SegRCDB: Semantic Segmentation via Formula-Driven Supervised Learning

Risa Shinoda[1], Ryo Hayamizu[1], Kodai Nakashima[1], Nakamasa Inoue[2], Rio Yokota[2], Hirokatsu Kataoka[1]
[1]National Institute of Advanced Industrial Science and Technology (AIST)
[2]Tokyo Institute of Technology

Abstract

Pre-training is a strong strategy for enhancing visual models to efficiently train them with a limited number of labeled images. In semantic segmentation, creating annotation masks requires an intensive amount of labor and time, and therefore, a large-scale pre-training dataset with semantic labels is quite difficult to construct. Moreover, what matters in semantic segmentation pre-training has not been fully investigated. In this paper, we propose the Segmentation Radial Contour DataBase (SegRCDB), which for the first time applies formula-driven supervised learning for semantic segmentation. SegRCDB enables pre-training for semantic segmentation without real images or any manual semantic labels. SegRCDB is based on insights about what is important in pre-training for semantic segmentation and allows efficient pre-training. Pre-training with SegRCDB achieved higher mIoU than the pre-training with COCO-Stuff for fine-tuning on ADE-20k and Cityscapes with the same number of training images. SegRCDB has a high potential to contribute to semantic segmentation pre-training and investigation by enabling the creation of large datasets without manual annotation. The SegRCDB dataset will be released under a license that allows research and commercial use. Code is available at: https://github.com/dahlian00/SegRCDB

1. Introduction

Preparing a semantic segmentation dataset requires pixel-level, dense annotation, and therefore, creating a fully annotated dataset incurs a huge amount of effort. For a dataset such as Cityscapes [7], around 90 minutes per image is required for pixel-level annotation. This makes it difficult to create a large semantic segmentation dataset.

To perform training with a limited dataset, using a pre-trained model with a large-scale image dataset is a promising method for enhancing network performance in terms of recognition accuracy. The use of a model pre-trained on a larger-scale image dataset has become a standard approach. Undoubtedly, ImageNet [8] is one of the de-facto-standard

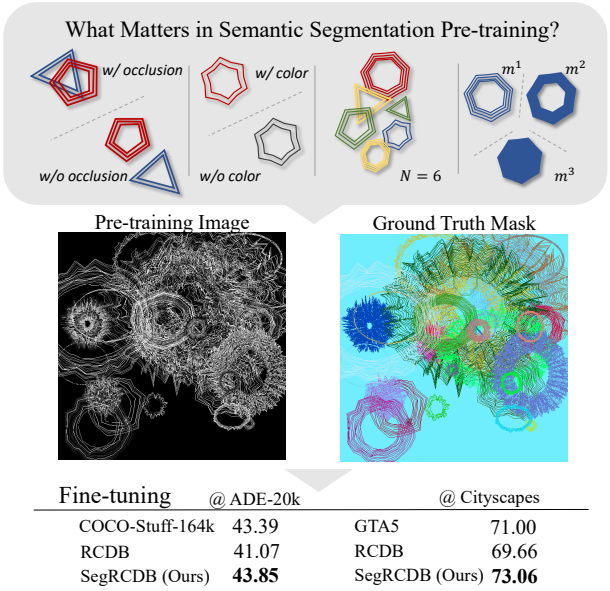


Figure 1. **Segmentation Radial Contour DataBase (SegRCDB).** We developed the first formula-driven supervised learning (FDSL) method for semantic segmentation. We created a precise pixel-wise ground truth mask without manual effort.

datasets, even in the semantic segmentation field. However, ethical issues have been reported in terms of dataset biases and privacy violations [24, 40, 41]. There have also been reports [42, 34] that one of the most frequently used segmentation datasets (Microsoft COCO [18]) also raises concerns regarding transfer learning due to ethical issues.

To overcome these technical (manual annotation) and ethical (fairness and dataset transparency) problems, a synthetic dataset can be implemented to construct a pre-trained segmentation model. Synthetic datasets for semantic segmentation have been created for specialized use in certain domains [27, 29, 22]. However, even though the annotation time is reduced, McCormac *et al.* reported that it took up to approximately one month using 4-12 GPUs to produce a 5M-image dataset [22]. For automatic dataset creation, we must define a 3D scene design that includes object categories and positional content, and consider the camera

trajectory. There is still potential for more efficient development of synthetic segmentation data that is applicable to general domains.

However, there has not been enough research to determine what factors are effective in pre-training semantic segmentation. It is not clear which dataset parameters are useful for pre-training semantic segmentation, such as the number of classes and the presence of occlusions. Identifying these critical parameters will allow us to build a synthetic dataset that is more efficient for pre-training semantic segmentation.

Formula-driven supervised learning (FDSL) [14] has been proposed and improved in the context of visual representation learning without real images. FDSL enables the automatic construction of large-scale image datasets through simultaneous image and label generation based on a simple mathematical formula. It was reported that the image dataset, which consists of complicated contours (e.g., Radial Contour DataBase; RCDB), allows effective image classification [12]. The object contours are mainly captured in the pre-training phase, and the visual representation is transferred to determine object categories in a real image. The FDSL framework has not been applied to acquiring a visual representation for semantic segmentation.

In this paper, we propose the first formula-driven supervised learning for semantic segmentation and create the SegRCDB dataset. We assume that a model pre-trained using radial contours can further improve the recognition ability of semantic segmentation since semantic masks are assigned to radial contour areas. To take advantage of making image patterns and ground truth masks based on a simple equation without real-image collection or manual annotation, we thoroughly investigated the effectiveness of different configurations (e.g., occlusion between objects, types of mask annotation) in pre-training. The SegRCDB will contribute to semantic segmentation tasks and reduce the time and effort required to create ground truths for large-scale image datasets.

The main contributions of this study are as follows:

- We developed the first FDSL method for semantic segmentation. We created a precise pixel-wise mask (Figure 1) without any manual effort.
- We investigated what elements are effective for improving the accuracy in pre-training for semantic segmentation, and created SegRCDB based on the results.
- Our SegRCDB has great potential for effectively pre-training a semantic segmentation model based on a massive amount of pixel-level ground truth. The proposed method performed better than the COCO-Stuff-164k baseline (e.g., 43.39 vs. 43.85 mIoU on ADE-20k) and the GTA5 baseline (e.g., 71.00 vs. 73.06 mIoU on Cityscapes).

2. Related Work

Here, we will limit the discussion to studies that are closely related to our proposed method.

Semantic segmentation. The earlier semantic segmentation models were implemented on CNN backbones and their heads [20, 2, 28, 5, 9]. In semantic segmentation tasks, we must effectively acquire context information and perform precise labeling for a whole image. This property is inherited by Transformer models [33, 37, 11, 19]. In particular, Swin Transformer [19] is frequently used as a strong baseline model due to its ability to capture spatial features. We also employed the Swin Transformer model in the experiments described in this paper.

Automatic/semi-automatic segmentation pre-training. One way to reduce annotation costs in semantic segmentation is by generating synthetic images and/or assigning ground truth masks. Previous studies have shown the effectiveness of increasing the ground truth by a pseudo-labeling technique [46, 45, 30, 43, 17], active learning [36, 31, 25] or by creating synthetic data and ground truth labels [16]. However, these techniques may produce a biased dataset with the dominant class or images when real images are used. Synthetic datasets also contribute to pre-training methods such as GTA5 [27] and SYNTHIA [29]. Large-scale synthetic datasets for semantic segmentation decrease the manual annotation time, but they still need to create and define models and environments.

Ethical issues with image datasets. Large-scale datasets have been reported to have ethical issues. For semantic segmentation, COCO-Stuff [4] is one of the largest datasets, and it contains 164k images from the COCO dataset. However, it has been reported that there are biases in the COCO dataset in relation to gender and race [42, 34]. Even the de-facto standard ImageNet dataset is associated with fairness and privacy problems due to biased inputs and human-related images [24, 40].

FDSL. To overcome dataset-related problems with real images, formula-driven supervised learning (FDSL) pre-training methods have been developed [14]. In FDSL, a simple mathematical formula (e.g., fractals [14, 1, 23, 12], tiling patterns [13], and contours [12]) allows automatic generation of image patterns and their training labels at the same time. This framework makes it possible to create large-scale pre-training datasets without any real images. Recent studies have verified that a mathematical formula can be used to pre-train models for video [15] and 3D objects [39, 38]. Moreover, Kataoka *et al.* [12] revealed that an image dataset consisting of complex contours (e.g., Radial Contour DataBase; RCDB) was very effective at pre-training models for image classification. We believe that RCDB is also effective for semantic segmentation because a model pre-trained using RCDB can acquire accurate object contours. Past studies (e.g., [3, 32]) have reported that

recognizing boundaries is important for semantic segmentation, and is also significant in segmentation pre-training.

3. Investigation Policy for Semantic Segmentation Pre-training

In this section, we present a framework for investigating what is most important for improving the pre-training accuracy for semantic segmentation without relying on a collection of real images. We first describe our investigation policy involving seven factors to be parameterized when synthesizing a dataset based on a formula-driven approach and then propose a simple yet effective method for dataset synthesis as well as a pre-training method that enables us to empirically investigate the effectiveness of each factor. For synthetic image generation using a formula-driven approach, we can create clear patterns and perfect semantic labels to easily analyze disassembled components in pre-training for semantic segmentation. We are the first to propose a pre-training method for semantic segmentation with the formula-driven supervised method.

Setting. This paper considers the setting where a training dataset for semantic segmentation consists of pairs of images and ground truth masks. Specifically, a dataset is given by $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{m}_i)\}_{i=1}^{N}$ where $\boldsymbol{x}_i$ is an image, $\boldsymbol{m}_i$ is a ground truth mask, and N is the number of images. Each pixel in a mask $\boldsymbol{m}_i$ represents a category label $c \in \{0, 1, 2, \cdots, C\}$ for its corresponding pixel at $\boldsymbol{x}_i$, where C is the number of object categories. $c = 0$ is used to indicate the background.

We assume that an image $\boldsymbol{x}_i$ is composed of M_i individual object instances $\{\boldsymbol{o}_j\}_{j=1}^{M_i}$ and a background, where the instances are ordered from the back to the front, *i.e.*, $\boldsymbol{o}_1$ is the backmost instance and $\boldsymbol{o}_{M_i}$ is the frontmost instance. Each instance has a category label c_j and a binary mask $\boldsymbol{b}_j \in \{0, 1\}^{W \times H}$ that describes the pixel-level appearance of the instance, where W and H are the image width and height, respectively. Under this assumption, the mask $\boldsymbol{m}_i$ is obtained by

$$[\boldsymbol{m}_i]_{p,q} = c_{j^*}, \quad j^* = \max(\{j : [\boldsymbol{b}_j]_{p,q} = 1\} \cup \{0\}) \quad (1)$$

where $[\cdot]_{p,q}$ indicates the element at pixel position (p, q) within the image. The max operation in Equation (1) is used to consider partial occlusion of instances, so that only the frontmost category label is observable. Note that $c_0 = 0$ is used for the background.

Investigation policy. In the above setting, the number of training images N is often important for improving the pre-training performance, and many previous studies have shown the effectiveness of large-scale datasets of real images [18]. However, the effects of other factors such as the number of instances M_i have been rarely investigated. When pre-training neural networks with synthesized images, what factors improve the performance? To answer this question, we consider the following factors:

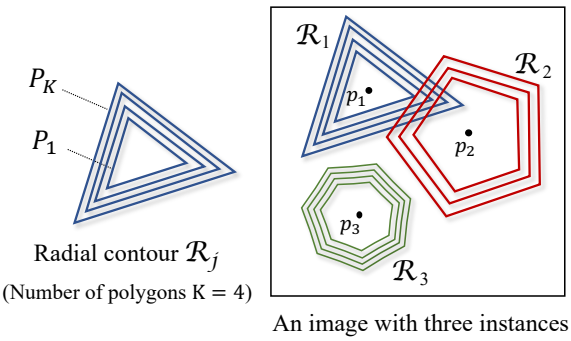


Figure 2. **Radial contours.** An image $\boldsymbol{x}_i$ is composed of multiple radial contours. The radial contours $\mathcal{R}_j$ are placed at positions $\boldsymbol{p}_j$. Each radial contour is an object made by superimposing polygons $\{P_k\}_{k=1}^{K}$.

- **(F1) Number of instances.** How many instances should we have in one image? As the number of instances increases, the segmentation task becomes difficult to solve. We investigate the optimal task difficulty for pre-training in terms of the number of instances M_i.
- **(F2) Mask accuracy.** How accurate should masks be? We investigate the effects of fine-to-coarse masks for $\boldsymbol{m}_i$.
- **(F3) Colors.** Are colors necessary? We investigate whether the color channels of $\boldsymbol{x}_i$ help improve the results.
- **(F4) Occlusion.** Does partial occlusion of some instances boost performance? We investigate how overlap between instances affects pre-training.
- **(F5) Instance shapes.** How complex should the instance shapes be? We investigate the necessary complexity of the boundary shapes for instances $\boldsymbol{o}_j$ as well as the optimal line width and number of polygons in each instance for pre-training.
- **(F6) Number of categories.** Do we need various categories? We investigate the importance of the number of categories C, which corresponds to the number of channels for the masks.
- **(F7) Number of images.** Is increasing the number of images the only way to improve pre-training accuracy? Finally, we compare the importance of increasing the number of images N with the other factors.

4. SegRCDB

Based on the Section 3, we propose SegRCDB, a dataset of synthesized images and masks for pre-training semantic segmentation networks. The dataset is designed to enable us to control the seven factors (F1)-(F7). In the following, we describe how SegRCDB, $\mathcal{D}_{\text{SegRCDB}} = \{(\boldsymbol{x}_i, \boldsymbol{m}_i)\}_{i=1}^{N}$ is constructed.

Instances. To control the complexity of instance shapes, we use *radial contours*, the polygonal objects proposed in [12],

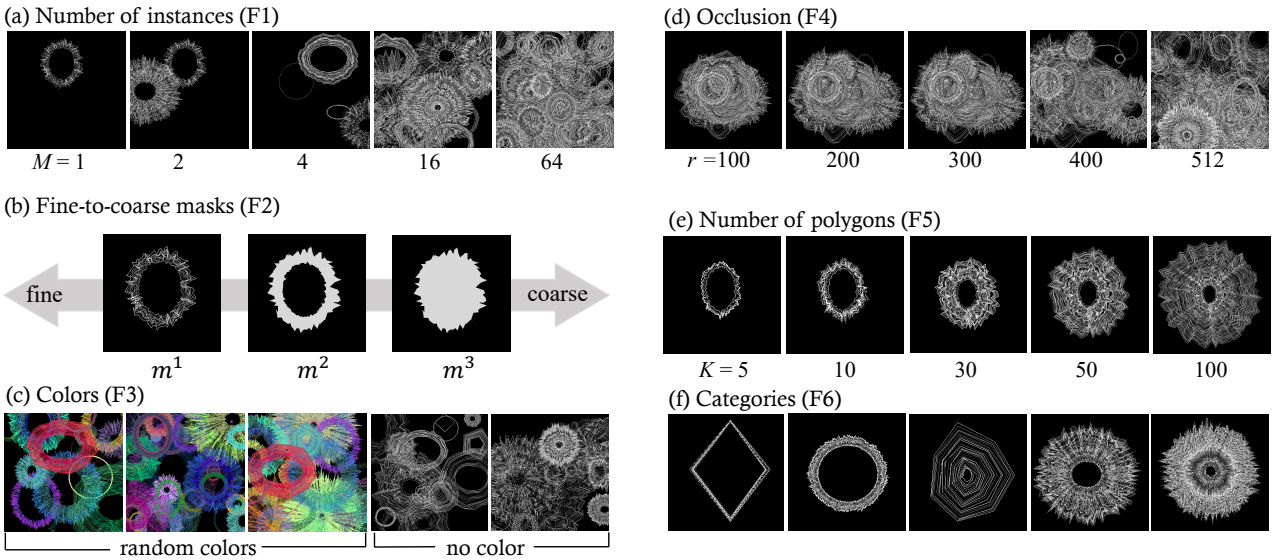


Figure 3. **Examples of images used in the investigation.**

for object instances $\{\boldsymbol{o}_j\}_{j=1}^{M_i}$. More specifically, an instance $\boldsymbol{o}_j$ is given by

$$\boldsymbol{o}_j = (\mathcal{R}_j, \boldsymbol{p}_j, c_j) \tag{2}$$

where $\mathcal{R}_j$ is a radial contour, $\boldsymbol{p}_j$ is the position in the image, and c_j is the category label. Here, a radial contour $\mathcal{R}_j \subset \mathbb{R}^2$ is a 2D object made by superimposing polygons as $\mathcal{R}_j = \cup_{k=1}^{K} P_k$ where P_k is a skeleton polygon (*e.g.*, a triangle) and K is the number of polygons. As shown in Figure 2, an image $\boldsymbol{x}_i$ is composed of M_i radical contours. Please refer to [12] for the detailed definition of each instance R_k.

Number of instances. The number of instances per image M (F1) is treated as a hyper-parameter by assuming $M_i = M$ for all $i = 1, 2, \cdots, N$. In the experiments, M is chosen from $\{1, 2, 4, 8, 16, 32, 64\}$. Some example images are shown in Figure 3a.

Masks. We introduce three types of masks in a fine-to-coarse manner for the investigation of (F2). The first mask $\boldsymbol{m}_j^1$ is the finest mask, which is the mask over skeleton lines obtained by the following two steps. First, binary masks $\boldsymbol{b}_j$ are synthesized by rendering a radial contour $\mathcal{R}_j$ as a binary image, *i.e.*, by rendering white lines on a black background using the same method as that for synthesizing color images. Second, masks are computed using Equation (1). The second mask $\boldsymbol{m}_j^2$ is a ring mask obtained by filling the region between the first polygon R_1 and the final polygon R_K with a value of 1. The third mask $\boldsymbol{m}_j^3$ is the coarsest mask obtained by filling the region inside R_K with the value of 1. Examples of these masks are shown in Figure 3b.

Colors. To render instances $\{\boldsymbol{o}_j\}_j^{M_i}$ into an image $\boldsymbol{x}_i$, we introduce coloring methods. Specifically, we use either of two methods for the investigation of (F3). The first method attaches a random color to each instance $\boldsymbol{o}_j$, where RGB color values are sampled from the uniform distribution over $\{0, 1, \cdots, 255\}$. The second method ignores colors and renders all instances in white. The background color is fixed to black for both methods. Example images are shown in Figure 3c.

Occlusion. Occlusion is introduced for the investigation of (F4). We generate occlusion by changing the position of polygons. The center position of the polygon is adjusted to $\{100, 200, 300, 400, 512\}$ pixels, centered on the image. Some images with different occlusion levels are shown in Figure 3d. The smaller the value of r, the more polygons are concentrated in the center and the more occlusions occur.

Instance shapes. The number of polygons per instance K (F5) is treated as a hyper-parameter. Some images with different numbers of polygons are shown in Figure 3e. The line width d is chosen from $\{1, 2, 3\}$ pixels.

Categories. The category (F6) is determined by the number of vertices, the radius, a resizing factor, and Perlin noise. Please refer to [12] for details. The parameter of the number of categories C is chosen from $\{64, 128, 255, 500\}$. Some example images are shown in Figure 3f.

Number of images. The number of images N (F7) is also treated as a hyper-parameter.

5. Experiments

In this section, we evaluate the pre-training effects on SegRCDB from multiple aspects. Especially, we follow the investigation policy described in Section 3. At the beginning of this section, we present results for factors (F1) to (F7). Based on the investigation, we build SegRCDB with the best parameters for each element and evaluate its performance compared to representative datasets.

Table 1. Baseline parameter set (see Section 3 for detailed parameter descriptions).

Baseline parameter	
Line width (d)	1.0
Number of polygons (K)	{1, 2, 3,...,50}
Occlusion (r)	512
Color	Grayscale
Number of categories (C)	255
Number of images (N)	20k

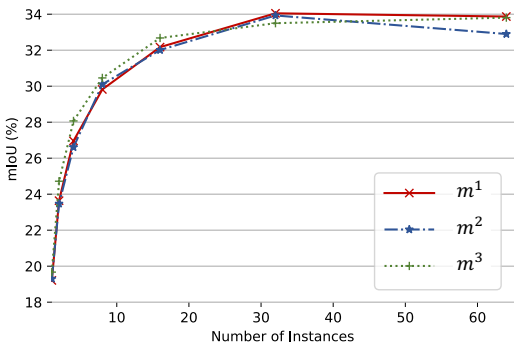


Figure 4. **Number of instances and mask types.** These experiments are related to factors (F1) and (F2).

5.1. Implementation Details

Model. In the segmentation pre-training experiments, we used Swin Transformer base model [19] as a backbone, and UPerNet [35] as the entire architecture. For implementation, we use codes and models in MMSegmentation [6].

Loss function. We follow the official implementation [19]. The backbone part was pre-trained with the AdamW optimizer [21] with a weight decay of 0.01. For the training of UPerNet, we adapted cross-entropy loss.

Learning schedule. The pre-training length is adjusted to 300 epochs by following the FDSL paper [12]. Additionally, the fine-tuning length was 60 epochs. Through the preliminary study, the batch size was set as 32.

Dataset for pre-training and fine-tuning. For comparison, we used ADE-20k [44] as a standard semantic segmentation dataset. This dataset contains 150 categories from daily living environments. The dataset is divided into 20,210 images for training, and 2,000 images for validation. In examining the effect of parameters, baseline parameters were set as shown in Table 1.

5.2. Investigation Results

(F1/F2) Number of instances, and mask accuracy. The experimental results for the number of instances per image and the mask type are both shown in Figure 4. We assigned {2, 4, 8, 16, 32, 64} for the number of instances and {$\boldsymbol{m}^1$, $\boldsymbol{m}^2$, $\boldsymbol{m}^3$} for the mask type. As shown in Figure 4, the more instances for an image, the better the accuracy that can be achieved, up to 32 instances. With regard to the mask type, the most pixel-wise annotation $\boldsymbol{m}^1$ achieves the highest results with 32 instances. Here, pixel-wise annotation is extremely difficult for human annotators in recently reported work [26]. Although humans always find annotation very difficult, our SegRCDB can easily create precise segmentation masks with semantic labels, due to the formula-driven approach using a simple equation. Hereafter, 32 instances per image and a m^1 mask type are used.

(F3) Colors. Table 2 shows the results for color type. We compared the grayscale dataset with a dataset randomly colored by class number. The grayscale dataset achieved a better result, indicating that focusing on object contours produces better results than learning colors for each class. As in a previous paper [12], object contours were found to be more important than color in transformer-based self-attention approaches, including Swin Transformer backbone and UPerNet head.

(F4) Occlusion. Table 3 shows how occlusion affects the experimental results. The parameter r represents the range where the polygon is drawn. After careful investigation, we assigned the parameters as {200, 300, 400, 512} for r. The highest mIoU was achieved for 400 for the fine-tuned dataset. This indicates that an appropriate amount of occlusions can contribute to better results. We can adjust the task difficulty by using the r parameter related to the amount of occlusion inside of the image.

(F5) Instance shapes. We investigated the effect of the complexity of the instance shape, by varying the number of polygons and the line width. We set the number of polygons as {1–25, 26–50, 1–50}, and the line width as {1, 2, 3} pixels. The effects of the number of polygons and the line width are shown in Table 4 and 5, respectively. Fewer polygons and thinner lines produced slightly better results than those of other configurations. Thinner lines lead to fine annotations.

(F6) Number of categories. The experimental results for different numbers of categories in the radial contours are shown in Table 6. We employed {64, 128, 255, 500} in pre-training. Almost all models adopt an 8-bit mask input, so we tried the maximum number 255 (category 256 is used for data augmentation). Using the 16-bit mask image, we also created 500 categories. Based on the results, the 255 class shows the highest results. Hereafter, the number of categories is set at 255.

(F7) Number of images. The pre-training effects of the number of images are shown in Table 7. Fine-tuning is set to 120 epochs to allow for convergence. As the number of images increases, the mIoU value improves. This confirms the intuition that large-scale datasets can be effective in pre-training for semantic segmentation. In Section 5.3, we adopt 118k images for SegRCDB, which has the same number of training images as the COCO-Stuff-

Table 2. (F3) Grayscale vs. color.

Type	Gray	Color
mIoU	**34.05**	28.49

Table 3. (F4) Occlusion.

r	100	200	300	400	512
mIoU	30.62	31.87	33.20	**34.12**	34.05

Table 4. (F5) Number of polygons.

K	1-25	26-50	1-50
mIoU	**34.12**	33.72	34.05

Table 5. (F5) Line width.

d	1px	2px	3px
mIoU	**34.05**	34.03	34.03

Table 6. (F6) Number of categories.

C	64	128	255	500
mIoU	31.67	33.62	**34.05**	31.58

Table 7. (F7) Number of images.

N	20k	40k	80k
mIoU	41.31	41.86	**42.25**

164k for the comparison of pre-training with semantic segmentation datasets.

5.3. Comparison

In this section, we investigate the pre-training effects on SegRCDB, the first semantic segmentation dataset created in the framework of FDSL. To evaluate the effectiveness of various factors, we compared the results of pre-training with semantic segmentation datasets and backbone pre-training with large datasets.

Supervised pre-training for semantic segmentation datasets. We compared the pre-training effects of our SegRCDB with representative semantic segmentation datasets based on supervised learning.

- **COCO-Stuff [4]** contains labeled images from the COCO dataset with pixel-level object annotations. The dataset has 171 categories and 118k training images.
- **Cityscapes [7]** contains labeled images with 19 categories captured from street scenes. The dataset is divided into 2,975 images for the training set, and 500 images for the validation set.
- **GTA5 [27]** contains synthetic street images rendered from the GTA5 video game. The dataset was annotated into 19 categories and contains only a training dataset for pre-training usage.
- **ADE-20k [44]** is described in Section 5.1.

All datasets were pre-trained by Swin Transformer base model [19] for the backbone and UPerNet [35] for the entire architecture. The entire UPerNet, including the backbone and head, is pre-trained on these semantic segmentation datasets and the pre-training effects are compared. We used the AdamW optimizer [21] with a weight decay of 0.1 following the official implementation. All datasets were cropped to produce input images of 512×512 pixels. The pre-training length was 300 epochs, and the fine-tuning length was 150 epochs. The batch size was set to 64 for pre-training, and 16 for fine-tuning. All datasets were fine-tuned with a backbone learning rate of 0.0005, and a weight decay of 0.1.

Table 8 shows comparisons with supervised pre-training methods for fine-tuning for ADE-20k and Cityscapes validation datasets. According to the results, our SegRCDB achieves the highest score. SegRCDB shows superior results to COCO-Stuff dataset even with the same amount of training data. Since SegRCDB selected effective parameters for pre-training, it outperformed the current standard pre-training dataset for semantic segmentation.

Backbone pre-training for semantic segmentation. Here, we compare our SegRCDB with backbone pre-training before semantic segmentation fine-tuning. RCDB-1k and ExFractalDB-1k are the FDSL classification datasets, containing 1.0 million images.

For ImageNet-1k, RCDB-1k, and ExFractalDB-1k training, we followed the official settings of Swin Transformer [19], except for the augmentation process. Augmentation is adjusted to the settings in MMSegmentation for all datasets. The pre-training length is 300 epochs, and the fine-tuning length is the 150 epochs. For the SegRCDB, the experiment settings are the same as in Table 8. SegRCDB is pre-trained on the UPerNet backbone and head, while ImageNet-1k, RCDB-1k, and ExFractalDB-1k datasets are trained on the backbone only.

Table 9 shows experimental results comparing the effects of backbone pre-training with large-scale datasets and SegRCDB's pre-training. For ADE-20k and Cityscapes fine-tuning, ImageNet shows the highest mIoU. Among FDSL methods, SegRCDB achieves the highest scores. SegRCDB contains only 118k images; however, it can surpass other FDSL backbone pre-training methods with 1 million images. This indicates that the FDSL method, which has been considered effective for classification problems so far, has been successfully applied to semantic segmentation in SegRCDB.

5.4. Explorative Study

Additional experiments were conducted to further investigate the performance of SegRCDB.

mIoU transition during fine-tuning. Models trained by formula-driven supervised learning might require a longer fine-tuning epoch to acquire a real-image representation. Since previous studies of FDSL have employed 1k epoch

Table 8. Comparison of pre-training with semantic segmentation datasets. The best and second-best values for each fine-tuning dataset are in underlined bold and bold, respectively.

Pre-training	#Img	ADE-20k mIoU	ADE-20k mAcc	Cityscapes mIoU	Cityscapes mAcc
Scratch	-	31.40	41.02	54.65	62.89
ADE-20k	20k	-	-	68.46	77.13
GTA5	25k	39.31	49.79	71.00	79.31
COCO-Stuff	118k	**43.39**	**54.41**	**72.21**	**80.62**
SegRCDB	118k	**43.85**	**54.98**	**73.06**	**81.59**

Table 9. Comparison with backbone pre-training. The best and second-best values for each fine-tuning dataset are in underlined bold and bold, respectively.

Pre-training	#Img	ADE-20k mIoU	ADE-20k mAcc	Cityscapes mIoU	Cityscapes mAcc
Scratch	-	31.40	41.02	54.65	62.89
ImageNet	1.28M	**46.37**	**57.11**	**75.26**	**83.60**
ExFractalDB	1M	40.96	52.13	68.93	77.96
RCDB	1M	41.07	51.89	69.66	78.35
SegRCDB	118k	**43.85**	**54.98**	**73.06**	**81.59**

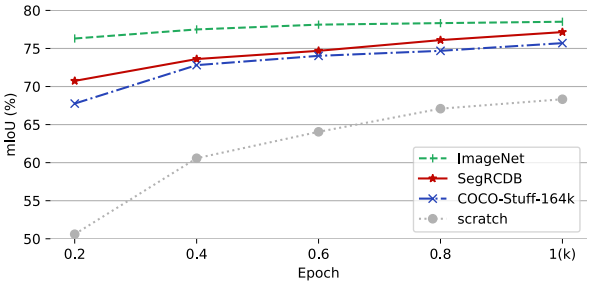


Figure 5. **mIoU transition during fine-tuning on Cityscapes.**

fine-tuning [12, 23], we also conduct 1k epoch fine-tuning using SegRCDB of 118k images. The results of 1k epoch fine-tuning on Cityscapes are shown in Figure 5. This shows that the pre-trained model with SegRCDB shows higher mIoU than from scratch and COCO-Stuff from the early stages of fine-tuning. The gap in mIoU between SegRCDB and ImageNet is wider in the early stages of fine-tuning but becomes narrower as the learning progresses.

Large-scale SegRCDB. Compared to one instance per image in the conventional FDSL method, SegRCDB has 32 instances per image, which increases its pre-training effect. We prepared a 1M scale SegRCDB to verify whether learning with a larger number of images would further enhance the pre-training effect. Table 10 shows the fine-tuning results for two different numbers of images of SegRCDB. SegRCDB with 1M images is trained for 150 epochs. Other parameter settings are the same as Table 8 and Table 9. This indicates that fine-tuning on ADE-20k is more effective when the number of images is increased, while fine-tuning on Cityscapes is not. It was also reported in previous studies that increasing the number of images does not improve the results for some fine-tuning datasets using the FDSL method [12, 23].

CNN backbone architecture. We also investigated the performance of SegRCDB with different backbone networks. We used the ResNet-101 [10] convolutional backbone, as implemented in MMSegmentation [6]. We used the same backbone model for both the pre-training and fine-tuning. Pre-training learning schedule was set to 300 epochs, while the fine-tuning was set to 150 epochs. We use 20k images of SegRCDB for training. Table 11 compares the backbone pre-training results for fine-tuning on ADE-20k and Cityscapes, where Swin-B shows higher results than those of the ResNet-101 model. As shown in previous work [12], a radial contour shape is more effective for transformer architectures than for CNN architectures.

Impact of annotation accuracy. In the semantic segmentation domain, it is often challenging to guarantee consistency between manually annotated ground truth masks. In this work, the FDSL approach ensured precisely labeled semantic annotations, thereby removing ambiguities and mistakes from annotations. In the dataset analysis of ADE-20k [44], it is reported that on average only 82.4% of pixels have the same annotation when re-annotated by the same annotator. Therefore, we deliberately shifted and inflated the pixel annotations on SegRCDB for pre-training. We varied the shift and inflation parameters in the range $\{10, 30, 100, 300\}$ to emulate the statistics from the ADE-20k paper [44]. We conducted two experiments related to annotations: (a) shift, where we added varying degrees of noise to the object vertexes, and (b) inflation, which was obtained by enlarging the area enclosed by polygons.

Table 12 shows the relationship between annotation precision and semantic segmentation performance in terms of mIoU. We used 20k training images for each trial. The learning schedule was set to 300 epochs for the pre-training stage, and to 150 epochs for fine-tuning on Cityscapes. Our experiments show that shift annotation has a bigger impact on performance compared to inflation, lowering the mIoU score by 6.54 points when switching from 0 to 100 pixels' shift ($70.23 - 63.69$). In contrast, the mislabeled annotation caused by inflation appears less critical to the mIoU performance, with a decrease of 2.93 points up to 100 pixels. For a 300 pixel shift and inflation, the mIoU drops more significantly.

5.5. Discussion and Limitations

We here summarize the findings obtained in the present study. According to the investigation described in Section 5.2, we confirmed that five factors are more important than other factors for semantic segmentation pre-training:

Table 10. The comparison for the number of images.

	ADE-20k		Cityscapes	
#Img	mIoU	mAcc	mIoU	mAcc
118k	43.85	54.98	73.06	81.59
1M	44.46	55.67	72.10	81.08

Table 11. Comparison of backbone architecture.

	ADE-20k		Cityscapes	
Backbone	mIoU	mAcc	mIoU	mAcc
ResNet-101	39.56	51.48	66.74	75.31
Swin-B	41.51	52.58	70.23	78.78

(F1) number of instances, (F3) grayscale, (F4) occlusion, (F6) number of categories, and (F7) number of images.

Number of instances (see also Figure 4). The more instances for an image, the better accuracy is until saturation occurs at 32 instances. Increasing the number of instances in one image made the task more difficult, which was effective for pre-training.

Grayscale representation (see also Table 2). In relation to (F3), we confirmed that a grayscale representation is much better than a color image in object areas. The performance gap between grayscale and color was at 5.56 for ADE-20k. This suggests that the object categories should not be distinguished by color alone in a pre-training task. The color of objects may be an easy pre-training task, and cause the pre-trained model to be weak. Conversely, grayscale objects must be classified by means of object shapes using radial contours. This is inherited from the previous work [12], and is also good for semantic segmentation.

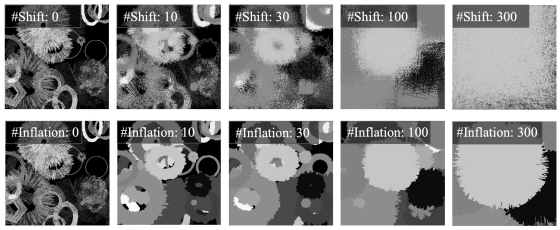
Occlusion (see also Table 3). From the results, occlusions among objects is important. Heavy occlusions can be solved in a pre-training task; however, for use as a hyper-parameter, there is an optimal occlusion amount. We confirmed that a value of 400 is the most effective, and saturation occurs at 400 because the value of 500 produces almost the same level of mIoU on the ADE-20k dataset. The performance rate was increased by 3.5 points from minimal occlusions to more occlusion areas for these experimental settings.

Number of categories and images (see also Tables 6 and 7). In the investigations regarding factors (F6) and (F7), the numbers of categories and images are related to the pre-training effect. We confirmed that 255 categories achieved the best result in the pre-training phase. Although more images tend to be better in Table 7, a huge 1M dataset does not necessarily lead to better results in Section 5.4.

Other investigations (see also Figure 4, Tables 4, and Tables 5). For investigations (F2) and (F5), these parameters achieved slightly higher mIoU, but had no significant effect on ADE-20k fine-tuning. A recent study [26] claimed that a highly accurate annotation like segmentation in m^1 is important, but it had slightly better accuracy than other configurations with a higher number of instances per image in SegRCDB pre-training.

Table 12. Annotation analysis on shift and inflation. The values are given in pixels.

Pixels	0	10	30	100	300
Shift	70.23	70.02	70.08	63.69	16.04
Inflation	70.23	70.86	69.18	67.30	15.73



Comparison experiments. Undoubtedly, the proposed SegRCDB pre-training model is more accurate than self-supervised learning with synthetic datasets such as GTA5, RCDB, and ExFractalDB for fine-tuning in indoor scenes (ADE-20k) and urban scenes (Cityscapes). Moreover, the SegRCDB pre-trained Swin Transformer performed equally well or even better than a sophisticated supervised learning method with semantic segmentation datasets. Actually, the proposed method exhibited better performance compared with that of COCO-Stuff pre-training. Among the FDSL methods, SegRCDB largely achieved better accuracy than those of ExFractalDB and RCDB. The FDSL method, specialized for classification problems, has been successfully applied to semantic segmentation tasks.

Limitations. We believe that there are additional factors other than (F1)–(F7) that have not been considered for improving segmentation pre-training. It is also possible to examine the relationship between factors in detail. Finally, the shapes in our SegRCDB were taken "as is" from a previous study [12]. Therefore, there is a room to further improve our SegRCDB for semantic segmentation pre-training.

6. Conclusion

We proposed SegRCDB (Segmentation Radial Contour DataBase), the first segmentation dataset developed by formula-driven supervised learning (FDSL). By investigating effective parameters for pre-training, SegRCDB outperformed COCO-Stuff-164k pre-training model with the same number of training data.

SegRCDB can reveal what is effective in pre-training for semantic segmentation by varying the dataset's parameters. We hope that our SegRCDB will promote research on semantic segmentation pre-training without any manual effort and real images.

7. Acknowledgement

This paper is based on results obtained from a project, JPNP20006, commissioned by the New Energy and Industrial Technology Development Organization (NEDO). Computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST) was used.

References

[1] Connor Anderson and Ryan Farrell. Improving fractal pre-training. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1300–1309, 2022. 2

[2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 39, pages 2481–2495, 2017. 2

[3] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani. Semantic segmentation with boundary neural fields. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3602–3610, 2016. 2

[4] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1209–1218, 2018. 2, 6

[5] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *European Conference on Computer Vision (ECCV)*, pages 833–851, Cham, 2018. Springer International Publishing. 2

[6] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020. 5, 7

[7] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016. 1, 6

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009. 1

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017. 2

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 7

[11] Jitesh Jain, Anukriti Singh, Nikita Orlov, Zilong Huang, Jiachen Li, Steven Walton, and Humphrey Shi. Semask: Semantically masking transformer backbones for effective semantic segmentation. *arXiv*, 2021. 2

[12] Hirokatsu Kataoka, Ryo Hayamizu, Ryosuke Yamada, Kodai Nakashima, Sora Takashima, Xinyu Zhang, Edgar Josafat Martinez-Noriega, Nakamasa Inoue, and Rio Yokota. Replacing labeled real-image datasets with auto-generated contours. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21232–21241, June 2022. 2, 3, 4, 5, 7, 8

[13] Hirokatsu Kataoka, Asato Matsumoto, Ryosuke Yamada, Yutaka Satoh, Eisuke Yamagata, and Nakamasa Inoue. Formula-driven supervised learning with recursive tiling patterns. In *IEEE/CVF International Conference on Computer Vision (CVPR)*, pages 4098–4105, 2021. 2

[14] Hirokatsu Kataoka, Kazushige Okayasu, Asato Matsumoto, Eisuke Yamagata, Ryosuke Yamada, Nakamasa Inoue, Akio Nakamura, and Yutaka Satoh. Pre-training without natural images. *International Journal of Computer Vision (IJCV)*, 130(2):990–1007, 2022. 2

[15] Hirokatsu Kataoka, Eisuke Yamagata, Kensho Hara, Ryusuke Hayashi, and Nakamasa Inoue. Spatiotemporal initialization for 3d cnns with generated motion patterns. In *Winter Conference on Applications of Computer Vision (WACV)*, pages 1279–1288, 2022. 2

[16] Daiqing Li, Junlin Yang, Karsten Kreis, Antonio Torralba, and Sanja Fidler. Semantic segmentation with generative models: Semi-supervised learning and strong out-of-domain generalization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8296–8307, 2021. 2

[17] Yunsheng Li, Lu Yuan, and Nuno Vasconcelos. Bidirectional learning for domain adaptation of semantic segmentation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6929–6938, 2019. 2

[18] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pages 740–755, 2014. 1, 3

[19] 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 *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9992–10002, 2021. 2, 5, 6

[20] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2015. 2

[21] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019. 5, 6

[22] John McCormac, Ankur Handa, Stefan Leutenegger, and Andrew J. Davison. Scenenet rgb-d: Can 5m synthetic images beat generic imagenet pre-training on indoor segmentation? In *IEEE International Conference on Computer Vision (ICCV)*, pages 2697–2706, 2017. 1

[23] Kodai Nakashima, Hirokatsu Kataoka, Asato Matsumoto, Kenji Iwata, Nakamasa Inoue, and Yutaka Satoh. Can vision transformers learn without natural images? In *AAAI Confer-*

ence on Artificial Intelligence, volume 36, pages 1990–1998, 2022. 2, 7

[24] Vinay Uday Prabhu and Abeba Birhane. Large image datasets: A pyrrhic win for computer vision? *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1536–1546, 2021. 1, 2

[25] Yu Qiao, Jincheng Zhu, Chengjiang Long, Zeyao Zhang, Yuxin Wang, Zhenjun Du, and Xin Yang. Cpral: Collaborative panoptic-regional active learning for semantic segmentation. *AAAI Conference on Artificial Intelligence*, 36(2):2108–2116, Jun. 2022. 2

[26] Xuebin Qin, Hang Dai, Xiaobin Hu, Deng-Ping Fan, Ling Shao, and Luc Van Gool. Highly accurate dichotomous image segmentation. In *European Conference on Computer Vision (ECCV)*, 2022. 5, 8

[27] Stephan R. Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *European Conference on Computer Vision (ECCV)*, pages 102–118, 2016. 1, 2, 6

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, Cham, 2015. Springer International Publishing. 2

[29] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3234–3243, June 2016. 1, 2

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

[31] Y. Siddiqui, J. Valentin, and M. Niessner. Viewal: Active learning with viewpoint entropy for semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9430–9440, Los Alamitos, CA, USA, jun 2020. IEEE Computer Society. 2

[32] Towaki Takikawa, David Acuna, Varun Jampani, and Sanja Fidler. Gated-scnn: Gated shape cnns for semantic segmentation. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 5228–5237, 2019. 2

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30:2980–2988, 2017. 2

[34] Tianlu Wang, Jieyu Zhao, Mark Yatskar, Kai-Wei Chang, and Vicente Ordonez. Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations. In *IEEE International Conference on Computer Vision (ICCV)*, pages 5309–5318, October 2019. 1, 2

[35] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *European Conference on Computer Vision (ECCV)*. Springer, 2018. 5, 6

[36] Binhui Xie, Longhui Yuan, Shuang Li, Chi Harold Liu, and Xinjing Cheng. Towards fewer annotations: Active learning via region impurity and prediction uncertainty for domain adaptive semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8068–8078, June 2022. 2

[37] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *Neural Information Processing Systems (NeurIPS)*, 2021. 2

[38] Ryosuke Yamada, Hirokatsu Kataoka, Naoya Chiba, Yukiyasu Domae, and Tetsuya Ogata. Point cloud pre-training with natural 3d structures. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21283–21293, 2022. 2

[39] Ryosuke Yamada, Ryo Takahashi, Ryota Suzuki, Akio Nakamura, Yusuke Yoshiyasu, Ryusuke Sagawa, and Hirokatsu Kataoka. Mv-fractaldb: Formula-driven supervised learning for multi-view image recognition. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2076–2083, 2021. 2

[40] Kaiyu Yang, Klint Qinami, Li Fei-Fei, Jia Deng, and Olga Russakovsky. Towards fairer datasets. In *Conference on Fairness, Accountability, and Transparency*, pages 547–558, jan 2020. 1, 2

[41] Kaiyu Yang, Jacqueline Yau, Li Fei-Fei, Jia Deng, and Olga Russakovsky. A study of face obfuscation in imagenet. In *International Conference on Machine Learning (ICML)*, 2022. 1

[42] Dora Zhao, Angelina Wang, and Olga Russakovsky. Understanding and evaluating racial biases in image captioning. In *International Conference on Computer Vision (ICCV)*, 2021. 1, 2

[43] Zhedong Zheng and Yi Yang. Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation. *International Journal of Computer Vision (IJCV)*, 2021. doi:10.1007/s11263-020-01395-y. 2

[44] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 5, 6, 7

[45] Yang Zou, Zhiding Yu, BVK Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *European Conference on Computer Vision (ECCV)*, pages 289–305, 2018. 2

[46] Yang Zou, Zhiding Yu, Xiaofeng Liu, B.V.K. Vijaya Kumar, and Jinsong Wang. Confidence regularized self-training. In *IEEE International Conference on Computer Vision (ICCV)*, pages 5981–5990, October 2019. 2