SegGen: Supercharging Segmentation Models with Text2Mask and Mask2Img Synthesis

Hanrong Ye[1], Jason Kuen[2], Qing Liu[2], Zhe Lin[2], Brian Price[2], Dan Xu[1]

[1] CSE, HKUST
[2] Adobe Research

Abstract. We present SegGen, a new data generation approach that pushes the performance boundaries of state-of-the-art image segmentation models. One major bottleneck of previous data synthesis methods for segmentation is the design of "segmentation labeler module", which is used to synthesize segmentation masks for images [48]. The segmentation labeler modules, which are segmentation models by themselves, bound the performance of downstream segmentation models trained on the synthetic masks. These methods encounter a "chicken or egg dilemma" and thus fail to outperform existing segmentation models. To address this issue, we propose a novel method that ***reverses*** the traditional data generation process: we first (i) generate highly diverse segmentation masks that match real-world distribution from text prompts, and then (ii) synthesize realistic images conditioned on the segmentation masks. In this way, we avoid the need for any segmentation labeler module. SegGen integrates two data generation strategies, namely MaskSyn and ImgSyn, to largely improve data diversity in synthetic masks and images. Notably, the high quality of our synthetic data enables our method to outperform the previous data synthesis method [48] by +25.2 mIoU on ADE20K when trained with pure synthetic data. On the highly competitive ADE20K and COCO benchmarks, our data generation method markedly improves the performance of state-of-the-art segmentation models in semantic segmentation, panoptic segmentation, and instance segmentation. Moreover, experiments show that training with our synthetic data makes the segmentation models more robust towards unseen data domains, including real-world and AI-generated images.

Keywords: Image Segmentation · Image Generation · Deep Learning

1 Introduction

Image segmentation explores the identification of objects in visual inputs at the pixel level. Based on the different emphases on category and instance membership information, researchers have divided image segmentation into several tasks [5,25,35,40]. For example, semantic segmentation studies pixel-level understanding of object categories, instance segmentation focuses on instance grouping

Fig. 1: Effectiveness of SegGen on various data domains: Through training with synthetic data generated by the proposed SegGen, we significantly boost the performance of state-of-the-art segmentation model Mask2Former [7] on evaluation benchmarks including ADE20K [60] and COCO [33], whilst making it more robust towards challenging images from other domains (the three columns on the left are from PASCAL [10]; the three on the right are synthesized by image generation model Kandinsky 2 [12]). SegGen outperforms the previous best data generation method (DiffuMask [48]) by a huge margin when models are trained on pure synthetic data. "mIoU*" is the average IoU metric defined by [48] which focuses on three common classes.

of pixels, while panoptic segmentation considers both. For all these segmentation tasks, obtaining high-quality annotation is challenging as every individual pixel requires human labeling, and a single image can contain millions of pixels. Therefore, compared to other public datasets like ImageNet-21K (with around 14M images), the prevailing human-annotated segmentation datasets are notably smaller. For example, ADE20K dataset [61] contains about 20K images in its training split, while COCO [33] has around 118K training images. Although there has been significant development in the structure of segmentation models [8, 29, 52, 55, 57], the limited size of training data hinders further performance enhancements and results in inadequate generalization ability to handle images from unfamiliar domains, such as those from other scenes or synthesized by generative models as shown in the second row of Fig. 1.

Inspired by the recent success of image generation [9, 23], researchers start to explore using generative models to enhance image segmentation [1]. A representative direction of this research focuses on synthesizing segmentation training data in a cost-effective manner [26]. In related methods, the synthetic segmentation masks are obtained from some manually designed "segmentation labeler modules". The segmentation labeler modules are essentially segmentation models by themselves. They are either small-scale segmentation networks as in DatasetGAN [56] and Grounded Diffusion [32], or post-processing methods on the image features as in DiffuMask [48]. The performance of the downstream segmentation models is constrained by the quality of the synthetic masks they are trained on, which in turn relies on the capabilities of those segmentation labeler modules. This is a "chicken or egg dilemma" and the performance bottleneck is the segmentation labeler module. Therefore, while they achieve encouraging success in settings with limited training data, their methods fail to deliver notable en-

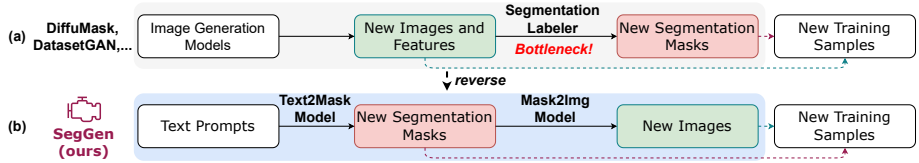

Fig. 2: Comparison with previous data generation methods for segmentation: (a) Earlier methods [48, 56] rely on segmentation labeler modules to produce segmentation masks for synthetic images. However, the performance of these downstream segmentation models, trained on synthetic data, is bounded by the capacity of the segmentation labeler modules. The segmentation labeler is a major bottleneck for the quality of the generated data. **(b)** We design a reverse pipeline: we first create diverse new masks from text prompts via a proposed Text2Mask generation model and then synthesize images conditioned on the segmentation masks. This methodology avoids any usage of segmentation labeler networks, resulting in significantly improved quality of our synthesis data.

hancements for the state-of-the-art models on the most popular segmentation benchmarks in settings using the complete training sets.

On the other hand, a handful of works [54, 58] have demonstrated that high-quality synthetic images can be generated by conditioning text-to-image models on dense input maps (*e.g.,* segmentation masks or canny edge). Moreover, these dense-input conditional models provide a strong alignment between the dense input label maps and generated images. These two qualities motivate us to leverage the powerful capabilities of such models for effective segmentation data generation. To this end, we propose a novel segmentation data generation method, coined as **SegGen**, for generating high-quality segmentation training data. We reverse the traditional data generation pipeline for segmentation: we first synthesize high-quality segmentation masks from a proposed text-to-mask generation model, and then generate images based on the segmentation masks using a mask-to-image generation model. In this new framework, we avoid using any segmentation labeler modules, which results in significantly improved data quality. Fig. 2 shows the difference between our method and previous data generation methods for segmentation [48, 56].

As each training sample of image segmentation consists of two components: segmentation masks and the corresponding image, we develop two novel data generation approaches, emphasizing improvements in two distinct aspects of data diversity: (i) segmentation masks and (ii) images. The first data generation approach, named **MaskSyn**, centers around the generation of new segmentation masks. It learns a text-to-mask (Text2Mask) generation model to produce completely new segmentation masks given text prompts. Then, it learns a mask-to-image (Mask2Img) generation model to synthesize images that align with the synthetic segmentation masks. The second data generation approach, named **ImgSyn**, utilizes the above-mentioned Mask2Img model to synthesize new images given human-annotated segmentation masks. With MaskSyn and ImgSyn, we can readily generate a vast array of diverse and high-quality synthetic training data. The combined synthetic data is used to train segmenta-

tion models in conjunction with real training samples from human-annotated datasets. Experiments show that our synthetic data can significantly boost the performance of the image segmentation models on challenging benchmarks including ADE20K semantic segmentation, COCO panoptic segmentation, and COCO instance segmentation, achieving new state-of-the-art performances without using extra human-annotated data. Notably, SegGen remarkably outperforms the previous segmentation data generation method [48] by **+25.2** mIoU when trained with pure synthetic data on ADE20K. Our method boosts the mIoU of Mask2Former by +2.7 (R50) and +1.3 (Swin-L) on the ADE20K semantic segmentation benchmark. Furthermore, segmentation models trained on our synthetic data exhibit a remarkably stronger ability to generalize across unfamiliar image domains, including real images from other distributions and machine-generated images.

We summarize the contribution of this work in three points: **(i)** We propose a revolutionary generation framework that reverses the traditional segmentation data generation pipeline and solves the "chicken or egg dilemma". The new framework enables us to produce high-quality segmentation training data at scale, thus enabling the training of more powerful image segmentation models. **(ii)** We introduce two effective generative models, one for text-to-mask generation and the other for mask-to-image generation. Based on these models, we propose two novel segmentation training data generation approaches, namely MaskSyn and ImgSyn. They significantly improve the data diversity, with MaskSyn focusing on new segmentation masks and ImgSyn on new images. **(iii)** SegGen successfully improves the performance of the leading-edge segmentation models across the highly competitive benchmarks on ADE20K and COCO. Moreover, SegGen enhances the generalization ability of segmentation models towards unseen image domains. Rigorous experiments, including ablation study and peer comparison, strongly suggest the effectiveness of the proposed method.

2 Related Work

Generation for Segmentation Image segmentation is one of the most studied visual perception problems [30]. In recent years, there has been a surge in efforts to harness the capabilities of generative models for segmentation tasks. These efforts can be broadly classified into three categories based on their methodologies: (i) Extracting visual features from generative models for segmentation [1, 38, 50, 59]. These methods harvest the strong representation ability of diffusion models trained on large-scale datasets but are limited in the precision of predicted masks. (ii) Formulating segmentation tasks directly as generative models [6,20,45]. Although these methods propose exciting new model architectures, they usually consume higher computational costs while showing unimproved performance compared with conventional segmentation models. (iii) Synthesizing segmentation training data using generative models [26, 32, 37, 47–49, 56]. While these methods have showcased commendable results against their respective baselines especially when the training data is highly limited, they have

yet to exhibit notably superior performance on the most rigorous benchmarks including ADE20K (150 categories) [61] and COCO (133 categories) [33] under fully-supervised setting. A primary concern with these techniques is the subpar quality of the generated segmentation masks. This stems from their dependence on the bottleneck segmentation labeler modules during the synthetic mask generation process. Therefore, a concurrent work [51] abandons the generation of segmentation masks and directly generates images based on existing masks, which is highly limited in mask diversity. In contrast, our data generation workflow reverses the traditional data generation pipeline: we first generate segmentation masks from text, and then synthesize images based on segmentation masks. Our method avoids the need for any segmentation labeler which results in largely improved data quality. Our SegGen can strongly enhance the existing state-of-the-art segmentation models on different challenging benchmarks.

Conditional Image Synthesis Within the realm of conditional image synthesis, Generative Adversarial Networks [2, 13, 21, 23, 43], Variational Autoencoders [24], and Diffusion Models [9, 17, 42, 44] have been at the forefront. Recently, the open-source research community has shown a burgeoning interest in the latent diffusion-based Stable Diffusion (SD) series for text-to-image synthesis [41]. The most recent iteration, SDXL, introduced by [39], expands the model capacity, yielding significantly enhanced results. Therefore, we build SegGen upon SDXL. Regarding mask-to-image generation models [18, 31, 53], ControlNet [54] and T2I-Adapter [36] suggest freezing the parameters of the SD model and introducing a set of more compact, learnable modules. This approach enables image generation conditioned on the given segmentation masks. We adopt the structure of ControlNet in our mask-to-image generation model. To the best of our knowledge, there have been no prior endeavors on text-to-mask generation.

3 Method

3.1 Overview

SegGen is designed to synthesize high-quality training samples for improving the performance of segmentation models. The overall workflow is shown on the left of Fig. 3. We first train SegGen with human-annotated training samples from public datasets. After training, we use SegGen to produce new segmentation training samples at scale. The generated training samples are incorporated into the training process of segmentation models to enhance the model performance.

3.2 Models in SegGen

As shown in Fig. 3, we first utilize a captioner model to extract captions of the real training images as text prompts from the target dataset. The text prompts will condition the data generation process. Then, two conditional generative models are introduced: a text-to-mask (Text2Mask) generation model and a mask-to-image (Mask2Img) generation model. Both generative models are built upon the SDXL model [39] which provides top-notch image generation quality.

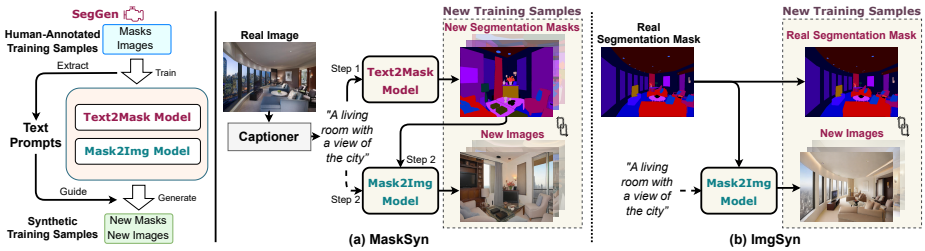

Fig. 3: Illustration of the workflow of our proposed SegGen: We introduce two generative models: a text-to-mask (Text2Mask) generation model and a mask-to-image (Mask2Img) generation model, based on which we design two approaches for synthesizing segmentation training samples: MaskSyn and ImgSyn. **(a)** MaskSyn focuses on generating new segmentation masks. It first extracts the caption of the real image as a text prompt and uses it to generate new masks with the Text2Mask model. Then, the new masks and text prompt are fed into the Mask2Img model to produce the corresponding new images. **(b)** ImgSyn focuses on the synthesis of new images. It inputs human-labeled masks and text prompts into the Mask2Img model.

Captioner Model To obtain image captions of existing training samples, we employ BLIP2-FlanT5$_{\text{xxl}}$ [28] model, which is a state-of-the-art vision-language model. We feed the text prompt "`Question: What are shown in the photo? Answer:`" and image as input to the model. The responses serve as text prompts to condition the following generation process.

Text2Mask Model We design a Text2Mask model in SegGen for generating diverse segmentation masks based on given text prompts. To leverage the generation capacity of text-to-image generation models pre-trained on large-scale datasets, we encode the segmentation masks (the pixel values are category IDs) as three-channel RGB-like color maps, where one color represents a certain category. From our experiments, the color map reconstructed by VAE [41] in SDXL appears almost indistinguishable from the original input as shown in the supplementary materials. Therefore, we can directly fine-tune the text-to-image SDXL-base model with [text, segmentation color map] training pairs, which are from the public image segmentation dataset (*e.g.* ADE20K). During sampling, our Text2Mask model can generate diverse color maps conditioned on text prompts, which are subsequently converted into segmentation masks. Formally, suppose the input text prompt is $\mathbf{T}$, the target height and width are H and W, the synthesized color map is $\mathbf{C}_{\text{syn}} \in \mathbb{W}^{H\times W\times 3}$, and the synthesized segmentation map (with N masks) is $\mathbf{M}_{\text{syn}} \in \mathbb{W}^{H\times W\times N}$, the generation process is:

$$\begin{aligned} \mathbf{C}_{\text{syn}} &= \text{Text2Mask}(\mathbf{T}), \\ \mathbf{M}_{\text{syn}} &= f_{\text{color}\to\text{mask}}(\mathbf{C}_{\text{syn}}), \end{aligned} \tag{1}$$

where $f_{\text{color}\to\text{mask}} : \mathbb{W}^{H\times W\times 3} \to \mathbb{W}^{H\times W\times N}$ is the function that projects the color maps to segmentation masks. Specifically, for each pixel on the color maps, we identify its nearest color (in Euclidean space) in the aforementioned lookup table, and assign the corresponding class to the pixel in the segmentation masks.

Fig. 4: Generated samples by MaskSyn on ADE20K: The third row overlays the mask and the image together to demonstrate the alignment between them. The generated segmentation masks and images demonstrate high perceptual quality and excellent alignment (see more samples in supplementary materials).

Mask2Img Model The goal of the Mask2Img model is to synthesize new images that align well with the given segmentation masks and text prompts. Specifically we adopt the structure of ControlNet [54]: we freeze the pre-trained weights of the SDXL-base model and train an additional side network for mask-conditioned image generation. It simultaneously keeps the generalization ability of the pre-trained diffusion model and provides excellent controllable generation ability. The Mask2Img model is trained with the [text, segmentation color map, image] triplets gathered from the training splits of the target datasets. We denote the input segmentation map as $\mathbf{M} \in \mathbb{W}^{H\times W\times N}$, the color map as $\mathbf{C} \in \mathbb{W}^{H\times W\times 3}$, the synthetic image as $\mathbf{I}_{\text{syn}} \in \mathbb{W}^{H\times W\times 3}$, and the generation process is:

$$\begin{aligned} \mathbf{C} &= f_{\text{mask}\to\text{color}}(\mathbf{M}), \\ \mathbf{I}_{\text{syn}} &= \text{Mask2Img}(\mathbf{T}, \mathbf{C}), \end{aligned} \tag{2}$$

where $f_{\text{mask}\to\text{color}} : \mathbb{W}^{H\times W\times N} \to \mathbb{W}^{H\times W\times 3}$ is the function to convert the segmentation masks into a color map. For semantic segmentation, the value of each pixel on the segmentation mask corresponds to a category ID, allowing us to convert the masks directly into an RGB color map using a pre-defined lookup table. For panoptic and instance segmentation, after mapping the category IDs to color maps, it is essential to outline each segment with a special edge color on the color map. This ensures the model recognizes the specific instance it belongs to. The segmentation map $\mathbf{M}$ can be human-annotated or synthetic (*i.e.*, $\mathbf{M}_{\text{syn}}$ from Text2Mask Model as shown in Eq. 1).

3.3 Data Generation

With the aforementioned generative models, SegGen proposes two approaches for synthesizing new segmentation training samples: MaskSyn and ImgSyn. MaskSyn focuses on enhancing the diversity of synthetic segmentation masks, whereas ImgSyn concentrates on diversifying synthetic images. These generation approaches are illustrated on the right of Fig. 3.

Fig. 5: Generated samples by ImgSyn on ADE20K: The generated images exhibit remarkable realism and align well with the human-labeled masks and text prompts (see more samples in supplementary materials).

Fig. 6: Zoom-in comparison of real images and our ImgSyn images: As highlighted in the circles, our synthetic images align better with the human-labeled masks in many cases because of the inaccuracies in annotations. The left 4 columns are from ADE20K and the right 4 from COCO.

MaskSyn MaskSyn starts with a real training sample pair [image, segmentation masks] from a human-annotated segmentation dataset. It first extracts the caption of the image with the image captioner model. The caption serves as a text prompt and is used to generate a set of diverse new segmentation masks with the Text2Mask model following Eq. 1. Subsequently, the new segmentation masks and the corresponding text prompt are fed into the Mask2Img model to generate a new image that aligns well with its mask. As such, each training sample crafted by MaskSyn includes both a novel segmentation mask and a new image. MaskSyn effectively increases the data diversity in segmentation masks to a great extent. Some generated samples are shown in Fig. 4.

ImgSyn Different from MaskSyn, ImgSyn focuses on increasing the data diversity of images based on human-annotated segmentation masks. For each real training sample pair [image, segmentation mask], it takes the human-annotated segmentation mask and the text prompt extracted from the image as input, and generates a set of varied images that align well with the human-annotated mask. In this way, new training samples consisting of human-labeled masks and new synthetic images are generated. ImgSyn can also be viewed as a kind of data augmentation method that enhances the data diversity on the image side. Our experiments reveal a remarkably high alignment between the synthetic images and their respective segmentation masks. Some synthesized results are showcased in Fig. 5. Our machine-generated synthetic images achieve a better mask-image alignment than real images in many cases, as shown in Fig. 6. This phenomenon occurs because human annotations tend to be imperfect due to the high difficulty of annotating segmentation masks.

Method	Venue	Backbone	Crop Size	Iterations	mIoU (s.s.)	mIoU (m.s.)
MaskFormer [8]	NeurIPS 2021	R50	512	160k	44.5	46.7
Mask DINO [27]	CVPR 2023	R50	512	160k	48.7	-
OneFormer [19]	CVPR 2023	R50	512	160k	47.3	-
Mask2Former [7]	CVPR 2022	R50	512	160k	47.2	49.2
—w/ **SegGen (ours)**	-	R50	512	160k	**49.9** (+2.7)	**51.4** (+2.2)
MaskFormer [8]	NeurIPS 2021	Swin-L	640	160k	54.1	55.6
Mask DINO [27]	CVPR 2023	Swin-L	640	160k	56.6	-
OneFormer [19]	CVPR 2023	Swin-L	640	160k	57.0	57.7
Mask2Former [7]	CVPR 2022	Swin-L	640	160k	56.1	57.3
—w/ **SegGen (ours)**	-	Swin-L	640	160k	**57.4** (+1.3)	**58.7** (+1.4)

Table 1: Semantic segmentation on ADE20K `val` (150 categories): Synthetic data is used as a data augmentation. With the same model structures and training recipes, our SegGen boosts the performance of Mask2Former by a large margin and establishes a new SOTA performance without extra real data under both single-scale (s.s.) and multi-scale (m.s.) test settings.

3.4 Training Segmentation Models with Synthetic Data

The final goal of this work is to improve the performance of current segmentation models with the synthetically generated training data. Therefore, we use both synthetic data produced by SegGen and training data from existing datasets in the training process of segmentation models.

We empirically investigate two training strategies utilizing the synthetic data: (i) Synthetic data augmentation strategy. The synthetic data is used for random data augmentation. In every iteration in the training process, each real training sample is replaced by the synthetic training sample with a probability p_{aug}. (ii) Synthetic data pre-training strategy. It comprises two training stages: pre-training and fine-tuning. In the pre-training stage, we pre-train the segmentation models on synthetic data, so that they learn good weights that are transferable and favorable for fine-tuning. In the subsequent fine-tuning stage, the segmentation models are trained with solely human-annotated data.

4 Experiments

To accurately evaluate the effectiveness of our data generation method in improving segmentation performance, we adopt mainstream segmentation models and commonly used evaluation benchmarks for several typical segmentation tasks. The experiments are conducted mostly under the fully-supervised learning setting, meaning all human-annotated training samples from the evaluated datasets are used alongside our synthetic data. To guarantee an unbiased assessment of the impact of our synthetic data, we keep the architectures of the segmentation models and training protocols consistent with their respective original implementations throughout our studies.

4.1 Implementation Details

Segmentation Datasets and Evaluation We conduct experiments on three image segmentation benchmarks following the main experimental settings of

Mask2Former [7]: ADE20K semantic segmentation [60], COCO panoptic segmentation, and COCO instance segmentation [33]. Our evaluation uses all 150 classes for ADE20K and 133 classes for COCO. We use all the images from the training splits in the training of segmentation models. For semantic segmentation, we show the mean Intersection-over-Union metric (mIoU). For instance segmentation, the average precision (AP) is used. For panoptic segmentation, we report panoptic quality (PQ), "thing" instance segmentation $\text{AP}^{\text{Th}}_{\text{pan}}$, and semantic segmentation mIoU_{pan}.

Segmentation Models We adopt Mask2Former [7], a recently-proposed prevalent transformer model, as the default segmentation model. Two typical backbones, *i.e.*, R50 [16] and Swin-L [34], are studied. We keep the official implementation and training hyper-parameters of the segmentation models unchanged. For more training details please refer to their paper. We also conduct experiments on Mask DINO [27], a detection-aided segmentation model, and HRNet W48 [46], a representative fully-convolutional model. More implementation details are introduced in the supplementary materials.

Data Generation The Text2Mask and Mask2Img models are both based on the SDXL-base model [39]. They are trained on the training splits of the target segmentation datasets (*i.e.* ADE20K or COCO) separately for 30,000 iterations using a learning rate of 10^{-5}. AdamW optimizer is employed, and the models are trained at a resolution of 768. Pre-trained weights from SDXL-base [39] are utilized. Random flipping data augmentation is used. During sampling, the default EDM sampler [22] is used. The sampling steps are 200 in the Text2Mask model and 40 in the Mask2Img model. During data sampling, for each training sample in the ADE20K semantic segmentation dataset, we produce 10 synthetic mask-image pairs using MaskSyn, resulting in 202,100 training samples. Additionally, we synthesize 50 images based on each human-labeled mask with ImgSyn, leading to a total of 1,010,500 samples. On COCO, we solely use ImgSyn for data synthesis. By generating 10 synthetic images conditioned on each human-labeled panoptic segmentation mask via ImgSyn, our synthetic set amounts to 1,182,870 synthetic samples, which are used in the training of both panoptic and instance segmentation models.

4.2 Main Results

ADE20K Semantic Segmentation We employ the synthetic data augmentation strategy with $p_{\text{aug}} = 60\%$ for Mask2Former, and show the results in Table 1. SegGen significantly boosts the mIoU of Mask2Former R50 by $+$**2.7** for single-scale inference and $+$**2.2** for multi-scale inference, achieving 49.9 and 51.4 correspondingly. The Swin-L variant is also largely improved from 56.1/57.3 (single-scale/multi-scale) to 57.4/58.7 ($+$**1.3**/$+$**1.4**). Notably, our data synthesis method helps Mask2Former surpass the newer methods such as Mask DINO and OneFormer [19], while establishing new SOTA results for R50 and Swin-L settings without using additional human-annotated data.

COCO Panoptic Segmentation On COCO we adopt the synthetic data pre-training strategy to utilize our synthetic data. Specifically, we pre-train the mod-

Method	Backbone	Queries	Epochs	PQ	PQ^{Th}	PQ^{St}	AP^{Th}_{pan}	$mIoU_{pan}$
DETR [3]	R50	100	500+25	43.4	48.2	36.3	31.1	-
MaskFormer [8]	R50	100	300	46.5	51.0	39.8	33.0	57.8
Mask2Former [7]	R50	100	50	51.9	57.7	43.0	41.7	61.7
Mask DINO [27]	R50	300	50	53.0	59.1	43.9	43.3	-
Mask2Former [7]	R50	100	50+50	52.0	57.9	43.4	42.0	61.0
— w/ **SegGen (ours)**	R50	100	50+50	**52.7** (+0.7)	**58.8** (+0.9)	**43.6** (+0.2)	**43.1** (+1.1)	**62.6** (+1.6)
Mask DINO [27]	R50	300	50+50	53.4	59.3	44.4	44.2	60.5
— w/ **SegGen (ours)**	R50	300	50+50	**54.0** (+0.6)	**60.2** (+0.9)	**44.7** (+0.3)	**45.4** (+1.2)	**61.5** (+1.0)
MaskFormer [8]	Swin-L	100	300	52.7	58.5	44.0	40.1	64.8
OneFormer [19]	Swin-L	150	100	57.9	64.4	48.0	49.0	67.4
Mask2Former [7]	Swin-L	200	100	57.8	64.2	48.1	48.6	67.4
Mask DINO	Swin-L	300	50	58.3	65.1	48.0	50.6	-
Mask2Former [7]	Swin-L	200	100+100	57.3	64.3	46.8	48.0	66.2
— w/ **SegGen (ours)**	Swin-L	200	100+100	**58.0** (+0.7)	**64.5** (+0.2)	**48.1** (+1.3)	**48.8** (+0.8)	**67.4** (+1.2)
Mask DINO [27]	Swin-L	300	50+50	58.6	65.4	48.3	50.4	67.0
— w/ **SegGen (ours)**	Swin-L	300	50+50	**59.3** (+0.7)	**65.9** (+0.5)	**49.3** (+1.0)	**51.1** (+0.7)	**68.1** (+1.1)

Table 2: Panoptic segmentation on COCO panoptic `val2017` (133 categories): Synthetic data is used for pre-training. Our SegGen significantly improves the performance compared with real data pre-training baselines, and establishes new SOTA performance without extra real data.

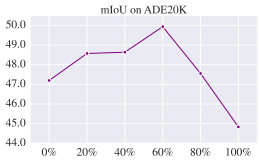

Fig. 7: ADE20K mIoU of Mask2Former R50 with different augmentation probability p_{aug}.

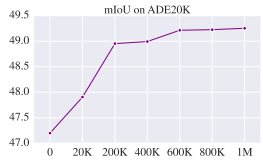

Fig. 8: ADE20K mIoU of Mask2Former R50 using different numbers of synthetic samples.

Method	mIoU*
Pure Real Data	83.3
DiffuMask (ICCV 2023)	57.0
MaskSyn+ImgSyn (ours)	**82.2**(+**25.2**)
MaskSyn (ours)	73.2
ImgSyn (ours)	80.0

Table 4: Comparison with DiffuMask [48] on ADE20K: All models trained purely on *synthetic data*. mIoU* is defined in [48].

els purely on our synthetic data and fine-tune the models on real data. The results are demonstrated in Table 2. For a fair comparison, we compare with models pre-trained on purely real data with the same training settings. Compared with the real data pre-trained baselines, SegGen consistently brings performance gains on all metrics with both R50 and Swin-L backbones, achieving new state-of-the-art performance without extra real data. Notably, for Mask DINO with Swin-L backbone, PQ is improved by +0.7, "thing" instance segmentation AP^{Th}_{pan} is increased by +0.7, and semantic segmentation $mIoU_{pan}$ is boosted by +1.1. These noteworthy findings demonstrate that training with our synthetic data systematically improves performance across various segmentation tasks. We observe similar significant improvement on Mask2Former, further showing the effectiveness of our synthetic data on different models.

COCO Instance Segmentation The performance on instance segmentation, both before and after incorporating our SegGen synthetic data, is detailed in Table 3. Compared to the baselines pre-trained on real data, our technique is superior across all metrics. Specifically, there is a solid +0.5 improvement in AP with the R50 backbone, and a +0.8 increase when using the Swin-L backbone. Additional assessment on the LVIS [14]-annotated COCO validation set, whose annotations are more accurate than the original version, provides additional evidence of the effectiveness of SegGen, as it enhances the average precision (AP) by +1.2 when used with the larger Swin-L backbone.

Method	Backbone	Queries	Epochs	AP	AP^S	AP^M	AP^L
Mask R-CNN [15]	R50	anchors	400	42.5	23.8	45.0	60.0
HTC [4]	R50	anchors	36	39.7	22.6	42.2	50.6
QueryInst [11]	R50	300	36	40.6	23.4	42.5	52.8
MaskFormer [8]	R50	100	300	34.0	16.4	37.8	54.2
Mask2Former [7]	R50	100	50	43.7	23.4	47.2	64.8
Mask2Former [7]	R50	100	50	47.1	26.6	55.4	72.7
Mask2Former [7]	R50	100	50+50	44.1	23.4	47.7	66.1
— w/ **SegGen (ours)**	R50	100	50+50	**44.6** (+0.5)	**24.0** (+0.6)	**48.2** (+0.5)	**66.2** (+0.1)
Mask2Former [7]	R50	100	50+50	47.9	27.8	56.3	73.5
— w/ **SegGen (ours)**	R50	100	50+50	**48.4** (+0.5)	**28.3** (+0.6)	**57.4** (+1.1)	**74.1** (+0.6)
Swin-HTC++ [34]	Swin-L	anchors	72	49.5	31.0	52.4	67.2
QueryInst [11]	Swin-L	300	50	48.9	30.8	52.6	68.3
Oneformer [19]	Swin-L	150	100	48.9	-	-	-
Mask2Former [7]	Swin-L	200	100	50.1	29.9	53.9	72.1
Mask2Former [7]	Swin-L	200	100	54.3	35.7	63.3	79.1
Mask2Former [7]	Swin-L	200	100+100	49.5	29.2	53.8	70.5
— w/ **SegGen (ours)**	Swin-L	200	100+100	**50.3** (+0.8)	**31.2** (+2.0)	**54.3** (+0.5)	**72.2** (+1.7)
Mask2Former [7]	Swin-L	200	100+100	53.6	35.2	62.6	77.8
— w/ **SegGen (ours)**	Swin-L	200	100+100	**54.8** (+1.2)	**35.9** (+0.8)	**64.2** (+1.6)	**79.2** (+1.5)

Table 3: Instance segmentation on COCO `val2017` (80 categories): Synthetic data is used for pre-training. The gray rows correspond to the evaluations conducted on LVIS [14] which offers instance masks of higher quality. Our SegGen significantly improves the performance of instance segmentation for various sizes of objects compared with pre-training on real data.

Comparison with Segmentation Data Generation Method In Table 4, we compare various versions of our SegGen with DiffuMask [48], which is a recently-proposed (ICCV2023) segmentation data generation approach. All methods employ the Mask2Former R50 model and are *purely trained on synthetic samples*. We adhere to the evaluation setting designed by DiffuMask where the IoU for three common classes are examined. Our method markedly surpasses DiffuMask by **+25.2 mIoU** and demonstrates performance comparable to training purely with real data, highlighting the unprecedented quality of our synthetic data.

Generalization Ability on Unseen Domains We visualize the segmentation results of Mask2Former R50 on images from PASCAL `val` dataset [10] in Fig. 1 and Fig. 9, comparing models trained both with and without our synthetic data. The models are trained on COCO or ADE20K. When trained using our extensively varied synthetic data, the segmentation model demonstrates notably improved performance on the unseen domain. We further study the segmentation results on AI-generated images, which are synthesized using different image generation models, in the supplementary materials.

Visual Analysis of Generated Samples by MaskSyn We randomly select some samples generated by MaskSyn in Fig. 4 (more in supplementary materials). The generative models are trained on ADE20K. Given the text prompts, our Text2Mask model can generate diverse segmentation masks, and the Mask2Img model produces realistic synthetic images with a good alignment with the masks and text prompts.

Visual Analysis of Generated Samples by ImgSyn We showcase the outputs by ImgSyn in Fig. 5 (more in supplementary materials). The generated images are in good agreement with the text prompts and human-labeled segmentation masks. Furthermore, we compare the mask-image alignment of the

Fig. 9: Generalization Ability on unseen domains: Segmentation outputs on unseen domain (images from PASCAL [10]). The models are trained on ADE20K. Training with our SegGen demonstrates enhanced robustness towards unfamiliar domains.

Method	Iterations	Real Samples	mIoU
Baseline	160K	20210	47.2
— w/ **SegGen** MaskSyn	160K	20210	48.5
— w/ **SegGen** ImgSyn	160K	20210	49.3
— w/ **SegGen** ImgSyn + MaskSyn	160K	20210	**49.9**
Baseline	160K	1000	23.7
— w/ **SegGen** MaskSyn	160K	1000	26.5
— w/ **SegGen** ImgSyn	160K	1000	24.1
— w/ **SegGen** ImgSyn + MaskSyn	160K	1000	**27.1**

(a) Mask2Former R50

Method	Iterations	Real Samples	mIoU
Baseline	100K	20210	44.4
— w/ **SegGen** MaskSyn	100K	20210	44.8
— w/ **SegGen** ImgSyn	100K	20210	45.4
— w/ **SegGen** ImgSyn + MaskSyn	100K	20210	**45.6**
Baseline	100K	1000	23.6
— w/ **SegGen** MaskSyn	100K	1000	26.0
— w/ **SegGen** ImgSyn	100K	1000	23.9
— w/ **SegGen** ImgSyn + MaskSyn	100K	1000	**26.9**

(b) HRNet W48

Table 5: Ablation Study of MaskSyn and ImgSyn on ADE20K: Both MaskSyn and ImgSyn notably enhance the performance of Mask2Former and HRNet, while MaskSyn has a more pronounced impact when there are fewer real samples available.

real training samples and our synthetic training samples in Fig. 6. Our synthetic samples exhibit superior alignment quality in many cases, compared to human annotations which are often imperfect on the boundaries.

4.3 Quantitative Analysis

Effectiveness of MaskSyn and ImgSyn We evaluate the data generated by MaskSyn and ImgSyn separately and jointly, by examining their impacts on the performance of segmentation models, as shown in Table 5. The experiments are conducted on ADE20K with Mask2Former R50 and HRNet W48. We find that the training samples generated by both MaskSyn and ImgSyn bring significant performance gains compared with the baseline. When combining these two types of synthetic data together, the segmentation model can achieve the best performance. We delve deeper into a scenario where only 1,000 real training samples are available in the training of both generative models and segmentation models. It is observed that MaskSyn substantially enhances the segmentation results in situations with limited real data. This could be attributed to the ability of MaskSyn to amplify data diversity by creating entirely new segmentation masks and images. Moreover, the significant performance improvements achieved by our SegGen, trained on merely 1,000 real samples, demonstrate the data efficiency and robustness of our proposal.

Influence of Synthetic Data Augmentation Probability p_{aug} on ADE20K We investigate the effect of p_{aug} in Fig. 7. When $p_{\text{aug}} = 0$, it implies training using only real data, whereas $p_{\text{aug}} = 100\%$ means training entirely with synthetic

Method	Iterations	mIoU
Syn. Pre-Train	160K+160K	47.2
Syn. Data Aug	160K	**49.9**

(a) ADE20K Semantic Segmentation: Using synthetic data as random data augmentation obtains the best performance on ADE20K.

Method	Epochs	PQ	AP
Syn. Pre-Training	50+50	**52.7**	**44.6**
Syn. Data Aug	50	51.3	43.2

(b) COCO Panoptic Segmentation and Instance Segmentation: Using synthetic data for pre-training achieves better performance on COCO dataset.

Table 6: Different training strategies with synthetic data using Mask2Former R50.

data. The best results are achieved at $p_{\text{aug}} = 60\%$. It is noteworthy that training exclusively on our synthetic data results in an impressive 44.8 mIoU.

Influence of Synthetic Data Size To evaluate the impact of synthetic data volume, we use different numbers of synthetic training samples produced by ImgSyn in training and report the results in Fig. 8. p_{aug} is fixed at 60%. We notice a substantial improvement in performance when increasing the synthetic data quantity from 20K to 200K, underscoring the importance of collecting a significantly larger synthetic dataset to avoid overfitting. The performance appears to plateau after 600K. More studies on the scale of synthetic data are presented in the supplementary materials.

Influence of Training Strategies We conduct an ablation study using different training strategies with synthetic data on ADE20K (Table 6a) and COCO (Table 6b). We find that using the synthetic data augmentation strategy achieves better performance on ADE20K semantic segmentation, While on COCO panoptic and instance segmentation, the synthetic pre-training strategy works better. We believe that due to the limited size of the ADE20K training set, which contains merely ∼20k images, segmentation models easily overfit the scarcely-available training images and segmentation layouts. Utilizing synthetic data (including synthetic masks and images) for data augmentation can help mitigate such an overfitting problem. On the other hand, for COCO, given its larger training set size (around 100K images), overfitting is less of a concern. Hence, it is sufficient to provide the COCO models with good initial weights pre-trained on our synthetic data. Meanwhile, previous work [14] finds that the annotation accuracy of COCO is significantly more biased than ADE20K, a finding that aligns with our visualization results depicted in Fig. 6. A domain gap exists between the COCO data and our high-quality synthetic data. Hence, employing our method for pre-training and incorporating real data for fine-tuning on COCO is a more favorable approach.

5 Conclusion

We present SegGen, a highly-effective data synthesis method for image segmentation. SegGen introduces a revolutionary segmentation data generation framework, and proposes two data generation approaches, namely MaskSyn and ImgSyn, to synthesize segmentation masks and images, with the help of the designed text-to-mask and mask-to-image generation models. Our method not only strongly enhances the model performance on semantic, panoptic, and instance segmentation benchmarks, but also substantially improves segmentation generalization ability in unseen data distribution.

Acknowledgement

This research is supported in part by the Early Career Scheme of the Research Grants Council (RGC) of the Hong Kong SAR under grant No. 26202321, SAIL Research Project, HKUST-Zeekr Collaborative Research Fund, HKUST-WeBank Joint Lab Project, and Tencent Rhino-Bird Focused Research Program.

References

1. Baranchuk, D., Rubachev, I., Voynov, A., Khrulkov, V., Babenko, A.: Label-efficient semantic segmentation with diffusion models. In: ICLR (2022)
2. Brock, A., Donahue, J., Simonyan, K.: Large scale gan training for high fidelity natural image synthesis. In: ICLR (2019)
3. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: ECCV (2020)
4. Chen, K., Pang, J., Wang, J., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Shi, J., Ouyang, W., et al.: Hybrid task cascade for instance segmentation. In: CVPR (2019)
5. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected crfs. In: ICLR (2015)
6. Chen, T., Li, L., Saxena, S., Hinton, G., Fleet, D.J.: A generalist framework for panoptic segmentation of images and videos. arXiv preprint arXiv:2210.06366 (2022)
7. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: CVPR (2022)
8. Cheng, B., Schwing, A.G., Kirillov, A.: Per-pixel classification is not all you need for semantic segmentation. In: NeurIPS (2021)
9. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. In: NeruIPS (2021)
10. Everingham, M., Eslami, S.A., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The PASCAL visual object classes challenge: A retrospective. IJCV (2015)
11. Fang, Y., Yang, S., Wang, X., Li, Y., Fang, C., Shan, Y., Feng, B., Liu, W.: Instances as queries. In: ICCV (2021)
12. Forever, A.: Kandinsky-2. GitHub repository (2023), `https://github.com/ai-forever/Kandinsky-2`
13. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. CACM (2020)
14. Gupta, A., Dollar, P., Girshick, R.: LVIS: A dataset for large vocabulary instance segmentation. In: ICCV (2019)
15. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: ICCV (2017)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
17. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: NeruIPS (2020)
18. Huang, L., Chen, D., Liu, Y., Yujun, S., Zhao, D., Jingren, Z.: Composer: Creative and controllable image synthesis with composable conditions. arXiv preprint arxiv:2302.09778 (2023)

19. Jain, J., Li, J., Chiu, M.T., Hassani, A., Orlov, N., Shi, H.: Oneformer: One transformer to rule universal image segmentation. In: CVPR (2023)
20. Ji, Y., Chen, Z., Xie, E., Hong, L., Liu, X., Liu, Z., Lu, T., Li, Z., Luo, P.: Ddp: Diffusion model for dense visual prediction. arXiv preprint arXiv:2303.17559 (2023)
21. Kang, M., Zhu, J.Y., Zhang, R., Park, J., Shechtman, E., Paris, S., Park, T.: Scaling up gans for text-to-image synthesis. In: CVPR (2023)
22. Karras, T., Aittala, M., Aila, T., Laine, S.: Elucidating the design space of diffusion-based generative models. In: NeruIPS (2022)
23. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: CVPR (2019)
24. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
25. Kirillov, A., He, K., Girshick, R., Rother, C., Dollár, P.: Panoptic segmentation. In: CVPR (2019)
26. Li, D., Ling, H., Kim, S.W., Kreis, K., Fidler, S., Torralba, A.: Bigdatasetgan: Synthesizing imagenet with pixel-wise annotations. In: CVPR (2022)
27. Li, F., Zhang, H., Xu, H., Liu, S., Zhang, L., Ni, L.M., Shum, H.Y.: Mask dino: Towards a unified transformer-based framework for object detection and segmentation. In: CVPR (2023)
28. Li, J., Li, D., Savarese, S., Hoi, S.: BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In: ICML (2023)
29. Li, X., Zhong, Z., Wu, J., Yang, Y., Lin, Z., Liu, H.: Expectation-maximization attention networks for semantic segmentation. In: ICCV (2019)
30. Li, X., Ding, H., Zhang, W., Yuan, H., Pang, J., Cheng, G., Chen, K., Liu, Z., Loy, C.C.: Transformer-based visual segmentation: A survey. arXiv preprint arXiv:2304.09854 (2023)
31. Li, Y., Liu, H., Wu, Q., Mu, F., Yang, J., Gao, J., Li, C., Lee, Y.J.: Gligen: Open-set grounded text-to-image generation. In: CVPR (2023)
32. Li, Z., Zhou, Q., Zhang, X., Zhang, Y., Wang, Y., Xie, W.: Open-vocabulary object segmentation with diffusion models. In: ICCV (2023)
33. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: ECCV (2014)
34. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. arXiv:2103.14030 (2021)
35. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR (2015)
36. Mou, C., Wang, X., Xie, L., Wu, Y., Zhang, J., Qi, Z., Shan, Y., Qie, X.: T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. arXiv preprint arXiv:2302.08453 (2023)
37. Nguyen, Q.N.T.V.A.T.K.: Dataset diffusion: Diffusion-based synthetic dataset generation for pixel-level semantic segmentation. arXiv preprint arXiv:2309.14303 (2023)
38. PNVR, K., Singh, B., Ghosh, P., Siddiquie, B., Jacobs, D.: Ld-znet: A latent diffusion approach for text-based image segmentation. In: ICCV (2023)
39. Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., Rombach, R.: Sdxl: Improving latent diffusion models for high-resolution image synthesis. arXiv preprint arXiv:2307.01952 (2023)
40. Qi, L., Kuen, J., Wang, Y., Gu, J., Zhao, H., Torr, P., Lin, Z., Jia, J.: Open world entity segmentation. TPAMI (2022)

41. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR (2022)
42. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic text-to-image diffusion models with deep language understanding. In: NeruIPS (2022)
43. Sauer, A., Karras, T., Laine, S., Geiger, A., Aila, T.: StyleGAN-T: Unlocking the power of GANs for fast large-scale text-to-image synthesis. In: ICML (2023)
44. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: ICML (2015)
45. Wang, H., Cao, J., Anwer, R.M., Xie, J., Khan, F.S., Pang, Y.: Dformer: Diffusion-guided transformer for universal image segmentation. arXiv preprint arXiv:2306.03437 (2023)
46. Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W., Xiao, B.: Deep high-resolution representation learning for visual recognition. TPAMI (2019)
47. Wu, W., Zhao, Y., Chen, H., Gu, Y., Zhao, R., He, Y., Zhou, H., Shou, M.Z., Shen, C.: Datasetdm: Synthesizing data with perception annotations using diffusion models. arXiv preprint arXiv:2308.06160 (2023)
48. Wu, W., Zhao, Y., Shou, M.Z., Zhou, H., Shen, C.: Diffumask: Synthesizing images with pixel-level annotations for semantic segmentation using diffusion models. In: ICCV (2023)
49. Xie, J., Li, W., Li, X., Liu, Z., Ong, Y.S., Loy, C.C.: Mosaicfusion: Diffusion models as data augmenters for large vocabulary instance segmentation. arXiv preprint arXiv:2309.13042 (2023)
50. Xu, J., Liu, S., Vahdat, A., Byeon, W., Wang, X., De Mello, S.: Open-vocabulary panoptic segmentation with text-to-image diffusion models. In: CVPR (2023)
51. Yang, L., Xu, X., Kang, B., Shi, Y., Zhao, H.: Freemask: Synthetic images with dense annotations make stronger segmentation models. In: NeurIPS (2023)
52. Ye, H., Xu, D.: Taskprompter: Spatial-channel multi-task prompting for dense scene understanding. In: ICLR (2023)
53. Zeng, Y., Lin, Z., Zhang, J., Liu, Q., Collomosse, J., Kuen, J., Patel, V.M.: Scenecomposer: Any-level semantic image synthesis. In: CVPR (2023)
54. Zhang, L., Anyi, R., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: ICCV (2023)
55. Zhang, W., Pang, J., Chen, K., Loy, C.C.: K-net: Towards unified image segmentation. In: NeurIPS (2021)
56. Zhang, Y., Ling, H., Gao, J., Yin, K., Lafleche, J.F., Barriuso, A., Torralba, A., Fidler, S.: Datasetgan: Efficient labeled data factory with minimal human effort. In: CVPR (2021)
57. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: CVPR (2017)
58. Zhao, S., Chen, D., Chen, Y.C., Bao, J., Hao, S., Yuan, L., Wong, K.Y.K.: Uni-controlnet: All-in-one control to text-to-image diffusion models. NeurIPS (2023)
59. Zhao, W., Rao, Y., Liu, Z., Liu, B., Zhou, J., Lu, J.: Unleashing text-to-image diffusion models for visual perception. In: ICCV (2023)
60. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing challenge 2016. `http://sceneparsing.csail.mit.edu/index_challenge.html` (2016)
61. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ADE20K dataset. In: CVPR (2017)