initialized weights carefully and avoid a long training. Meanwhile, due to our 3D adjacency pixel grouping, our method can better capture the outline of objects and detect finer objects as Figure 5. PiCIE can only predicts rough shapes for big objects and stuff, leading to a good accuracy but poor mIoU. Our method performs consistently when transferring to the other dataset, but the performance of PiCIE drops noticeably.

Method	Init.	Training Data	CS-Sem.		KT-Sem.	
			Acc	mIoU	Acc	mIoU
PiCIE[9]	scratch	CS[11]	33.56	8.33	32.20	6.52
Ours($\lambda = 0.5$)	scratch	CS[11]	65.42	20.49	**68.37**	21.03
PiCIE[9]	scratch	KT[16]	30.28	6.81	30.62	7.54
Ours($\lambda = 0.5$)	scratch	KT[16]	49.18	17.20	49.58	18.22
PiCIE[9]	IN[35]	CS[11]	**68.50**	16.24	56.74	13.54
Ours($\lambda = 0.5$)	IN[35]	CS[11]	66.70	**23.38**	68.25	**22.50**
PiCIE[9]	IN[35]	KT[11]	53.24	12.55	61.74	12.92
Ours($\lambda = 0.5$)	IN[35]	KT[16]	56.96	18.85	59.11	19.57

Table 1: Unsupervised semantic segmentation performance on Cityscapes[11] *val* set and KITTI[16] *train* set. We retrained PiCIE with equivalent setting to ours.

Figure 5: Qualitative comparison for unsupervised semantic segmentation on Cityscapes [11]

4.3 Semi-Supervised Segmentation

Evaluation setting We fine-tuned pre-trained models on Cityscapes [11] and KITTI [16] for semantic and instance segmentation using semantic FPN [27] and Mask R-CNN [23]

Method	Training Data	CS-Sem.	CS-Inst.		KT-Sem.	KT-Inst.	
		mIoU	AP	AP_{50}	mIoU	AP	AP_{50}
scratch	-	65.11	24.30	46.98	32.99	8.15	15.79
supervised	IN[35]	70.54	27.34	50.59	40.09	12.38	23.42
SwAV[3]	IN[35]	71.07	28.08	52.25	40.52	**13.78**	**27.90**
DenseCL[40]	IN[35]	72.09	28.97	51.93	40.88	12.63	22.74
PixPro[43]	IN[35]	72.66	29.04	52.59	40.50	13.04	24.95
ORL[42]	CC[28]	72.32	**29.94**	52.55	41.88	12.02	23.48
CAST[36]	CC[28]	69.92	27.33	51.31	38.78	10.67	20.13
SwAV[3]	CS[11]	61.69	23.62	46.21	36.10	9.22	18.01
PixPro[43]	CS[11]	61.64	23.78	46.45	36.99	9.61	18.73
SwAV[3]	KT[16]	60.74	23.51	46.08	36.90	9.42	18.11
PixPro[43]	KT[16]	61.25	23.23	46.23	37.28	9.45	18.57
Ours($\lambda = 1$)	CS[11]	**73.55**	**29.94**	**52.88**	**42.70**	12.58	24.98
Ours($\lambda = 0.5$)	CS[11]	73.03	29.11	51.87	42.32	12.22	23.16
Ours($\lambda = 1$)	KT[16]	71.62	28.77	52.71	41.17	11.74	20.57
Ours($\lambda = 0.5$)	KT[16]	71.45	27.86	51.16	41.03	11.36	20.49

Table 2: Segmentation performance over Cityscapes *val* set and 5-fold validation of KITTI *train* set. SwAV and PixPro on Cityscapes and KITTI are trained with limited device as ours. For SwAV, we spliting the images into 18 128×128 patches since it works on simpler images.

implemented by detectron2 [41]. The batch size was 16 and the fine-tuning on Cityscapes and KITTI took 30k and 6k iterations, respectively. We set a smaller initial learning rate $1e^{-2}$ for all pre-trained parameters. For other randomly initialized parameters, the learning rate was tuned from $3e^{-2}$ to $1e^{-1}$. We smoothly fused these parameters by a cosine decay scheduler with a final learning rate of $1e^{-5}$. This strategy worked well for all pre-trained models. We used random scale, random crop, and color augmentation. We repeated the training for 3 times and report the average performance.

Semantic segmentation We demonstrate the semi-supervised semantic segmentation performance in Table 2 on Cityscapes and KITTI. Our method pre-trained on Cityscapes with $\lambda = 1$ outperforms other existing pre-trained models for semantic segmentation by a noticeable margin on both datasets. It exceeds the SwAV[3] pre-trained model by $+3.01\%$ on Cityscapes and $+2.18\%$ on KITTI in mIoU. The KITTI pre-trained model also achieves a comparable result on both datasets with less training data. Overall, our method demonstrates superior performance and generalization across different urban scene data.

Instance segmentation We show the semi-supervised instance segmentation performance in Table 2 on Cityscapes and KITTI. Our Cityscapes pre-trained model with $\lambda = 1$ still outperforms other baseline on Cityscape but ORL[42] gains very close result. It still exceeds SwAV[3] by $+1.86\%$ in AP and $+0.63\%$ in AP_{50} on Cityscapes. However, the performance is only around the average level on KITTI. The AP and AP_{50} is -1.20% and -2.92% below SwAV[3] respectively. The lower performance is expected because some layers of semantic FPN[27] are discarded to adapt Mask-RCNN[23] architecture for instance segmentation.

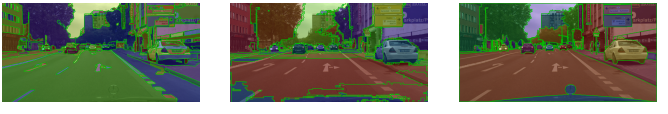
Training baseline methods on urban scenes The gaining comes from not only the pre-training on similar data to downstream tasks but also the effectiveness of the design. With equivalent settings to ours(e.g., on one 16GB GPU), training SwAV[3] and PixPro[43] on Cityscpaes[11] or KITTI[16] yields inferior results, indicate that these methods are less adaptive to urban scenes or need more computation resources. Training details and discussion are provided in supplementary materials.

4.4 Ablation Study

We performed ablation studies to validate our design choices. In ablation experiments, the models were trained on the 2975 images from the *train* set of the Cityscapes[11] for 100 epochs. The semantic segmentation performance is reported after fine-tuning on $1/16$ of the *train* set for 6k iterations.

4.4.1 Region proposal

To study if our 3D adjacency region is significant, we used different region proposals for comparison, including owt-ucm[2] used by Zhang et al.[49], and ground truth as Figure 8. We show the fine-tuning performance in Table 4. Our 3D adjacency region proposal produces the closest performance to ground truth.



(a)owt-ucm[2] (b)Ours (c)GT

Figure 6: Examples of different region proposals.

Weight	Fine-tuning			Clustering		
	$\lambda=0$	$\lambda=0.5$	$\lambda=1$	$\lambda=0$	$\lambda=0.5$	$\lambda=1$
owt-ucm[2]	44.83	47.01	47.24	13.64	15.70	12.11
Ours	46.91	48.87	48.55	15.94	20.71	13.94
GT	49.92	52.26	48.82	24.05	27.35	14.59

Table 3: Effects of different object proposals, objective8. Measurement is based on mIoU by fine-tuning on a subset of Cityscapes[11] for semantic segmentation.

4.4.2 Copy-Paste

To further investigate the effectiveness of copy-paste, we altered the number of images M for copy-paste. We report the fine-tuning and clustering performance in Table 4. Mixing more images to build richer cross-context correspondence is beneficial in all cases, especially when region-level positive samples are sampled($\lambda < 1$).

4.4.3 Loss Weight

We can inspect the effects of loss weight8 in Table 4. For high-quality proposals like ground truth, combining the pixel-level and region-level positive samples remarkably increases the performance for both fine-tuning and unsupervised clustering. As to our 3D adjacency region

Weight	M / Proposal	0	2	4	8	0	2	4	8
		Fine-tuning				Clustering			
$\lambda = 0$	Ours	35.18	42.34	46.75	46.91	5.93	13.92	13.65	15.94
	GT	37.70	46.74	48.64	49.92	9.39	21.31	23.51	24.05
$\lambda = 0.5$	Ours	34.35	45.92	48.23	48.87	3.97	14.03	20.32	20.71
	GT	38.01	49.26	50.06	52.26	8.20	24.26	24.82	27.35
$\lambda = 1$	Ours	39.93	48.24	48.09	48.55	4.89	10.59	13.65	13.94
	GT		48.20	49.25	48.81		10.36	14.08	14.59

Table 4: Effects of different objectives8 and number of sample images M for copy-paste. 0 means disabling copy-paste. Measurement is based on mIoU by fine-tuning on subset of Cityscapes[11] for semantic segmentation.

proposal, the noisy regions do not strictly contain a single class. Including region-level positive samples takes little effect on fine-tuning performance as Table 4 or even slightly degrades the performance in our main result2. For better clustering, their combination is still necessary.

5 Discussion and Limitations

In this work, we argue that adjacent pixels in 3D space determine their semantics and this semantic is invariant across different contexts. Fortunately, self-supervised depth estimation is possible on most urban scene datasets so we utilize estimated depth to group adjacent pixels and copy-paste to build cross-context correspondence in contrastive learning. Experiments on Cityscapes[11] and KITTI[16] demonstrate the effectiveness of our method. Trained on one GPU, it surpasses previous state-of-the-art unsupervised semantic segmentation method and achieves similar fine-tuning performance to existing models that are pre-trained on ImageNet[35] or COCO[28] with 8 GPUs. Our method is potentially effective in other scenarios where depth is available.

Unfortunately, our experiments were performed on a smaller scale, limited by hardware resources. Besides, the pipeline consists of multiple components and the final performance is subject to each part. Especially, the pixel grouping algorithm is handcrafted and sensitive to hyperparameters. Future work may replace it with a data-driven method.

References

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurélien Lucchi, Pascal Fua, and Sabine Süsstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(11):2274–2282, 2012. doi: 10.1109/TPAMI.2012.120. URL https://doi.org/10.1109/TPAMI.2012.120.

[2] Pablo Arbelaez, Michael Maire, Charless C. Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(5):898–916, 2011. doi: 10.1109/TPAMI.2010.161. URL https://doi.org/10.1109/TPAMI.2010.161.

[3] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski,

and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/70feb62b69f16e0238f741fab228fec2-Abstract.html.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 2020. URL http://proceedings.mlr.press/v119/chen20j.html.

[5] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E. Hinton. Big self-supervised models are strong semi-supervised learners. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/fcbc95ccdd551da181207c0c1400c655-Abstract.html.

[6] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 15750–15758. Computer Vision Foundation / IEEE, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Chen_Exploring_Simple_Siamese_Representation_Learning_CVPR_2021_paper.html.

[7] Xinlei Chen, Haoqi Fan, Ross B. Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *CoRR*, abs/2003.04297, 2020. URL https://arxiv.org/abs/2003.04297.

[8] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 9620–9629. IEEE, 2021. doi: 10.1109/ICCV48922.2021.00950. URL https://doi.org/10.1109/ICCV48922.2021.00950.

[9] Jang Hyun Cho, Utkarsh Mall, Kavita Bala, and Bharath Hariharan. Picie: Unsupervised semantic segmentation using invariance and equivariance in clustering. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 16794–16804. Computer Vision Foundation / IEEE, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Cho_PiCIE_Unsupervised_Semantic_Segmentation_Using_Invariance_and_Equivariance_in_Clustering_CVPR_2021_paper.html.

[10] Sungha Choi, Joanne Taery Kim, and Jaegul Choo. Cars can't fly up in the sky: Improving urban-scene segmentation via height-driven attention networks.

In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9370–9380. Computer Vision Foundation / IEEE, 2020. doi: 10.1109/CVPR42600.2020.00939. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Choi_Cars_Cant_Fly_Up_in_the_Sky_Improving_Urban-Scene_Segmentation_CVPR_2020_paper.html.

[11] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 3213–3223. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.350. URL https://doi.org/10.1109/CVPR.2016.350.

[12] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 2292–2300, 2013. URL https://proceedings.neurips.cc/paper/2013/hash/af21d0c97db2e27e13572cbf59eb343d-Abstract.html.

[13] Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 1422–1430. IEEE Computer Society, 2015. doi: 10.1109/ICCV.2015.167. URL https://doi.org/10.1109/ICCV.2015.167.

[14] Haoshu Fang, Jianhua Sun, Runzhong Wang, Minghao Gou, Yonglu Li, and Cewu Lu. Instaboost: Boosting instance segmentation via probability map guided copy-pasting. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 682–691. IEEE, 2019. doi: 10.1109/ICCV.2019.00077. URL https://doi.org/10.1109/ICCV.2019.00077.

[15] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Unsupervised semantic segmentation by contrasting object mask proposals. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 10032–10042. IEEE, 2021. doi: 10.1109/ICCV48922.2021.00990. URL https://doi.org/10.1109/ICCV48922.2021.00990.

[16] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013.

[17] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard S. Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nat. Mach. Intell.*, 2(11):665–673, 2020. doi: 10.1038/s42256-020-00257-z. URL https://doi.org/10.1038/s42256-020-00257-z.

[18] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D. Cubuk, Quoc V. Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 2918–2928. Computer Vision Foundation / IEEE, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Ghiasi_Simple_Copy-Paste_Is_a_Strong_Data_Augmentation_Method_for_Instance_CVPR_2021_paper.html.

[19] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=S1v4N2l0-.

[20] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J. Brostow. Digging into self-supervised monocular depth estimation. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 3827–3837. IEEE, 2019. doi: 10.1109/ICCV.2019.00393. URL https://doi.org/10.1109/ICCV.2019.00393.

[21] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), 17-22 June 2006, New York, NY, USA*, pages 1735–1742. IEEE Computer Society, 2006. doi: 10.1109/CVPR.2006.100. URL https://doi.org/10.1109/CVPR.2006.100.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[23] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2980–2988. IEEE Computer Society, 2017. doi: 10.1109/ICCV.2017.322. URL https://doi.org/10.1109/ICCV.2017.322.

[24] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9726–9735. Computer Vision Foundation / IEEE, 2020. doi: 10.1109/CVPR42600.2020.00975. URL https://doi.org/10.1109/CVPR42600.2020.00975.

[25] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B. Girshick. Masked autoencoders are scalable vision learners. *CoRR*, abs/2111.06377, 2021. URL https://arxiv.org/abs/2111.06377.

[26] Lukas Hoyer, Dengxin Dai, Yuhua Chen, Adrian Köring, Suman Saha, and Luc Van Gool. Three ways to improve semantic segmentation with self-supervised depth estimation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*,

pages 11130–11140. Computer Vision Foundation / IEEE, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Hoyer_Three_Ways_To_Improve_Semantic_Segmentation_With_Self-Supervised_Depth_Estimation_CVPR_2021_paper.html.

[27] Alexander Kirillov, Ross B. Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 6399–6408. Computer Vision Foundation / IEEE, 2019. doi: 10.1109/CVPR.2019.00656. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Kirillov_Panoptic_Feature_Pyramid_Networks_CVPR_2019_paper.html.

[28] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, volume 8693 of *Lecture Notes in Computer Science*, pages 740–755. Springer, 2014. doi: 10.1007/978-3-319-10602-1_48. URL https://doi.org/10.1007/978-3-319-10602-1_48.

[29] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2015. doi: 10.1109/CVPR.2015.7298965.

[30] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI*, volume 9910 of *Lecture Notes in Computer Science*, pages 69–84. Springer, 2016. doi: 10.1007/978-3-319-46466-4_5. URL https://doi.org/10.1007/978-3-319-46466-4_5.

[31] Deepak Pathak, Ross B. Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 6024–6033. IEEE Computer Society, 2017. doi: 10.1109/CVPR.2017.638. URL https://doi.org/10.1109/CVPR.2017.638.

[32] Pedro O. Pinheiro, Amjad Almahairi, Ryan Y. Benmalek, Florian Golemo, and Aaron C. Courville. Unsupervised learning of dense visual representations. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/3000311ca56a1cb93397bc676c0b7fff-Abstract.html.

[33] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information*

Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada, pages 91–99, 2015. URL https://proceedings.neurips.cc/paper/2015/hash/14bfa6bb14875e45bba028a21ed38046-Abstract.html.

[34] M. Rosvall, D. Axelsson, and C. T. Bergstrom. The map equation. *The European Physical Journal Special Topics*, 178(1):13–23, nov 2009. doi: 10.1140/epjst/e2010-01179-1. URL https://doi.org/10.1140%2Fepjst%2Fe2010-01179-1.

[35] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

[36] Ramprasaath R. Selvaraju, Karan Desai, Justin Johnson, and Nikhil Naik. Casting your model: Learning to localize improves self-supervised representations. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 11058–11067. Computer Vision Foundation / IEEE, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Selvaraju_CASTing_Your_Model_Learning_To_Localize_Improves_Self-Supervised_Representations_CVPR_2021_paper.html.

[37] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In Andrew W. Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *Computer Vision - ECCV 2012 - 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V*, volume 7576 of *Lecture Notes in Computer Science*, pages 746–760. Springer, 2012. doi: 10.1007/978-3-642-33715-4_54. URL https://doi.org/10.1007/978-3-642-33715-4_54.

[38] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIII*, volume 11217 of *Lecture Notes in Computer Science*, pages 402–419. Springer, 2018. doi: 10.1007/978-3-030-01261-8_24. URL https://doi.org/10.1007/978-3-030-01261-8_24.

[39] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 2794–2802. IEEE Computer Society, 2015. doi: 10.1109/ICCV.2015.320. URL https://doi.org/10.1109/ICCV.2015.320.

[40] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 3024–3033. Computer Vision Foundation / IEEE, 2021. URL

https://openaccess.thecvf.com/content/CVPR2021/html/Wang_Dense_Contrastive_Learning_for_Self-Supervised_Visual_Pre-Training_CVPR_2021_paper.html.

[41] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

[42] Jiahao Xie, Xiaohang Zhan, Ziwei Liu, Yew Soon Ong, and Chen Change Loy. Unsupervised object-level representation learning from scene images. In *NeurIPS*, 2021.

[43] Zhenda Xie, Yutong Lin, Zheng Zhang, Yue Cao, Stephen Lin, and Han Hu. Propagate yourself: Exploring pixel-level consistency for unsupervised visual representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 16684–16693. Computer Vision Foundation / IEEE, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Xie_Propagate_Yourself_Exploring_Pixel-Level_Consistency_for_Unsupervised_Visual_Representation_Learning_CVPR_2021_paper.html.

[44] Yufei Xu, Qiming Zhang, Jing Zhang, and Dacheng Tao. Regioncl: Can simple region swapping contribute to contrastive learning? *CoRR*, abs/2111.12309, 2021. URL https://arxiv.org/abs/2111.12309.

[45] Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Seong Joon Oh, Youngjoon Yoo, and Junsuk Choe. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 6022–6031. IEEE, 2019. doi: 10.1109/ICCV.2019.00612. URL https://doi.org/10.1109/ICCV.2019.00612.

[46] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 12310–12320. PMLR, 2021. URL http://proceedings.mlr.press/v139/zbontar21a.html.

[47] Feihu Zhang, Philip Torr, Rene Ranftl, and Stephan Richter. Looking beyond single images for contrastive semantic segmentation learning. *Advances in Neural Information Processing Systems*, 34, 2021.

[48] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful image colorization. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III*, volume 9907 of *Lecture Notes in Computer Science*, pages 649–666. Springer, 2016. doi: 10.1007/978-3-319-46487-9_40. URL https://doi.org/10.1007/978-3-319-46487-9_40.

[49] Xiao Zhang and Michael Maire. Self-supervised visual representation learning from hierarchical grouping. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia

Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/c1502ae5a4d514baec129f72948c266e-Abstract.html.

[50] Chaoqiang Zhao, Qiyu Sun, Chongzhen Zhang, Yang Tang, and Feng Qian. Monocular depth estimation based on deep learning: An overview. *CoRR*, abs/2003.06620, 2020. URL https://arxiv.org/abs/2003.06620.

[51] Nanxuan Zhao, Zhirong Wu, Rynson W. H. Lau, and Stephen Lin. Distilling localization for self-supervised representation learning. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 10990–10998. AAAI Press, 2021. URL https://ojs.aaai.org/index.php/AAAI/article/view/17312.

[52] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G. Lowe. Unsupervised learning of depth and ego-motion from video. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 6612–6619. IEEE Computer Society, 2017. doi: 10.1109/CVPR.2017.700. URL https://doi.org/10.1109/CVPR.2017.700.

6 Supplementary Materials

6.1 Evaluation Setting

6.1.1 Unsupervised semantic segmentation

Seldom works try to address the challenging unsupervised urban-scene semantic segmentation. We compare our results to the recently introduced PiCIE[9], which achieved the previous state-of-art in this scenario. Noted that, in the original paper, PiCIE is trained with resnet-18 initialized by ImageNet pre-trained model and their results are evaluated on raw 27 classes instead of the conventional 19 evaluated classes on Cityscapes[11]. We retrained PiCIE with the equivalent setting to ours, e.g., architecture, and batch size. We report the performance given the optimal match between the clustering results and ground truth in Table 1.

6.1.2 Semi-supervised segmentation

6.2 Iterative InfoMap for community detection

Directly executing InfoMap[34] one time cannot detect multi-scale communities. Therefore, we apply InfoMap iteratively. In each iteration, the target number of communities is set to half of the current nodes in the current graph. The communities with all outward edges larger than T are fixed and other communities are treated as new nodes for the next iteration until only one community is left. Pseudo code is shown in Algorithm1. The result is the groups of superpixels according to their final communities.

Algorithm 1 Community Detection

Input: Graph G, Threshold T, Function InfoMap(graph, expected community num)
Output: Community for node in G

$INITIAL\_NODES \leftarrow nodes \in G$
$COMMUNITY \leftarrow [0, 0, ..., 0]$
$NEXT\_ID = 0$
for all $n \in G$ **do**
  $n.initial\_nodes \leftarrow n$
end for
$N \leftarrow number\ of\ nodes \in G$
while $N > 1$ **do**
  $community = InfoMap(G,\ N//2)$
  for all $n \in G$ **do**
    $n.c \leftarrow community[n]$
  end for
  $newG \leftarrow Empty\ Graph$
  for all $c \in unique(community)$ **do**
    $add\ node\ c\ in\ newG$
    $c.initial\_nodes \leftarrow all\ n.initial\_nodes\ where\ n \in G\ and\ n.c = c$
  end for
  for all $a \in newG\ and\ b \in newG$ **do**
    $weight \leftarrow avg(edge_{nm}.weight)\ where\ edge_{nm} \in G\ and\ n.c = a\ and\ m.c = b$
    if $weight > 0$ **then**
      $edge_{ab}.weight \leftarrow weight$
      $add\ edge_{ab}\ in\ newG$
    end if
  end for
  for all $a \in newG$ **do**
    if $all\ edge_{ab} > T\ where\ b \in newG\ and\ edge_{ab} \in newG$ **then**
      for all $n \in a.initial\_nodes$ **do**
        $COMMUNITY[n] \leftarrow NEXT\_ID$
      end for
      $NEXT\_ID++$
      $remove\ a\ from\ newG$
    end if
  end for
  $G \leftarrow newG$
  $N \leftarrow number\ of\ nodes\ in\ G$
end while
return $COMMUNITY$

6.3 Quantitative Evaluation for Pixel Grouping

To quantify the grouping quality, we split the instance label in Cityscapes *train* into only connected regions as the ground truth of our idea. We calculate the mIoU in two ways in comparison with owt-ucm[2] in Table5. When using the GT mask as the query to fetch the the closest mask in region proposal, our pixel grouping method produces higher mIoU. When using the optimal bilateral match of the grouping result and GT, our method falls behind because of generating many tiny pieces. We will show that our pixel grouping method produces better performance in our framework in experiments.

Method	mIoU(GT query)	mIoU(bilateral match)
Ours	10.80	3.86
otw-ucm(0.1)[2]	4.90	4.89
otw-ucm(0.05)[2]	8.52	8.08
otw-ucm(0.01)[2]	7.94	5.30

Table 5: Pixel grouping mIoU by GT query or bilateral match. For owt-ucm[2], boundaries under the strength threshold will be filtered.

6.4 Copy-Paste

Algorithm 2 Copy-Paste

Input: Num of sample image M, Size threshold T, Expectation $[e_1, e_2]$
Output: Images for training *IMAGES_train*

```
IMAGES_train ← []
IMAGES ← sampling M images from Dataset along with depth and region proposals
R ← all regions ∈ IMAGES above size T
N ← number of regions in R
for iter in range(2) do
    for all img ∈ IMAGES do
        img = deepcopy(img)
        R_cp ← sampling e[iter] * N/M regions from R
        for all r ∈ R_cp do
            r ← Augmentation(r)
            copy − paste r on img with DepthMix
        end for
        IMAGES_train.append(img)
    end for
end for
return IMAGES_train
```

6.5 Training SwAV on Cityscapes

We designed experiments to investigate the best practice for general SwAV[3] on urban scene data in our setting. With Cityscapes[11], we use the raw 384×768 image, 128×128 patch images, or object-centric images generated from ground truth label to train SwAV. To the maximum, the batch size is 16 for raw images and 288 for other cases. From table6, We

see that object-centric prior is important for SwAV. Splitting the images into patches is an approximation without labels. Using the raw images of complex scenes leads to the worst performance. Overall, the gaining from the pre-training using SwAV is still trivial with limited GPU resources.

Method	Training data	mIoU
scratch	-	37.85
SwAV[3]	CS-raw	31.92
SwAV[3]	CS-patch	37.88
SwAV[3]	CS-object	39.63
Ours($\lambda = 0$)	CS-raw	46.91
Ours($\lambda = 0.5$)	CS-raw	48.87
Ours($\lambda = 1$)	CS-raw	48.55
supervised	CS-object	39.28
supervised	ImageNet[35]	48.33

Table 6: Effect of using different pre-processing of Cityscapes[11] for SwAV[3]. Measurement is based on mIoU by fine-tuning on $1/16$ subset of Cityscapes for semantic segmentation.

6.6 More Visualization

We provide more visualization to demonstrate the effectiveness of copy-paste and the combination of pixel-level and region-level positive samples intuitively.

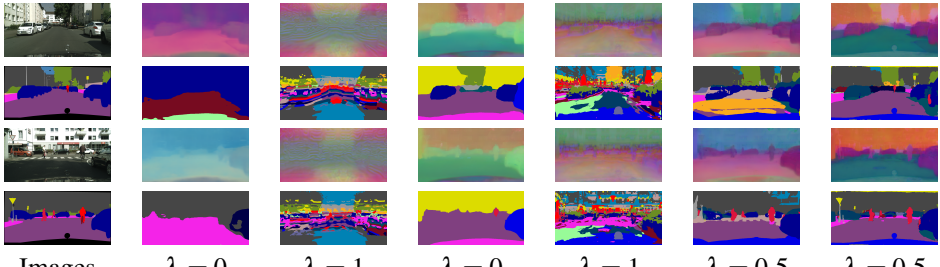


Figure 7: We visualize the feature maps as RGB images by PCA reduction[33] and the feature clustering result by different settings. R means sampling region-level positive samples and P means sampling pixel-level positive samples. CP means enable copy-paste and the image sample number for copy-paste is 8. We use our 3D adjacency region proposal except the last column which uses the ground truth proposal. Copy-paste and the combination of pixel-level and region-level positive samples encourage richer and object-specific features.

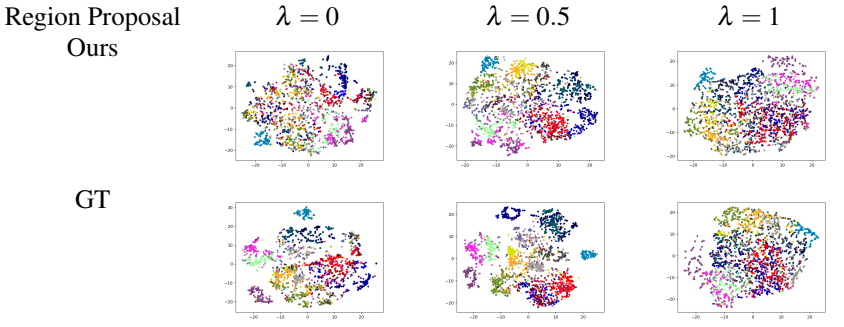


Figure 8: We visualize the distribution of 3000 samples feature by t-sne. Samples are colored according to their true classes. Copy-paste is enabled. We can see that the combination of pixel-level and region-level positive samples gives better features clustering results. There is a gap between our 3D adjacency region proposal and GT. Especially, region-level positive samples are more sensitive to proposal quality