Semantic Information in Contrastive Learning

Shengjiang Quan[1]*, Masahiro Hirano[2] and Yuji Yamakawa[3]
[1]Graduate School of Engineering, The University of Tokyo, Japan
[2]Institute of Industrial Science, The University of Tokyo, Japan
[3]Interfaculty Initiative in Information Studies, The University of Tokyo, Japan
quan@g.ecc.u-tokyo.ac.jp, {mhirano, y-ymkw}@iis.u-tokyo.ac.jp

Abstract

*This work investigates the functionality of **Sem**antic information in **C**ontrastive **L**earning (SemCL). An advanced pretext task is designed: a contrast is performed between each object and its environment, taken from a scene. This allows the SemCL pretrained model to extract objects from their environment in an image, significantly improving the spatial understanding of the pretrained models. Downstream tasks of semantic/instance segmentation, object detection and depth estimation are implemented on PASCAl VOC, Cityscapes, COCO, KITTI, etc. SemCL pretrained models substantially outperform ImageNet pretrained counterparts and are competitive with well-known works on downstream tasks. The results suggest that a dedicated pretext task leveraging semantic information can be powerful in benchmarks related to spatial understanding. The code is available at https://github.com/sjiang95/semcl.*

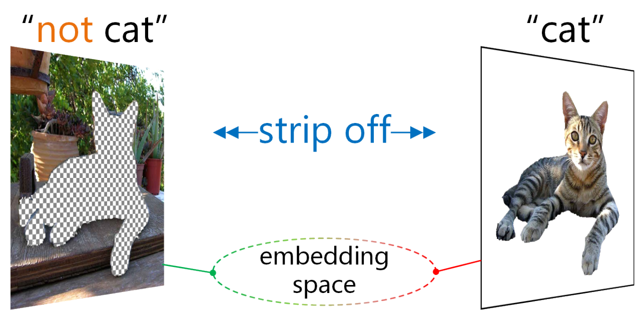

Figure 1: Perceptual cognition of SemCL. Leveraging semantic information, SemCL separates an object from its environment. Each sample is divided into objects and surroundings and considered as a contrastive pair. The task is to maximize the distance between the object and its environment in the embedding space.

1. Introduction

Within the field of visual representation learning, *contrastive learning* attracts special attention for its inspiring performance in transfer learning [57, 49, 31, 22, 8, 9]. The concept of contrastive learning can literally be explained as discovering the difference between positive and negative samples. The definition of positive-negative samples is one of the main subjects of contrastive learning, which directly determines the pretext task and the corresponding loss function.

We humans can recognize an object from its even complicated environment because we have learned and bound the typical geometric feature and the corresponding semantic concept of the object class. For example, we can recognize a cat in a scene thanks to our knowledge of its appearance (typical geometric feature) and the concept of *cat* (semantic information). Similarly, the target of prevalent visual tasks, such as semantic/instance segmentation, object detection, etc., can be summarized as *distinguishing and localizing subjects from surroundings*. Inspired by and to extend the spatial recognition mechanism to CV, we propose our method SemCL which directly teaches models to extract a subject from its environment. Utilizing publicly off-the-shelf datasets providing semantic labels, SemCL can be considered as a supervised contrastive learning approach whose samples consist of subjects and their corresponding surroundings (semantically *not the subject*). The pretext task of SemCL is to tell the difference between one subject and its surroundings. From a representation learning perspective, the pretext task is to maximize the distance (L^2 norm) between a subject and its environment in the embedding space (see Figure 1). The pretext task mimics the spatial recognition mechanism to discriminate objects and surroundings, which significantly improves the spatial information understanding of pretrained models.

In SemCL, contrast is performed pairwise between positive-negative pairs from one image/scene. Inter-scene contrast is considered inappropriate: only the contrasts

*Corresponding author.

among subject-surrounding pairs from one scene are considered valid. Therefore, paired InfoNCE loss is adopted. With the mechanism of contrasting only paired samples, SemCL is characterized by decoupling the number of negative samples from the batch size. Compared to MoCo [22], whose performance is positively correlated with batch size and thus increases hardware cost (e.g. $batchsize = 4096$ on 128 GPUs for ViT-B in MoCo v3 [9]) for satisfactory results, SemCL can achieve consistent yet competitive results on downstream tasks with relatively small batch size (e.g., $batchsize = 64$ @ 224×224). We adopt the MoCo v3 [9] as the pretraining framework.

The primary motivation of representation learning is to pretrain general representations that can be fine-tuned and transferred to downstream tasks. In this work, the SemCL pretrained models are benchmarked on semantic segmentation, object detection and instance segmentation, and depth estimation tasks. SemCL models substantially outperform their ImageNet pretrained counterparts due to the improved ability of spacial information understanding. In a system-level comparison, SemCL models also show gains over previous well-known works (e.g. Deeplabv3+ [7] in semantic segmentation, Mask2Former [10] in instance segmentation and Binsformer [35] in depth estimation).

2. Related Work

2.1. Contrastive Learning

The seminal idea of contrastive learning dates back at least to the 1990s [2, 3, 31], but it has become prominent in recent years thanks to the large pretrained models in the fields of NLP and CV [31]. As the name suggests, contrastive learning mines abstract representations by comparing between "similar" and "dissimilar" samples: minimizing the representation of "similar" samples and maximizing that of "dissimilar" samples in the embedding space. Formally, the "similar" inputs are called positive samples, and the "dissimilar" inputs are called negative samples.

Defining "similar" and "dissimilar" samples is an important topic in contrastive learning. In instance discrimination [57], which is a ubiquitous approach to define positive-negative samples in CV, each sample in a mini-batch $x \in \{x_1, \ldots, x_n\}$ is considered mutually exclusive to the rest. The encoded output $\theta(\cdot)$ of each sample is pushed away from the others in the embedding space, denoted as $\theta(x_1) \Leftrightarrow \{\theta(x_2), \ldots, \theta(x_n)\}$ ($\Leftrightarrow$ for push). Contrastive Predictive Coding (CPC) [49] defines positive-negative samples in a generative way: Given a sequential input $\{\ldots, x_{t-3}, x_{t-2}, x_{t-1}, x_t\}$, a context latent representation C_t is computed to predict the encoded outputs of future inputs $\{\theta(\mathbb{X}_{t+1}), \theta(\mathbb{X}_{t+2}), \theta(\mathbb{X}_{t+3}), \ldots\}$. Those predicted future embeddings are considered as positive samples to real future embeddings, and exclusive to some other irrelevant inputs.

2.2. Momentum Contrast

He et al. [22] proposed a momentum contrast framework, MoCo, for contrastive representation learning. MoCo summed unsupervised visual representation learning [57, 49, 26, 69, 25, 48, 1] up as dictionary look-up: the input data are represented and sampled by an encoder network θ_k as "keys" $k = \theta_k\left(x^k\right)$. The goal of contrastive learning is to train encoders θ_q by which a "query" $q = \theta_q(x^q)$ should be similar to its matching key and dissimilar to others. Since dictionary consistency is considered crucial for unsupervised learning with contrastive loss [22], MoCo introduces momentum updating. In the MoCo framework, the key coder θ_k is progressively updated by the weighted average of θ_k and the query coder θ_q.

$$\theta_k \leftarrow m\theta_k + (1-m)\,\theta_q, \tag{1}$$

where the momentum coefficient $m \in [0, 1)$ is set quite high ($m = 0.999$ in [22]) for slower key encoder update. The MoCo series has proven the power of the Momentum Contrast framework on feature transferring and is therefore employed as the basis for this work.

2.3. Labeled Data In Unsupervised Framework

Large networks are quite thirsty for training data, while labelling the large amount of collected data is a human resource consuming challenge. Apart from turning to unsupervised learning, which inherently avoids using labelled data, using unlabeled data in supervised learning is also a solution. Pseudo-Label [44, 38, 56] is a trick that uses (usually supervised) pretrained networks to generate features that can play the role of labels. And the generated pseudo-labels participate in the iterations of the target network just like annotations.

In contrast, Khosla et al. [28] also investigated the function of labelled data in unsupervised learning. As a contrastive learning research, Khosla et al. [28] intuitively uses the classification label to define samples within the same class are mutually positive, and those from other categories are negative. To exploit the label information, clusters of points belonging to the same class are pulled together in the embedding space, while clusters of samples from different classes are pushed apart [28]: $\left\{\theta\left(x_1^{c0}\right) \rightleftharpoons \theta\left(x_2^{c0}\right) \rightleftharpoons \theta\left(x_{\ldots}^{c0}\right) \rightleftharpoons \theta\left(x_n^{0}\right)\right\} \Leftrightarrow \left\{\theta\left(x_1^{c1}\right) \rightleftharpoons \theta\left(x_2^{c1}\right) \rightleftharpoons \theta\left(x_{\ldots}^{c1}\right) \rightleftharpoons \theta\left(x_m^{c1}\right)\right\}$ ($\rightleftharpoons$ for pull). The experimental result shows that supervised contrastive learning can improve both the accuracy and robustness of classifiers – [28] achieves a top-1 accuracy of 81.4% on the ImageNet (IN) dataset, which is 0.8% above the best result reported by ResNet-200 architecture [24]. Likewise, our work investigates the effectiveness of semantic information in unsupervised framework contrastive learning.

2.4. Semantic Information in Contrastive Learning

Other than [28], there are works that use small amounts of labelled data in unsupervised representation learning. Wang et al. [54] proposed a pixel-wise contrastive algorithm for semantic segmentation: for a pixel i, it is pulled towards pixels belonging to the same category and pushed away from pixels in other categories in the embedding space. This contrast is performed across images from the same dataset. In the Cityscapes test benchmark, [54] R101 achieves 79.2 mIoU, which is 1.1 points better than Deeplabv3 [6].

The attempt and results of works using labels in unsupervised representation learning, such as [28, 54], indicate that label information can bring gains to a fully unsupervised representation. It is promising to explore advanced pretext tasks to improve the performance of pretrained models in downstream tasks.

3. Method

3.1. SemCL Dataset

The core of this work is to define semantically contrastive pairs by utilizing off-the-shelf semantic labels, which are mappings between categories of objects and their coordinates on the raw image at the pixel level. As decided by the authors of a dataset, semantic labels can be color or grayscale images. Each predefined color or grayscale denotes a category, and the distribution of label pixels reflects the location of pixels belonging to a particular category in the raw image.

The SemCL dataset is generated by the mechanism shown in Figure 2, where the raw image and its semantic label are selected from the PASCAL VOC2012 dataset [14]. From the semantic label annotating two objects - a motorcycle and a rider - two binary masks BM0 and BM1 can be extracted to separately represent two annotated objects. Their inverted counterparts ¬BM0 and ¬BM1 represent pixels other than the target object. These binary masks are then bitwisely applied to the raw image to separate an object (Anchor) from its environment (¬Anchor), forming a semantically contrastive pair. Each element in a raw Anchor-¬Anchor pair is considered to be semantically opposite to the other.

In practice, the ***SemCL-1M*** dataset is produced from training sets of augmented PASCAL VOC2012 [14] (also known as SBD [20]), Cityscapes [11], ADE20K [67, 68] and COCO [36] (stuff+thing 2017 [4]) datasets. The components of the SemCL dataset are illustrated in Table 1. When generating the dataset, a threshold $t = 0.01$ is set to filter out too small (less than $\left[\sqrt{t} \times height\right] \times \left[\sqrt{t} \times width\right]$) objects that are considered semantically unsaturated.

3.2. Paired InfoNCE Loss

Since contrastive learning can be considered as a dictionary look-up task [22], it is assumed that within a set of keys $\{k_0, k_1, k_2, \dots\}$ of a dictionary, a query q matches only one key k_+ of the dictionary [22]. The value of the objective function should be low if q is similar to its positive key k_+ and dissimilar to all negative keys $\{k_0, k_1, k_2, \dots\} \setminus \{k_+\}$. In MoCo [22], the form of a contrastive loss function InfoNCE [49] is adopted

	VOC2012	Cityscapes	ADE20K	COCO	Total
# training samples	10,582	2,975	25,574	118,287	157,418
# SemCL pairs	14,203	20,280	270,218	712,212	1,016,913

Table 1: Components of the SemCL dataset. The SemCL datasets spawned from VOC2012(aug), Cityscapes, ADE20k and COCO 2017 are named SemCL-{VOC, City, ADE, COCO} respectively.

$$\mathcal{L}_q = -\log \frac{\exp\left(q \cdot k_+ / \tau\right)}{\sum_{i=0}^{K} \exp\left(q \cdot k_i / \tau\right)} \tag{2}$$

where τ is a scalar temperature hyperparameter controlling the strength of penalties on hard negative samples [50], e.g. in the instance discrimination task [49, 22], there may be potential positive samples in the randomly chosen *negative* samples. Summing over one positive and K negative samples, Equation (2) is the log loss of a (K+1)-way softmax-based classifier trying to classify q as k_+ [22]. While in our case an anchor instance q compares only with its own n augmentations (positive) $K^+ \coloneqq \left\{k_1^+, k_2^+, k_3^+, \dots, k_n^+\right\}$ and paired $n+1$ negatives $K^- \coloneqq \left\{k_0^-, k_1^-, k_2^-, k_3^-, \dots, k_n^-\right\}$ using paired InfoNCE

$$\mathcal{L}_q = -\sum_{j=t+1}^{n} \log \frac{\exp\left(q_t \cdot k_j^+ / \tau\right)}{\sum_{i=0}^{K^-} \exp\left(q_t \cdot k_i^- / \tau\right) + \exp\left(q_t \cdot k_j^+ / \tau\right)}. \tag{3}$$

For one anchor as query q_0 $(t = 0)$, the paired InfoNCE loss tries to classify it as $k_j^+, j \in [1, n]$. The next step is to go forward: classify the first positive q_1 $(t = 1)$ of the anchor as $k_j^+, j \in [2, n]$, and so on, as shown in the algorithm 1, Appendix A.1. To contrast with the paired InfoNCE, we refer to the original InfoNCE loss as *unpaired* InfoNCE hereafter. The behavior difference between unpaired and paired InfoNCE losses is compared in Figure 3. The paired InfoNCE only maximizes the distance between anchors and ¬anchors in the embedding space, neglecting irrelevant samples. The implementation of the paired InfoNCE loss is introduced in detail in Appendix A.2.

3.3. Pretext Task

Since the aim of this work is to enable networks to strip targets (anchors) from their surroundings (¬anchors), we use a restricted instance discrimination task [57]: a query and a key are positive pairs if they are different views (by

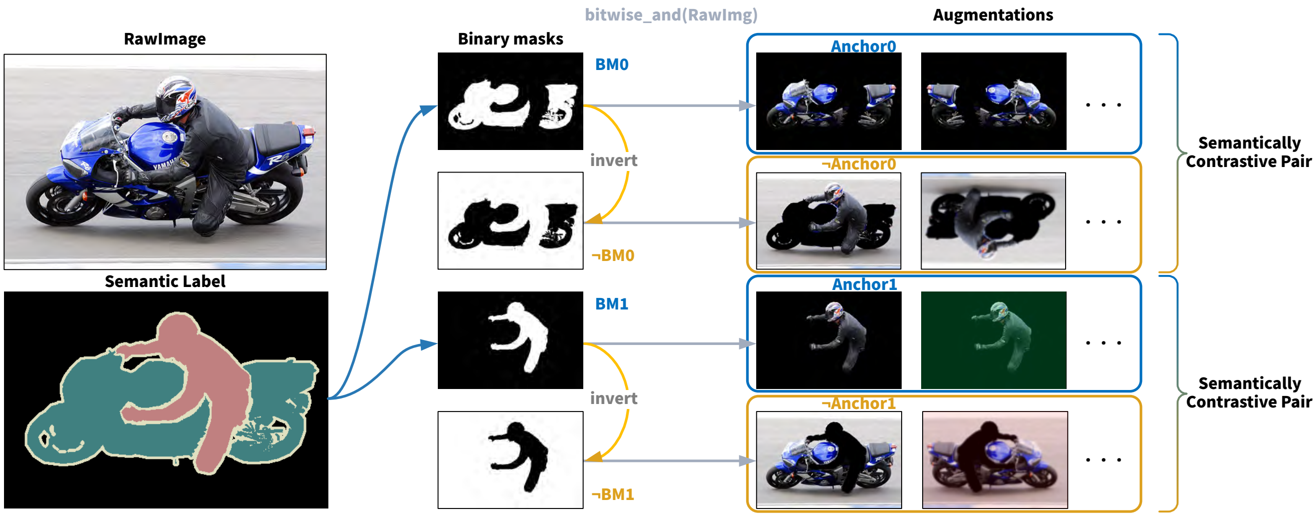

Figure 2: SemCL dataset. The semantic label of a sample is utilized to generate binary masks. Each binary mask depicts the pixel-level information of one object (anchor) in the raw image. Those binary masks, together with their inverts, are applied on the raw image to extract objects (anchors) and corresponding surroundings (¬anchors).

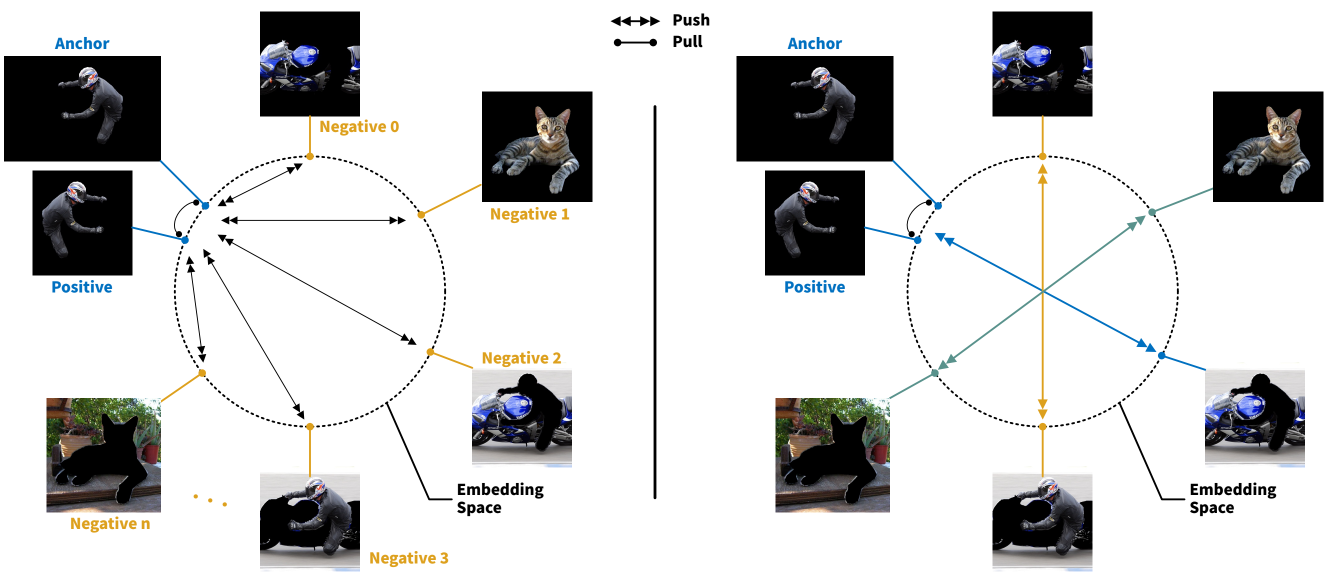

Figure 3: Comparison between unpaired (left) and paired InfoNCE losses (right). Left: a sample (anchor) is positive only to its own augmentations, while negative to others in the same mini-batch. Right: A sample (anchor) is positive only to its own augmentations, while negative only to its *paired* counterparts.

data augmentation) of the same anchor, and are negative pairs if they originate from an anchor and its corresponding surroundings (¬anchors). A PyTorch-style pseudocode is given in the Appendix A.2. The inputs are encoded by base encoder f_b and momentum encoder f_m, producing queries and keys. Theoretically, the backbone of f_b and f_m can be any architecture. In practice, we use ResNet [23] and Swin Transformer [37] because they are the most representative architectures in recent years. All keys are stacked along dimension 1 for paired InfoNCE loss.

4. Experiments

Considering the enormous scale difference between ImageNet and SemCL dataset, all SemCL models are initialized by corresponding IN pretrained weights, which are compared as counterparts in downstream tasks. The total number of pretraining iterations is set to $30k$ to minimize the significant variation in training time caused by scale differences among SemCL sub-datasets. We adopt AdamW [40] with $wd = 0.1$, and the OneCycle [46] learning rate schedule consisting of warmup [17] for the initial $3,750$ (warmup ratio 0.125) steps and cosine annealing [9, 39]. Following [9], the base lr is set to 1.5×10^{-4} and scaled linearly by $lr \times batchsize/256$ and temperature $\tau = 0.2$. The initial MoCo momentum is 0.99 and is gradually increased to 1 with a half cycle cosine schedule. The data augmentation strategy includes 224×224 random crops after random rescaling the original image with an area ratio in the range $[0.08, 1.0]$, followed by random color jitter, random grayscale, random Gaussian blur, and random horizontal flip. All batch norm (BN) layers are synchronized across GPUs (SyncBN [41]). Limited by the two dual-GPU nodes (RTX3090×2+Titan RTX×2) we use, a native batch size of 64 can be applied to all ResNets and up to Swin-B for Swin Transformers with crop size 224×224. For Swin-L in particular, we compromise by using a batch size of 32 with gradient accumulation over 2 iterations to achieve an equivalent batch size of 64.

SemCL aims to improve the understanding of spatial information of models for which downstream tasks of semantic segmentation, object detection and depth estimation are conducted. Unless otherwise stated, all SemCL backbones are pretrained on the corresponding SemCL sub-datasets. The implementation details of downstream tasks are listed Appendix A.3.

4.1. Semantic Segmentation

Backbone comparison. In the semantic segmentation benchmark on validation sets of VOC2012, Cityscapes, ADE20K and COCO 2017, SemCL models pretrained on the corresponding SemCL sub dataset are compared with ImageNet pretrained counterparts in Table 2. For VOC2012 and COCO, SemCL outperforms its IN counterparts by a maximum of 0.83 and 0.66 points respectively. SemCL can comprehensively outperform its IN counterparts on all backbones in the Cityscapes and ADE20k benchmarks.

System-level comparison. SemCL pretrained backbones are compared with other methods on semantic segmentation tasks in Table 3. On VOC2012, SemCL R50/R101 outperform Deeplabv3+ [7] by 2.37/3.12 points. The Deeplabv3+ [7] X-71 is 0.55 points behind SemCL R101 on Cityscapes. For CP^2 [51] ViT-S/16, SemCL Swin-S significantly outperforms it on all benchmarks by 4.97, 4.53 and 6.71 points. Compared to MAE [21] ViT-B, SemCL Swin-B achieves a gain of 0.58 points. A qualitative comparison between MAE [21] ViT-B and SemCL Swin-B is shown in Figure 4.

4.2. Object Detection and Instance Segmentation

Backbone comparison. In the benchmark for object detection and instance segmentation on the test set of VOC07 and validation sets of Cityscapes and COCO 2017, SemCL

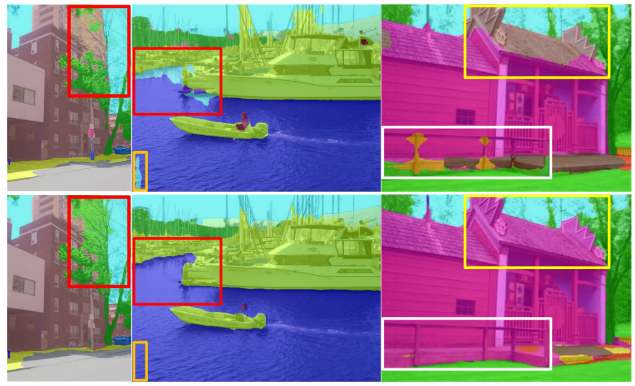

Figure 4: Qualitative comparison of semantic segmentation between MAE [21] ViT-B (top) and SemCL Swin-B (bottom) on ADE20K semantic segmentation validation set. Both decoders are UPerNet[58].

models pretrained on the corresponding SemCL subset are compared with IN pretrained counterparts in Table 4. SemCL comprehensively outperforms its IN counterparts on the Cityscapes and COCO benchmarks. In particular, on the instance segmentation task, SemCL improves by a maximum of 0.5 and 0.3 points on Cityscapes and COCO respectively. However, SemCL performs worse than its IN counterparts in VOC detection, which is considered to be a result of insufficient training samples in the SemCL-VOC dataset and is further investigated in the ablation study (see Appendix A.4).

Method	VOC2007 AP_{50}	Cityscapes† AP^{box}	AP^{mask}	COCO AP^{box}	AP^{mask}
MoCo v3[9] IN-1k	81.88	42.7	37.5	42.6	37.9
SemCL	81.57($-$0.31)	42.9(+0.2)	38.0(+0.5)	42.8(+0.2)	38.0(+0.1)

(a) R50

Method	VOC2007 AP_{50}	Cityscapes† AP^{box}	AP^{mask}	COCO AP^{box}	AP^{mask}
Swin[37] IN-22k	86.58	45.9	39.8	49.2	43.0
SemCL	86.53($-$0.05)	46.0(+0.1)	40.2(+0.4)	49.0($-$0.2)	43.3(+0.3)

(b) Swin-T

Table 4: Results of object detection and instance segmentation on various backbones w/ Cascade Mask R-CNN. All results are from our implementation. †: Sync BN is disabled due to gradient accumulation.

System-level comparison. Table 5 compares the object detection and instance segmentation results of SemCL backbones with other models. On Cityscapes and COCO, SemCL backbones comprehensively outperform their twin-born MoCo Specifically, SemCL ResNet50 backbones achieve $+1.7$ over MoCo IG-1B [22] in COCO detection and

Backbone	Method	VOC2012	Cityscapes	ADE20K	COCO
R50	MoCo v3[9] IN-1k†	79.65	79.01	42.71	40.59
	SemCL	79.57(−0.08)	79.13(+0.12)	42.96(+0.25)	40.80(+0.21)
R101	timm[55] IN-1k	81.82	79.87	44.43	41.43
	SemCL	81.97(+0.15)	80.10(+0.23)	44.66(+0.23)	41.58(+0.15)
Swin-T	Swin[37] IN-22k	80.90	78.67	44.60	42.39
	SemCL	81.73(+0.83)	78.91(+0.26)	44.72(+0.12)	42.53(+0.14)
Swin-S	Swin[37] IN-22k	83.71	81.25	47.62	44.75
	SemCL	84.47(+0.76)	81.53(+0.28)	47.91(+0.29)	44.57(−0.18)
Swin-B	Swin[37] IN-22k	84.64	81.61	48.21	44.81
	SemCL	84.67(+0.03)	81.94(+0.33)	48.68(+0.47)	45.47(+0.66)

† Although MoCo v3 [9] is originally based on ViT, R50 pretrained models are provided in their repo https://github.com/facebookresearch/moco-v3.

Table 2: SemCL vs. ImageNet pretrained backbones on VOC2012, Cityscapes, ADE20K and COCO semantic segmentation (mIoU). All results are from our implementation.

Method	Backbone	mIoU
Deeplabv3+[7]	R50	77.2
Supervised-IN[22]	R50	74.4
MoCo IG-1B[22]	R50	75.5
BYOL[18]	R50	76.3
SemCL	R50	**79.57**
Deeplabv3+[7]	R101	78.85
Res2Net[15]	Res2Net-101	80.2
DFN[61]	R101	80.60
SemCL	R101	**81.97**
CP^2[51]	ViT-S/16	79.5
SemCL	Swin-S	**84.47**
Leopart[70]	ViT-B/8	76.3
SemCL	Swin-B	**84.67**

(a) VOC2012

Method	Backbone	mIoU
MoCo IN-1M[22]	R50	72.5
MoCo IG-1B[22]	R50	73.6
Supervised-IN[22]	R50	74.6
CP^2[51]	R50	76.5
SemCL	R50	**79.13**
Deeplabv3+[7]	X-71	79.55
SemCL	R101	**80.10**
Trans4Trans-T[62]	PVTv2-B1	78.23
SemCL	Swin-T	**78.91**
CP^2[51]	ViT-S/16	77.0
Trans4Trans-S[62]	PVTv2-B2	80.02
SemCL	Swin-S	**81.53**

(b) Cityscapes

Method	Backbone	mIoU
CP^2[51]	R50	39.2
PSPNet[66]	R50	42.78
SemCL	R50	**42.96**
PSPNet[66]	R101	43.29
SemCL	R101	**44.66**
CP^2[51]	ViT-S/16	41.2
Seg-S/16[47]	ViT-S	45.37
SemCL	Swin-S	**47.91**
Seg-B/16[47]	Vi-B/16	48.06
MAE[21]	ViT-B	48.1
SemCL	Swin-B	**48.68**

(c) ADE20K

Table 3: Comparison with previous best results on validation sets of VOC2012, Cityscapes, and ADE20K semantic segmentation (mIoU). Best results are **bolded**.

+5.1/+0.6 in Cityscapes/COCO instance segmentation tasks. For SemCL Swin-T backbones, SemCL achieves +1.2/+0.1 over Focal [60] on COCO detection/instance segmentation. Figure 5 demonstrates a qualitative comparison between Mask2Former [10] and SemCL Swin-T on Cityscapes validation set. With the improved spatial information understanding, SemCL model clearly discriminates (even overlapped) instances in Figures 5a and 5b and successfully classify part of an instance to a defined category as in Figure 5c.

4.3. Depth Estimation

Backbone comparison. SemCL pretrained backbones are compared with corresponding IN pretrained ones in Table 6. For cityscapes, SemCL comprehensively outperforms its IN pretrained counterparts. In particular, the ResNet50 backbones improve the RMSE by 0.138 points. For KITTI and NTUv2, all SemCL backbones show improvements over the IN pretrained ones. The SemCL pretrained Swin-L model achieves improvements of 0.001/0.026 points on KITTI AbsRel/RMSE, and the SemCL pretrained ResNet50 model achieves improvements of 0.003/0.005 points on NYUv2.

System-level comparison. We report the comparison of SemCL pretrained backbones with the previous state-of-the-art method Binsformer[35] and other works in Table 7. As for the Cityscapes results shown in Table 7a, SemCL pretrained ResNet50 significantly outperforms SDC [53] by −0.089 and −2.218 points on AbsRel and RMSE respectively. And SD-SSMDE [43] lags behind SemCL by 0.005/3.742 points. Meanwhile, the test on KITTI (Eigen split [13]) is shown in Table 7b, SemCL pretrained ResNet50 and Swin-T comprehensively outperform the previous SOTA

Method	Backbone	AP_{50}
Supervised-IN[22]	R50	81.3
MoCo IN-1M[22]	R50	81.5
MoCo IG-1B[22]	R50	82.2
MoCo v2 800ep[8]	R50	**82.5**
SemCL	R50	81.57
UP-DETR/300[12]	*	80.1
OW-DETR[19]	DDETR	82.1
SemCL	Swin-T	**86.53**

(a) Object det. on VOC2007 test.

Method	Backbone	AP^{mask}
MoCo IN-1M[22]	R50	32.3
MoCo IG-1B[22]	R50	32.9
Supervised-IN[22]	R50	32.9
Mask2Former[10]	R50	37.4
SemCL	R50	**38.0**
BoundaryFormer[30]	*	38.3
Mask2Former[10]	Swin-T	39.7
SemCL	Swin-T	**40.2**

(b) Instance seg. on Cityscapes val.

Method	Backbone	AP^{box}	AP^{mask}
Supervised-IN[22]	R50	40.6	36.8
MoCo IN-1M[22]	R50	40.8	36.9
MoCo IG-1B[22]	R50	41.1	37.4
SemCL	R50	**42.8**	**38.0**
MST[34]	Swin-T	42.7	38.8
VSA[63]	Swin-T	46.9	42.1
Focal[60]	Focal-Base	47.8	43.2
SemCL	Swin-T	**49.0**	**43.3**

(c) Object det. and instance seg. on COCO val2017.

Table 5: System-level comparison on object detection and instance segmentation. The best results are **bolded**. * claims their backbones are CNN-based, but are considered and compared as transformer-based networks, since transformer blocks are massively used.

Backbone	Method	Cityscapes		KITTI		NYUv2	
		AbsRel	RMSE	AbsRel	RMSE	AbsRel	RMSE
R50	MoCo v3[9] IN-1k	0.138	4.837	0.059	2.337	0.120	0.405
	SemCL	0.138	4.699(−0.138)	0.058(−0.001)	2.337	0.117(−0.003)	0.400(−0.005)
Swin-T	Swin[37] IN-22k	0.134	4.643	0.057	2.247	0.114	0.386
	SemCL	0.133(−0.001)	4.577(−0.066)	0.057	2.237(−0.010)	0.112(−0.002)	0.381(−0.005)
Swin-L	Swin[37] IN-22k	0.130	4.536	0.053	2.130	0.095	0.329
	SemCL	0.128(−0.002)	4.426(−0.110)	0.052(−0.001)	2.104(−0.026)	0.094(−0.001)	0.329

Table 6: SemCL vs. ImageNet pretraining on Cityscapes, KITTI and NYUv2 depth estimation (lower is better). All results are from our implementation.

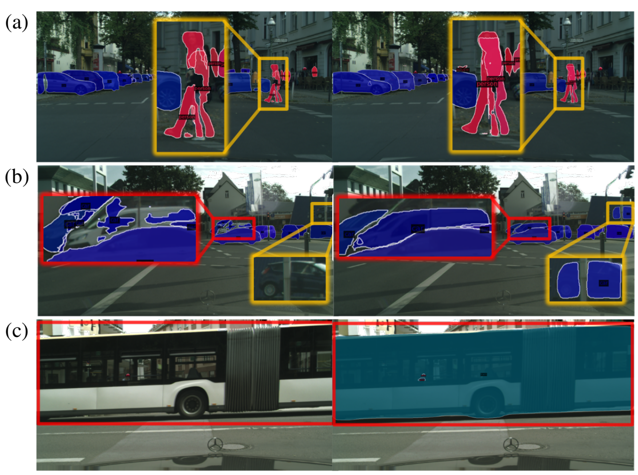

Figure 5: Qualitative comparison of instance segmentation between Mask2Former [10] (left) and SemCL (right) Swin-T on Cityscapes validation set.

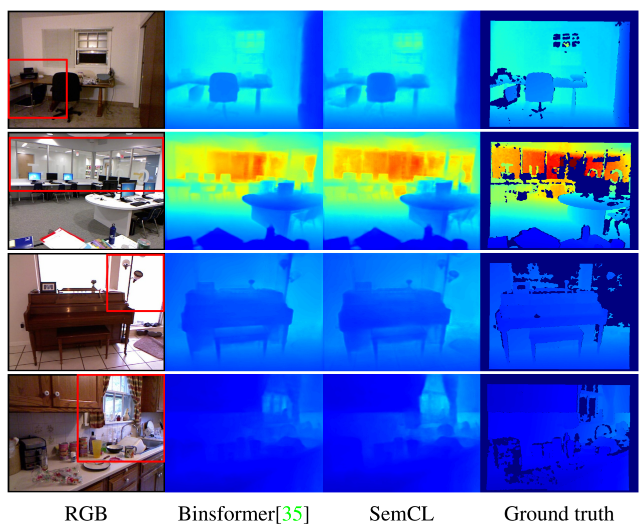

Figure 6: Qualitative comparison of depth estimation between Binsformer[35] and SemCL Swin-T on the NYUv2 test set.

Binsformer. In particular, SemCL ResNet50 achieves a gain of $-0.003/-0.089$ over Binsformer, and SemCL Swin-L is on par with Binsformer Swin-L. On NYUv2 (see Table 7c), the gains of SemCL pretrained ResNet50 are also high with -0.010 AbsRel and -0.173 RMSE. The SemCL Swin-L RMSE is 0.001 points lower than the SOTA Binsformer. Qualitative comparison between Binsformer [35] and SemCL Swin-T is given in Figure 6.

Method	Backbone	AbsRel	RMSE
UpProj[29]	R50	0.257[53]	7.273[53]
Pad-net[59]	R50	0.246[53]	7.117[53]
TRL[64]	R50	0.234[53]	7.104[53]
Pilzer, et al.[45]	-	0.440	5.443
SDC-Depth[53]	R50	0.227	6.917
DTS-Depth[27]	MobileNetv2	0.167	7.785
struct2depth[5]	-	0.151	7.024
SD-SSMDE[43]	R50	0.143	8.441
SemCL	R50	**0.138**	**4.699**

(a) Cityscapes

Method	Backbone	AbsRel	RMSE
EPCDepth [42]	R50	0.091	4.207
Binsformer[35]	R50	0.061	2.426
SemCL	R50	**0.058**	**2.337**
MonoViT[65]	-	0.098	4.333
Binsformer [35]	Swin-T	0.058	2.286
SemCL	Swin-T	**0.057**	**2.237**
DepthFormer [33]	Swin-L +R50$_{C1}$	**0.052**	2.143
Binsformer[35]	Swin-L	**0.052**	**2.098**
SemCL	Swin-L	**0.052**	2.104

(b) KITTI

Method	Backbone	AbsRel	RMSE
Pad-net [59]	R50	0.214	0.792
TRL [64]	R50	0.144	0.501
SDC-Depth [53]	R50	0.128	0.497
UpProj [29]	R50	0.127	0.573
SemCL	R50	**0.117**	**0.400**
Binsformer [35]	Swin-T	0.113	**0.379**
SemCL	Swin-T	**0.112**	0.381
Binsformer [35]	Swin-L	**0.094**	0.330
SemCL	Swin-L	**0.094**	**0.329**

(c) NYUv2

Table 7: System-level comparison on depth estimation. Best results (lower is better) are **bolded**.

4.4. Attention Map Visualization

To improve the qualitative understanding of the SemCL representation, we visualize attention maps of Swin-T pretrained on SemCL-VOC and ImageNet using `PyTorch library for CAM methods` [16] with the Score-CAM method [52]. These samples in Figure 7 selected from VOC2012 val and have not been seen by the pretrained models are characterized by simple centered subjects (Figure 7a), complex scene with occlusion (Figure 7b) and multiple subjects (Figures 7c and 7d).

In Figure 7a, the IN pretrained model failed to focus on the subject, while the SemCL model accurately depicted the area of the main subject. Meanwhile, SemCL performs better in complex scenes containing foreground-background relationships, as shown in Figure 7b. The IN pretrained model pays no attention to an occluded object, but the SemCL model recognizes that the occluded part also belongs to the horse, as shown in Figure 7b. In Figure 7c, the IN model directly ignores the distant person, but the SemCL model not only annotates two people but also focuses more on the nearby one, suggesting that the SemCL model can understand the spatial distribution of subjects in an image. Also, in Figure 7d, the SemCL model successfully recognizes one plant and its pot as one object.

Summary. SemCL pretrained backbones can substantially outperform ImageNet pretrained counterparts on semantic segmentation, object detection, instance segmentation and depth estimation tasks. Using semantically contrastive pairs generated from off-the-shelf datasets, SemCL significantly improves the spatial information understanding of pretrained models. Such an improvement is also promising to benefit more challenging tasks including generative ones [32]. Ablation study concerning paired/unpaired InfoNCE loss, pretraining batch size, training length and dataset scale are given in Appendix A.4.

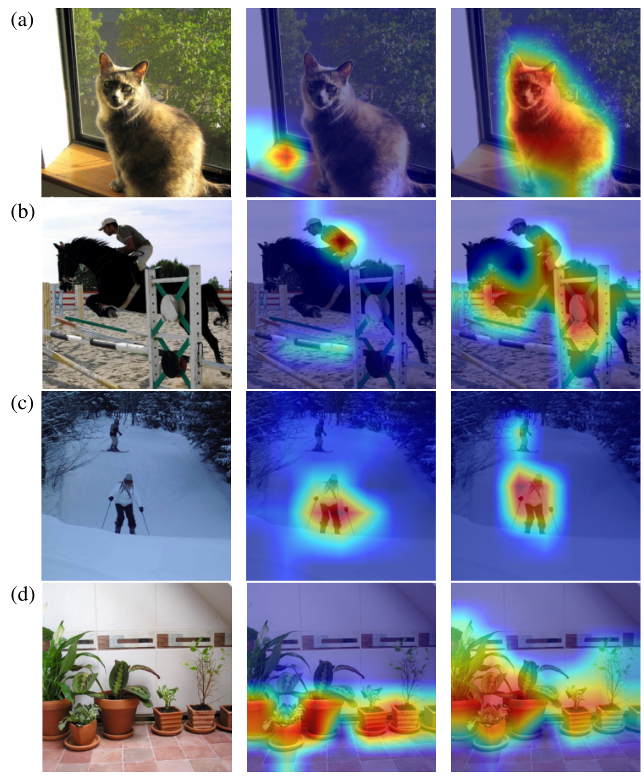

Figure 7: Swin-T attention map visualization. Left: raw image. Middle: attention maps of the IN pretrained model. Right: attention maps of the SemCL pretrained model.

5. Conclusion

We investigate the effectiveness of semantic information in the contrastive learning pretext task on the spatial understanding ability of models. Tests are conducted on semantic/instance segmentation, object detection and depth estimation tasks with both ResNets and Swin Transformers, and SemCL pretrained backbones substantially outperform their ImageNet pretrained counterparts. By supporting small batch sizes and fast pretraining, SemCL is a lightweight yet effective approach. In contrastive representation learning, if dedicated pretext task is designed properly, *four ounces can move a thousand pounds*. We hope that SemCL will inspire more advanced pretext tasks for contrastive learning.

References

[1] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *Advances in neural information processing systems*, 32, 2019. 2

[2] Suzanna Becker and Geoffrey E Hinton. Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, 355(6356):161–163, 1992. 2

[3] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a "siamese" time delay neural network. *Advances in neural information processing systems*, 6, 1993. 2

[4] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Computer vision and pattern recognition (CVPR), 2018 IEEE conference on*. IEEE, 2018. 3

[5] Vincent Casser, Soeren Pirk, Reza Mahjourian, and Anelia Angelova. Unsupervised monocular depth and ego-motion learning with structure and semantics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 8

[6] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 3

[7] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 2, 5, 6

[8] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 1, 7

[9] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9640–9649, 2021. 1, 2, 5, 6, 7

[10] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1290–1299, 2022. 2, 6, 7

[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 *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 3

[12] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. Up-detr: Unsupervised pre-training for object detection with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1601–1610, 2021. 7

[13] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27, 2014. 6

[14] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 3

[15] Shang-Hua Gao, Ming-Ming Cheng, Kai Zhao, Xin-Yu Zhang, Ming-Hsuan Yang, and Philip Torr. Res2net: A new multi-scale backbone architecture. *IEEE transactions on pattern analysis and machine intelligence*, 43(2):652–662, 2019. 6

[16] Jacob Gildenblat and contributors. Pytorch library for cam methods. `https://github.com/jacobgil/pytorch-grad-cam`, 2021. 8

[17] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017. 5

[18] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 33:21271–21284, 2020. 6

[19] Akshita Gupta, Sanath Narayan, KJ Joseph, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. Ow-detr: Open-world detection transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9235–9244, 2022. 7

[20] Bharath Hariharan, Pablo Arbelaez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *International Conference on Computer Vision (ICCV)*, 2011. 3

[21] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 5, 6

[22] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 1, 2, 3, 5, 6, 7

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016. 2

[25] Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. In *International Conference on Machine Learning*, pages 4182–4192. PMLR, 2020. 2

[26] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *International Conference on Learning Representations*, 2019. 2

[27] Hatem Ibrahem, Ahmed Salem, and Hyun-Soo Kang. Dts-depth: Real-time single-image depth estimation using depth-to-space image construction. *Sensors*, 22(5):1914, 2022. 8

[28] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673, 2020. 2, 3

[29] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *2016 Fourth international conference on 3D vision (3DV)*, pages 239–248. IEEE, 2016. 8

[30] Justin Lazarow, Weijian Xu, and Zhuowen Tu. Instance segmentation with mask-supervised polygonal boundary transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4382–4391, 2022. 7

[31] Phuc H. Le-Khac, Graham Healy, and Alan F. Smeaton. Contrastive representation learning: A framework and review. *IEEE Access*, 8:193907–193934, 2020. 1, 2

[32] Yuheng Li, Yijun Li, Jingwan Lu, Eli Shechtman, Yong Jae Lee, and Krishna Kumar Singh. Contrastive learning for diverse disentangled foreground generation. In *European Conference on Computer Vision*, pages 334–351. Springer, 2022. 8

[33] Zhenyu Li, Zehui Chen, Xianming Liu, and Junjun Jiang. Depthformer: Exploiting long-range correlation and local information for accurate monocular depth estimation. *arXiv preprint arXiv:2203.14211*, 2022. 8

[34] Zhaowen Li, Zhiyang Chen, Fan Yang, Wei Li, Yousong Zhu, Chaoyang Zhao, Rui Deng, Liwei Wu, Rui Zhao, Ming Tang, et al. Mst: Masked self-supervised transformer for visual representation. *Advances in Neural Information Processing Systems*, 34:13165–13176, 2021. 7

[35] Zhenyu Li, Xuyang Wang, Xianming Liu, and Junjun Jiang. Binsformer: Revisiting adaptive bins for monocular depth estimation. *arXiv preprint arXiv:2204.00987*, 2022. 2, 6, 7, 8

[36] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 3

[37] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 4, 5, 6, 7

[38] Vishnu Suresh Lokhande, Songwong Tasneeyapant, Abhay Venkatesh, Sathya N Ravi, and Vikas Singh. Generating accurate pseudo-labels in semi-supervised learning and avoiding overconfident predictions via hermite polynomial activations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11435–11443, 2020. 2

[39] Ilya Loshchilov and Frank Hutter. SGDR: stochastic gradient descent with warm restarts. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. 5

[40] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 5

[41] Chao Peng, Tete Xiao, Zeming Li, Yuning Jiang, Xiangyu Zhang, Kai Jia, Gang Yu, and Jian Sun. Megdet: A large mini-batch object detector. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6181–6189, 2018. 5

[42] Rui Peng, Ronggang Wang, Yawen Lai, Luyang Tang, and Yangang Cai. Excavating the potential capacity of self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15560–15569, 2021. 8

[43] Andra Petrovai and Sergiu Nedevschi. Exploiting pseudo labels in a self-supervised learning framework for improved monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1578–1588, 2022. 6, 8

[44] Hieu Pham, Zihang Dai, Qizhe Xie, and Quoc V Le. Meta pseudo labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11557–11568, 2021. 2

[45] Andrea Pilzer, Dan Xu, Mihai Puscas, Elisa Ricci, and Nicu Sebe. Unsupervised adversarial depth estimation using cycled generative networks. In *2018 international conference on 3D vision (3DV)*, pages 587–595. IEEE, 2018. 8

[46] Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, volume 11006, pages 369–386. SPIE, 2019. 5

[47] Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7262–7272, 2021. 6

[48] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *European conference on computer vision*, pages 776–794. Springer, 2020. 2

[49] Aaron Van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv e-prints*, pages arXiv–1807, 2018. 1, 2, 3

[50] Feng Wang and Huaping Liu. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504, 2021. 3

[51] Feng Wang, Huiyu Wang, Chen Wei, Alan Yuille, and Wei Shen. Cp 2: Copy-paste contrastive pretraining for semantic segmentation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXX*, pages 499–515. Springer, 2022. 5, 6

[52] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-cam: Score-weighted visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 24–25, 2020. 8

[53] Lijun Wang, Jianming Zhang, Oliver Wang, Zhe Lin, and Huchuan Lu. Sdc-depth: Semantic divide-and-conquer network for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 541–550, 2020. 6, 8

[54] Wenguan Wang, Tianfei Zhou, Fisher Yu, Jifeng Dai, Ender Konukoglu, and Luc Van Gool. Exploring cross-image pixel contrast for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7303–7313, 2021. 3

[55] Ross Wightman, Hugo Touvron, and Hervé Jégou. Resnet strikes back: An improved training procedure in timm. *arXiv preprint arXiv:2110.00476*, 2021. 6

[56] Hao Wu and Saurabh Prasad. Semi-supervised deep learning using pseudo labels for hyperspectral image classification. *IEEE Transactions on Image Processing*, 27(3):1259–1270, 2017. 2

[57] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742, 2018. 1, 2, 3

[58] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 5

[59] Dan Xu, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Padnet: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 675–684, 2018. 8

[60] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Xiyang Dai, Bin Xiao, Lu Yuan, and Jianfeng Gao. Focal attention for long-range interactions in vision transformers. *Advances in Neural Information Processing Systems*, 34:30008–30022, 2021. 6, 7

[61] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Learning a discriminative feature network for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1857–1866, 2018. 6

[62] Jiaming Zhang, Kailun Yang, Angela Constantinescu, Kunyu Peng, Karin Müller, and Rainer Stiefelhagen. Trans4trans: Efficient transformer for transparent object and semantic scene segmentation in real-world navigation assistance. *IEEE Transactions on Intelligent Transportation Systems*, 23(10):19173–19186, 2022. 6

[63] Qiming Zhang, Yufei Xu, Jing Zhang, and Dacheng Tao. Vsa: learning varied-size window attention in vision transformers. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXV*, pages 466–483. Springer, 2022. 7

[64] Zhenyu Zhang, Zhen Cui, Chunyan Xu, Zequn Jie, Xiang Li, and Jian Yang. Joint task-recursive learning for semantic segmentation and depth estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 235–251, 2018. 8

[65] Chaoqiang Zhao, Youmin Zhang, Matteo Poggi, Fabio Tosi, Xianda Guo, Zheng Zhu, Guan Huang, Yang Tang, and Stefano Mattoccia. Monovit: Self-supervised monocular depth estimation with a vision transformer. *arXiv preprint arXiv:2208.03543*, 2022. 8

[66] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 6

[67] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017. 3

[68] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019. 3

[69] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6002–6012, 2019. 2

[70] Adrian Ziegler and Yuki M Asano. Self-supervised learning of object parts for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14502–14511, 2022. 6