Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institut für Geodäsie und
Photogrammetrie

Towards Exploiting the Generative Image Diffusion Models for Unsupervised Semantic Segmentation Domain Adaptation

Master's Thesis

Yuru Jia

Department of Civil, Environmental and Geomatic Engineering
ETH Zürich

Advisors:	Dr. Anton Obukhov
	Shengyu Huang
	Lukas Hoyer
Supervisor:	Prof. Dr. Konrad Schindler

July 31st, 2023

Acknowledgements

I would like to thank several individuals who have played a pivotal role in the completion of this thesis:

- **Prof. Dr. Konrad Schindler** for providing me with the opportunity to work on this project, and for offering insightful feedback and access to computing resources.
- **Dr. Anton Obukhov** for undertaking the supervision and offering unwavering support throughout the project. His willingness to offer fresh perspectives and engage in brainstorming sessions have been immensely valuable.
- **Shengyu Huang** for undertaking the supervision and providing profound insights. I am grateful for his continuous encouragement. Additionally, his valuable suggestions for presentations have been tremendously beneficial.
- **Lukas Hoyer** for undertaking the supervision and providing technical assistance, particularly with regards to domain adaptation aspects. His patience and commitment to addressing all my inquiries have been truly remarkable.

Last but not least, I would like to extend my thanks to my family members and my friends who have supported me throughout this journey.

Abstract

Deep generative models have made significant progress in generating high-fidelity, photorealistic images from textual cues. Though the results are impressive to human eyes, it remains unclear if these synthetic images can be used to improve the performance of visual recognition tasks, such as semantic or panoptic segmentation, especially in the challenging unsupervised domain adaptation (UDA) context. UDA aims to train models on more accessible synthetic data and adapt them to real images without requiring their costly pixel-wise annotations. In this research, we investigate the applicability of state-of-the-art text-to-image generative diffusion models within an unsupervised semantic segmentation domain adaptation scenario. Using labeled source images and unlabeled target images, we probe the capability of Stable Diffusion in generating image-label pairs akin to the target domain. We then conduct a closed-loop evaluation, training the model on our generated dataset and assessing its performance on real-world datasets. We employ a conditional diffusion model, ControlNet, trained on labeled source data to facilitate alignment and implement U-Net swap to achieve style transfer, thereby producing a highly realistic labeled dataset. Additionally, we propose strategies for better improving the alignment of the generated data. We show that a semantic segmentation model trained on our generated dataset outperforms those trained on conventional game-engine synthetic datasets, highlighting the promising role of data generation techniques in addressing UDA challenges.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Supervised learning requires a large volume of annotations, which can be particularly expensive and time-consuming to acquire, especially for pixel-wise semantic segmentation tasks. A common solution to address such annotation-heavy tasks is to utilize synthetic data, typically synthesized via renderers or simulators, allowing for the inexpensive creation of extensive labeled datasets. For instance, the GTA dataset [48] generates pixel-level semantic segmentation ground truth for 25,000 images from modern computer games, a task completed in merely 49 hours. Conversely, annotating a single real-world image, like those from the Cityscapes dataset [13], would take a person around 1.5 hours. Synthetic datasets extremely reduce human labor costs, however, training with such data induces domain generalization challenges due to the discrepancies between synthetic and real-world data. In the GTA dataset, object arrangements follow specific patterns and distributions that may not accurately replicate real-world environments. Fine-grained annotations for certain classes, such as bicycles and trees differ from human annotation policies. Additionally, the domain gap between these datasets extends to aspects such as texture, image quality, and environmental variations.

To mitigate the domain shift challenge, numerous studies have explored unsupervised domain adaptation (UDA). UDA deals with three domains: the prior domain, the target domain, and the source domain. The prior domain involves large-scale natural images from datasets like ImageNet [14] and LAION [55]. These datasets are commonly used to pre-train models like [56], as they facilitate learning a wide variety of features and gaining a generalized understanding of images. The target domain consists of raw images from a specific data distribution, such as the Cityscapes dataset [13], where the model is expected to perform well. However, no annotations are available for training in the target domain. The source domain contains images with comparable content and meaning to the target domain but suffers from low quality and exhibits a clear domain gap. Nevertheless, these source domain images are accompanied by a wealth of semantic annotations, such as the GTA dataset [48]. The primary objective of UDA is to adapt the model trained on the source domain to perform well in the target domain without access to target labels, potentially leveraging the knowledge gained from the prior domain.

Previous UDA methods [31, 32, 9, 30, 63, 22] primarily transfer knowledge through two approaches: adversarial training and self-training. In adversarial training, the segmentation network is considered a generator in a GAN setup, and a domain discriminator is used to encourage domain-invariant knowledge learning. In self-training, pseudo-labels for the target domain are generated based on models trained on the source domain, providing supervision to retrain the model. This process is repeated several times. Various data augmentation techniques are also employed for the network to learn robust features, such as rotation, color jitter, and ClassMix [44], etc.

Recently diffusion models [28, 58] have achieved extraordinary results in image generation tasks. Building on the foundational formulation of diffusion models, several text-to-image models [53, 49, 46] have exhibited enhanced utility in handling image-text relationships and leveraging large-scale training data. For example, Stable Diffusion [49], trained on billion-scale datasets [55] composed of noisy image-caption pairs collected from the internet, showcases the exceptional capacity to generate high-fidelity, photorealistic images from textual prompts. This leads us to a natural question, i.e., can high-realism domain-like datasets be generated to alleviate the domain generalization issue to help UDA? This perspective shifts the focus onto the training data itself instead of UDA methods. Several works [26, 2, 54, 39] have made attempts to utilize images generated by diffusion models for downstream tasks. For instance, [26] studies how generated data can improve classification models in data-scarce settings, while [54] trains classification models using different ImageNet clones generated by Stable Diffusion and compares the performance of the model with one trained on real images. Despite demonstrating the effectiveness of generated data, these studies predominantly focus on classification tasks, naturally employing class names as prompts, while semantic segmentation tasks which require more complex pixel-wise labeling are yet to be explored.

In this study, we aim to explore the advantages and challenges of using generative image diffusion models for the semantic segmentation task and investigate their usefulness under the UDA scenario. Our primary objective is to generate target domain-like images along with their corresponding semantic segmentation labels, utilizing pretrained text-to-image diffusion models. Furthermore, we evaluate the effectiveness of our generated dataset by training a segmentation model on it and assessing its performance on the validation dataset from the target domain.

1.1 Focus of this Work

We try to answer the following three questions:

- *How to employ diffusion models to generate images and corresponding segmentation masks?* Essentially, given complete access to a specific domain, our objective is to generate images that resemble this domain and are labeled correspondingly. It is worth noting that we operate within a single domain for this task, in order to explore the generative capabilities of diffusion models by simplifying the problem.

- *How to adapt a diffusion model pre-trained on the source domain to the target domain?* Once the model is ready for paired image and label generation, the aim within the UDA context is to transfer the style of the generated data into the target images, while also maintaining the aligned generation ability of the generative diffusion model learned from the source domain data.

- *How useful are the generated datasets in the UDA setting?* To examine the viability of the generated dataset for pre-training the semantic segmentation model, we implement a closed-loop evaluation by testing the model on a real-world dataset.

1.2 Thesis Organization

This thesis is organized as follows:

- Chapter 2 introduces works related to our topic. The focus primarily lies on controllable generative diffusion models and their applications. Additionally, we will review their utilization in dataset synthesis.

- Chapter 3 details our proposed approach. The problem statement is clarified. We explain the methodology from the viewpoint of labeled data generation and style transfer, and further propose strategies to tackle specific challenges within the dataset.

- Chapter 4 illustrates the experiments and presents the results accordingly. Experiments in different settings and implementation details are explained. Ablations to examine the synthesis quality of our dataset are also provided.

- Chapter 5 discusses our observations, potential applications, and future perspectives of our approach.

- Chapter 6 presents the conclusions drawn from our work.

Chapter 2

Related Work

In this chapter, we review works related to generative diffusion models, with a particular emphasis on controllable diffusion models and their diverse applications. Additionally, we delve into the subject of generating synthetic datasets using these generative models.

2.1 Generative Diffusion Models

Advancements in deep learning techniques have led to a surge in deep generative models. Variational autoencoders (VAEs) [36] model a lower bound, implicitly learning the probability density over the latent space. Generative adversarial networks (GANs) [23] provide a sampling mechanism for generating new data, without offering a likelihood estimate. On the other hand, normalizing flows [47] model the true data distribution by using a sequence of invertible functions. However, even the most advanced of these methods, GANs, still encounter challenges such as training instability and mode collapse issues [6]. Recently, diffusion probabilistic models have demonstrated state-of-the-art image generation quality, and we will delve into these diffusion models in the following sections.

Diffusion Probabilistic Models. Diffusion probabilistic models[57, 28] are capable of creating data through a process of iterative denoising. Their approach involves a forward process in which noise is added into data distributions, which is subsequently reversed in order to recover the original data. These methods essentially use U-Net [50] as their neural network architecture. Notably, Dhariwal *et al.* [15] introduce various techniques such as architectural improvements and classifier guidance, that help diffusion models beat GANs [23] in image generation tasks. Song *et al.*propose Denoising diffusion implicity models (DDIM) [58] to improve sampling methods. An inherent challenge in image diffusion methods is their direct usage of pixel colors as training data, leading to computational intensiveness. As a result, a substantial amount of research [49, 46, 64] has been focused on reducing computational power requirements and scaling up these models. Especially, latent diffusion model (LDM) [49] is a prominent model that decreases computational costs by applying the diffusion process to a low-resolution latent space.

Text-to-Image (T2I) Diffusion Models. With improved inference speed and lower memory cost, diffusion models can be successfully scaled for text-to-image generation with webscale data. Radford *et al*. [45] introduce CLIP as scalable approach for learning joint representations between text and images. The CLIP model includes an image encoder and a caption encoder. During its training phase, the model is optimized via a contrastive cross-entropy loss, which promotes a high dot product for pairs of images and their associated captions. The result is a model capable of effectively mapping text descriptions to corresponding images. Following the introduction of CLIP, a number of subsequent studies [43, 49, 46] have utilized it to guide their image generation processes. For instance, the Stable Diffusion (SD) model [49] leverages the text encoder of CLIP to transform text tokens into a textual representation. This representation is then used as an input for the cross-attention layer within a denoising U-Net model, helping to guide the image generation process. To gain a deep level of language understanding, Imagen [53] uses larger pretrained frozen language models, and introduces a comprehensive and challenging benchmark for text-to-image models.

2.2 Controllable Diffusion Models

Text prompt offers only an approximate definition of an object's position or appearance in images, presenting a limited degree of controllability. To improve this, there is a growing need to integrate more diverse control modes, such as user-generated sketches, semantic masks, reference styles, personalization features, and so on, in conjunction with the text description in these models.

Conditional Image Generation. Conditional generation methods require training new diffusion models that accept the prompt as an additional input [29, 41, 72, 33, 74, 20, 24]. For instance, solutions like ControlNet [72] and T2I-Adapter [41] seamlessly integrate lightweight adapters into pre-existing T2I diffusion models. This integration facilitates the addition of extra condition signals, making the process of fine-tuning more cost-effective. Zhao *et al*. [74] further categorizes various conditions into local conditions and global conditions and proposes Uni-ControlNet to combine multiple conditions. [20] inject structure information (a segmentation map) and appearance information (image features from a pretrained encoder) into the diffusion model, allowing for object-level modifications. Several researchers [12, 10, 34] have expanded these methods to control video generation or 3D creation. For example, [10] builds upon ControlNet and introduces a technique based on residual noise initialization, which leverages prior motion data to produce consistent videos.

Guided Image Generation. Research in this area [15, 3, 5] uses pre-trained diffusion models as base models and tweaks the sampling process to direct image generation based on feedback from the guidance function. Dhariwal *et al*. [15] propose classifier guidance, in which a classifier is trained on images across different noise scales as the guidance function f. This function's gradients are incorporated during the sampling stage. [3] investigates universal guidance algorithms that employ any off-the-shelf guidance functions f, like object detection or segmentation networks, to guide image generation with diffusion models. Also, several studies modify the self-attention and cross-attention maps to influence the sampling process. For instance, [4] facilitates structure-guided generation by manipulating intermediate self-attention maps of a masked generative transformer. [27] modifies images by injecting the cross-attention maps during the diffusion process, regulating which pixels pay attention to which tokens of the prompt text at various diffusion stages.

Personalized Text-to-Image Synthesis. Numerous studies have explored how to customize images of individual items by leveraging the power of pre-trained text-to-image models. [18] finds text representations (e.g., embedding, token) corresponding to a set of images of an object without changing the parameters of the text-to-image model. DreamBooth [52] on the other hand, fine-tunes the whole text-to-image model based on a few images that depict the subject of interest, offering more expressiveness and detailed capture of the subject. Custom Diffusion [37] and SVD-iff [25] have taken DreamBooth a step further by enabling the simultaneous synthesis of multiple subjects, offering advantages such as a smaller model size and faster fine-tuning speed.

Image Style Transfer with Diffusion Models. A large body of work has investigated style transfer using deep networks by solving a composite objective of style and content consistency [19]. In this context, we will discuss a few approaches that leverage diffusion models. Kwon and Ye [38] propose a method that guides the style transfer process using content and style inference. Additionally, Kawar *et al*. [35] introduce optimization-based methods for style transfer. However, these methods often require considerable computational resources for inference and carefully adjusted hyperparameters. On the other hand, ArtFusion [7] takes a different approach by treating both content and style as conditions for the LDM, demonstrating its potential for effective style transfer.

2.3 Synthetic Datasets

Utilizing generative models for meaningful data generation to support downstream applications has been a subject of active research. DatasetGAN [73] upsamples feature maps from StyleGAN and trains a supplementary label branch for StyleGAN using a limited number of labeled examples. This technique permits the automatic generation of images and their corresponding pixel-wise labels, yielding results that surpass semi-supervised baselines in object-part segmentation. Sun *et al*.[59] leverage pre-trained GAN models to synthesize images and object bounding boxes guided by the few-shot samples, helping object detection domain adaptation in a source-free few-shot manner. Owing to superior training stability and convergence, recent techniques [26, 54, 2, 16, 39, 60, 65] have begun adopting cutting-edge diffusion-based models for dataset creation. Works like [26, 54, 2] make use of large-scale pretrained TI2 diffusion models with label names built as language inputs to generate images, and the coupled labels and images are then used to train classification models, thereby examining the potential usability of synthetic data for image recognition. [39] takes a different route by studying the applicability of synthetic data to knowledge distillation. [16, 60] augment the training data and evaluate the model on image classification tasks, but they employ distinct methods to create augmented data. The former leverages large language models and text-conditioned image editing methods, while the latter edits images to alter their semantic attributes.

While these studies have shown the potential usefulness of synthetic data generated from state-of-the-art diffusion models, the scope of these investigations has been largely confined to classification issues or object detection, and the tasks are typically limited to particular settings. In this thesis, we focus on more complex semantic segmentation tasks and investigate the feasibility of using the generated data for downstream applications.

2.4 UDA for Semantic Segmentation

The domain-adaptive semantic segmentation field has been actively researched, especially with the emergence of autonomous industries. Under the UDA scenario, semantic segmentation tasks involve input images from both source and target domains, along with the source's semantic ground truth labels. These tasks are generally approached using two main frameworks. The first framework utilizes GANs, which leverage adversarial loss to align the source and target domain distributions [66, 22, 62, 63]. The second approach employs a self-training framework [77, 31, 61, 67, 32], where a "teacher" model trained on the labeled data annotates the unlabeled data with pseudo-labels. Subsequently, the "student" model learns from both labeled and pseudo-labeled data, and this process is iterated. Moreover, various data augmentation techniques have been incorporated into many works [1, 32, 11, 44, 61, 76]. For instance, [1] employs photometric noise, multi-scale fusion, and random flipping to encourage model invariance to photometric perturbations. [61, 76] use cross-domain mixup strategies for consistency regularization. In the typical UDA setting for semantic segmentation, commonly used source datasets are Synthia [51] and GTA5 [48], while Cityscapes [13] and Mapillary Vistas [42] are frequently adopted as target datasets.

Chapter 3

Methodology

In this chapter, we lay out the methodologies adopted in this study, with the goal of aiding unsupervised domain adaptation using generative diffusion models to produce high-realism datasets. The initial step explores the synthesis capabilities of the generative diffusion model, with a particular focus on generating images and corresponding labels within a given domain. Thereafter, we shift our focus to stylization techniques that empower the model to transfer style from one domain to another while retaining the knowledge derived from the original domain. This approach enables us to fully utilize the wealth of segmentation masks in the source domain where masks are readily available, and simultaneously generate images reminiscent of the target domain. We also confront two challenges commonly encountered in dataset handling, namely, imbalanced distribution and the presence of small objects. To address these issues, we have employed two strategies: rare class sampling and small component refinement, which are aimed at mitigating bias and enhancing the accuracy of the model's conditioning respectively.

This chapter is organized as follows:

1. We first outline the primary goal of this research and formulate the problem.
2. We discuss three pathways in labeled image generation for semantic segmentation, namely grounding-based, guidance-based, and condition-based generation.
3. We show the DreamBooth fine-tuning technique for different domain representations.
4. We present the complete model employing U-Net swap to generate an aligned, yet realistic, labeled dataset.
5. We propose strategies to address issues of imbalanced class distribution and the absence of small objects.

3.1 Problem Statement

The aim of this thesis is to produce images that closely resemble the target domain and their corresponding semantic segmentation masks, in the context of unsupervised semantic segmentation domain adaptation. Given the labeled source domain (a synthetic dataset), represented as $\mathbf{D}^S = \left\{\left(x_i^S, y_i^S\right)\right\}_{i=1}^{N_s}$ and the unlabeled target domain (a real dataset), symbolized as $\mathbf{D}^T =$

$\left\{x_i^T\right\}_{i=1}^{N_t}$, the primary objective is to create the target domain-like labeled dataset, denoted by $\mathbf{D}^G=\left\{\left(x_i^G, y_i^G\right)\right\}_{i=1}^{N_g}$. In these notations, x and y stand for the images and their corresponding labels, respectively, whereas N_s, N_t, and N_g symbolize the total number of images present in each dataset. The generated images are expected to closely match the distribution of the target domain, implying that x_i^G should bear a resemblance to $\hat{x}^T$.

3.2 Preliminary: Stable Diffusion

In this study, we base our methodology on the recent state-of-the-art text-to-image diffusion model, i.e., Stable Diffusion (SD) [49]. SD is a two-stage diffusion model, which contains an autoencoder and an U-Net denoiser. During the first phase, SD trains an autoencoder with the capacity to transform natural images, denoted as $\mathbf{X}_0$, into a latent space and subsequently reconstruct them. Following this, in the second phase, a modified U-Net [50] denoiser is trained by SD to execute denoising operations directly within the latent space. In the inference phase, the input latent map $\mathbf{Z}_T$ is randomly generated from a Gaussian distribution. With $\mathbf{Z}_T$ given, it provides a noise estimation ϵ at each iteration t and subtracts it from $\mathbf{Z}_T$. The final output $\mathbf{Z}_0$, representing the uncorrupted latent, is passed into the autoencoder's decoder to yield natural images. In the conditional part, SD employs the pretrained CLIP [45] text encoder to convert text inputs into embedding sequences y. It then leverages a cross-attention model to integrate y into the denoising procedure. This can be expressed as:

$$\left\{\begin{array}{l}\mathbf{Q}=\mathbf{W}_Q \cdot \phi\left(\mathbf{Z}_t\right) ; \mathbf{K}=\mathbf{W}_K \cdot \tau(\mathbf{y}) ; \mathbf{V}=\mathbf{W}_V \cdot \tau(\mathbf{y}) \\ \operatorname{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V})=\operatorname{softmax}\left(\frac{\mathbf{Q K}^T}{\sqrt{d}}\right) \cdot \mathbf{V}\end{array}\right. \tag{3.1}$$

where $\phi(\cdot)$ and $\tau(\cdot)$ represent two learnable embeddings, while W_Q, W_K, and W_V denote learnable projection matrices.

3.3 Labeled Image Generation

Pretrained text-to-image diffusion models have demonstrated remarkable proficiency in generating images based on prompts. However, to allow for the automatic acquisition of corresponding segmentation masks, the model needs additional capabilities to manipulate and work with masks within its structure. To explore this, we begin by simplifying the problem by operating within a single domain. Here, we have paired images and labels for a specific domain, and the goal is to train the model to recognize the semantic information of the generated image.

To achieve this goal, we explore different approaches where segmentation masks play distinct roles in the generation process. In the grounding-based generation approach, masks are generated as an augmented product of the diffusion model. In contrast, in the guidance-based and condition-based generation approaches, existing segmentation masks are employed to influence and guide the image generation process, albeit in different manners: a non-learning-based fashion for the former and a learning-based fashion for the latter.

We provide a detailed introduction to each method below, and we will select the most promising one for the remainder of our study. The performance of each method and their advantages and disadvantages are presented in section 4.3.1.

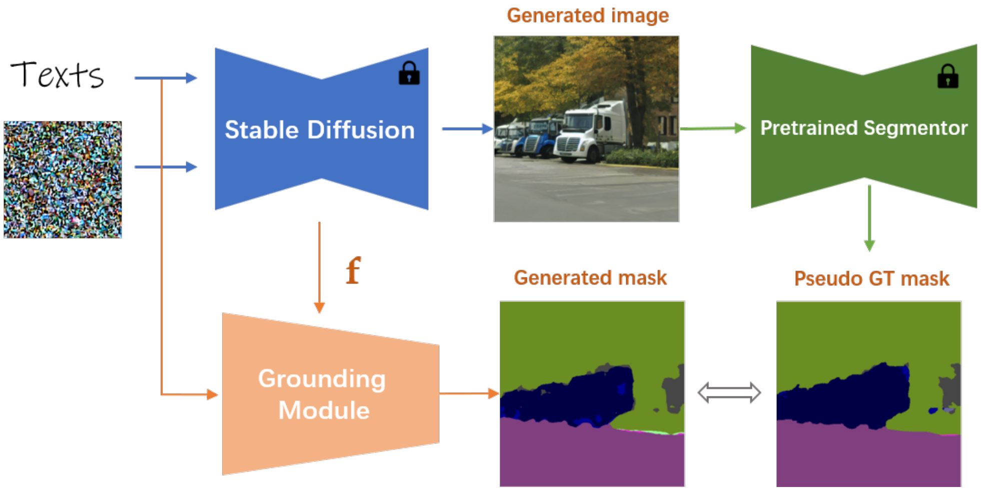


Figure 3.1: The overview of grounding-based generation. The image generation branch adopts the original SD model. Meanwhile, the label generation branch utilizes prompt embeddings and intermediate features extracted from the SD models as inputs to train a grounding module. This grounding module is supervised by pseudo-labels obtained from an off-the-shelf semantic segmentation model. A lock sign is used to indicate that the weights are frozen.

3.3.1 Grounding-based Generation

Generative models that are trained to synthesize highly realistic images usually rely on vast collections of datasets sourced from various fields, which implies that these models implicitly acquire semantic knowledge represented in their high-dimensional latent space. DatasetGAN [73] utilizes this concept by extracting the GAN's feature maps and then training a multilayer perceptron (MLP) classifier on top of these feature maps in a supervised manner. Similarly, pretrained diffusion models have been shown to provide excellent feature representations for semantic segmentation tasks [40, 75, 69, 21]. In light of this, we implement grounding-based image-label pair generation, based on the work of Li *et al.* [40].

The workflow of the grounding-based generation method is illustrated in Figure 3.1. For the image generation, we adopt the original SD to generate the image, with street scene-related text information and Gaussian noise as input. For the label generation, the extracted intermediate features of SD, along with the text embeddings, are used as inputs to train a grounding module that predicts the semantic labels.

To train the grounding module, we leverage an off-the-shelf model pretrained on the target domain to generate pseudo-ground truth labels. These pseudo-ground truth labels then supervise the grounding network during training. In essence, this approach treats the masks as an additional output of the generative model. This method attempts to harness the implicit semantic information present in the diffusion model's intermediate features, using them to generate corresponding segmentation masks for the generated images.

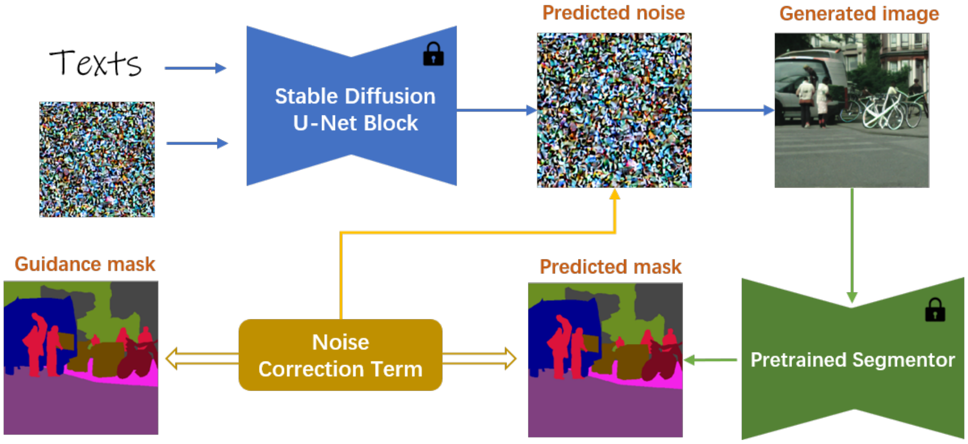


Figure 3.2: The overview of guidance-based generation. The guidance function calculates the noise correction term to influence the sampling process.

3.3.2 Guidance-based Generation

Several studies [3, 71, 70, 8] have developed training-free, guidance-based generation methods to control image generation. These methods exploit a unique feature of the diffusion model - its iterative denoising process. In the context of this work, we explore the guidance-based generation following the approach outlined by Bansal *et al*. [3].

As depicted in Figure 3.2, the guiding principle behind this approach is to modify the noise prediction during the sampling process and direct its trajectory based on the input mask. Specifically, at each forward sampling timestep, the predicted noise helps initially reconstruct a clean data point using [58]. Following this, an off-the-shelf auxiliary semantic segmentation network is subsequently applied to the reconstructed clean image to derive the predicted logits. To guide the generation, the guidance function cross-entropy loss between the predicted logits and the input guidance mask is computed and the gradient step of this calculated loss then serves as a corrective term for the original noise estimation.

This correction to the noise prediction allows us to guide the image generation process without needing to train the model. The corrected noise prediction is given by:

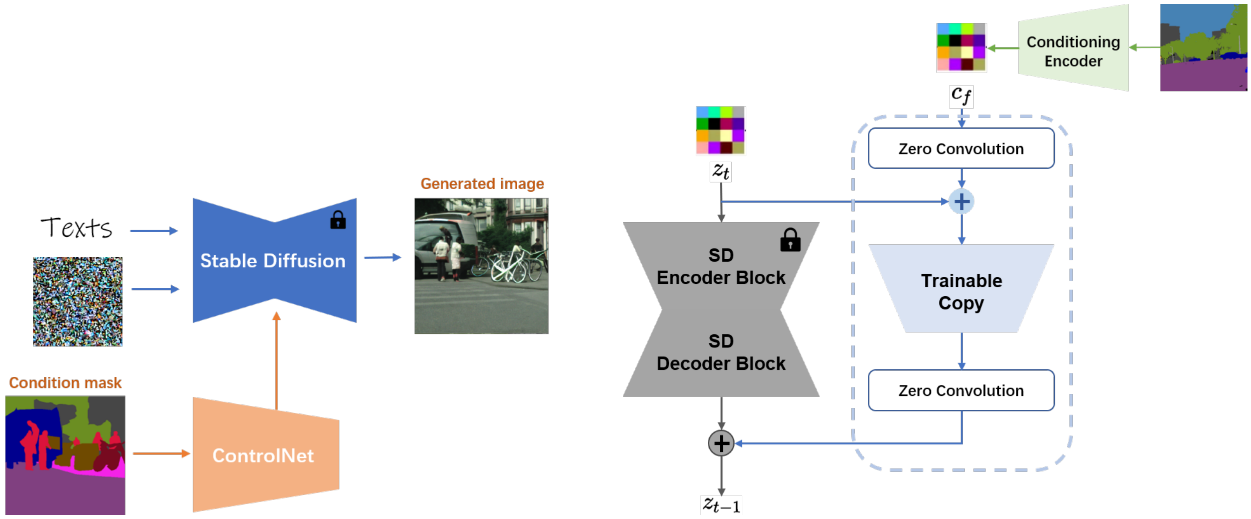
$$\hat{\epsilon}_\theta\left(z_t, t\right) = \epsilon_\theta\left(z_t, t\right) + m \cdot \sqrt{1-\alpha_t} \cdot \nabla_{z_t} \ell\left(c, f\left(\hat{z}_0\right)\right) \tag{3.2}$$

Here, $\sqrt{1-\alpha_t}$ controls guidance strength for each timestep t. The term c is the input guidance mask. ℓ and f represent the guidance function and the off-the-shelf segmentation network respectively.

Overall, this modification of the sampling process steers the image generation process to produce results that adhere to the semantics present in the input guidance mask. However, the guidance-based generation requires the use of existing masks, which is not available in the UDA setting. Despite this limitation, we still consider this approach worth exploring with the assumption that acquiring masks is a relatively easy task to achieve, and we will talk about the specific strategy used to deal with masks in section 3.4.

3.3.3 Condition-based Generation

The condition-based generation approach regards the mask as an additional input for the neural network blocks, allowing for more control over the image generation process, as demonstrated in Figure 3.3a. With this method, we generate an image-label pair dataset where the generated images adhere to the structure outlined by the conditioning masks. Works such as those by [72, 41] embed the conditioning information through lightweight adapters and inject the embedding into frozen pretrained SD models. Meanwhile, [33] trains a comprehensive diffusion model from scratch to achieve substantial controllability for both single and multiple conditions. We employ the ControlNet [72] in this study due to its powerful controllability and manageable computational requirements.



(a) The overview of condition-based generation.

(b) The architecture of ControlNet.

Figure 3.3: The condition-based generation. (a) The overview of condition-based generation. In this approach, an additional network is employed to encode the conditions and incorporate them into the SD model. The pre-trained SD model utilizes fixed parameters for generating natural images based on both the text condition and the additional conditions. Here we adopt ControlNet as the condition encoder network. (b) The overview of the architecture of ControlNet. It is comprised of replicated SD blocks (encoder and middle blocks) and zero convolutions."

Figure 3.3b shows the basic architecture of ControlNet, indicated by the dashed line. The input condition, which is represented as a one-hot encoding, is initially processed via a series of convolutional layers. This step ensures that the size of the latent code of the SD model matches the conditioning encoding. The conditioning encoding is subsequently passed through ControlNet. This network makes a direct copy of the original SD's encoder and middle blocks, retaining their weights as well. This feature speeds up the optimization process as it eliminates the need for training from scratch. These copied weights also make optimization faster because it is not trained from scratch. In addition, the copied network is wrapped by zero convolution layers, which consist of a 1×1 convolution layer with both weights and bias terms initialized as zeros. This ensures that the additional input, when injected into the original SD model, doesn't have any initial influence on the optimization process.

Like the guidance-based generation approach, masks are essential for the generation process in this method. The same assumption regarding masks holds in this context as well.

3.3.4 The Role of Text Prompts

SD model is originally a text-conditioned diffusion model, with image generation significantly reliant on text prompts. In the three approaches discussed above, different text prompts are utilized according to the varying controllability characteristics of each approach.

For both grounding-based and guidance-based generation, the generated image is directly guided by the text prompts. Thus, these prompts must be highly detailed and designed with care. In practice, we apply a prompt generator to create a variety of text descriptions utilizing pre-specified class names and their respective characteristics, drawing from domain knowledge. As an example, a class of interest, such as ”bus”, can be represented as either a singular noun ”a bus” or a plural noun ”a group of buses”, and can possess different states such as ”parking”, ”riding”, or ”waiting”. Moreover, it can be described to denote its relations with other classes, such as ”on the road”, ”near a person”, etc. Different qualifiers like colors, vehicle styles, and various weather conditions can also be utilized to diversify the resulting image. An example of a comprehensive text prompt could be: ”a photo of a yellow bus parking near a sidewalk, during intense traffic, in focus, unoccluded, centered, high quality, professional, very detailed”.

While in the condition-based generation approach, the direction of image generation is driven towards the input condition through a relatively high guidance scalar. This allows the generated image to identify objects from the structural guidance provided by the conditioning mask, thereby reducing the influence of text prompts, which can hardly provide structural guidance to image synthesis. This approach simplifies the process by eliminating the need for an artificial prompt generator, and instead, directly using a list of class names that are present in the input condition masks as text prompts. For instance, a comprehensive text prompt could be ”car, road, sky, rider, bicycle, vegetation, building”.

3.4 U-Net Swap Style Transfer

After a comparative review of the results from each method as detailed in section 4.3.1, we opted to use the conditional model ControlNet for generating labeled data, given its relatively superior accuracy. Next, we propose ways to modify ControlNet to further improve its controllability concerning style transfer, thereby aiding in data generation in the context of UDA.

3.4.1 DreamBooth Fine-tuning

DreamBooth [52] is designed to fine-tune text-to-image diffusion models for what’s known as a subject-driven generation, the process of creating new images of a specific subject within varied contexts. It utilizes a unique token to represent the subject (for example, ”A S^* dog”, with S^* standing for the unique token), and fine-tunes the pretrained diffusion models using a few subject images to bind the unique token identifier with the subject instance. To prevent the model from overfitting the subject instance on the subject class, it also introduces a class-specific prior preservation loss. This loss supervises the model using its own generated samples guided by a general

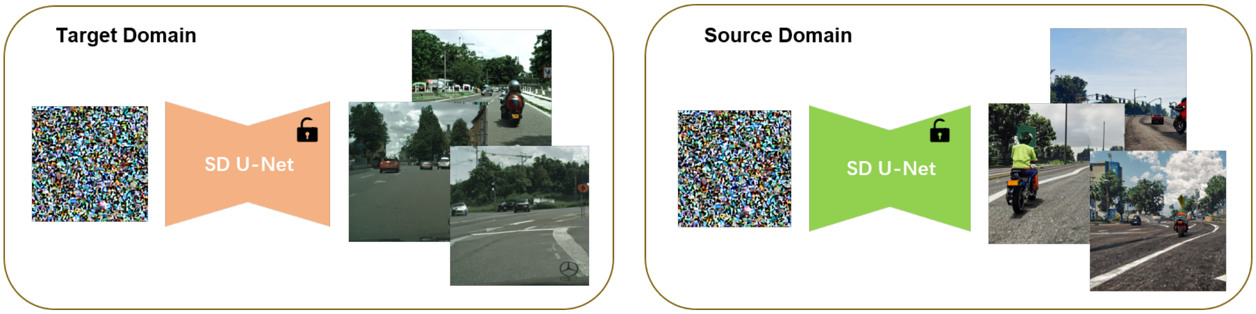


Figure 3.4: The DreamBooth fine-tuning on the target domain and the source domain.

class description that the subject instance belongs to (e.g., ”A dog”). This approach ensures the model learns about the subject instance while still preserving knowledge about the class prior, enabling it to produce a ”personalized” image that still maintains diversity.

Instead of requiring subject preservation, our work seeks to generate images that mimic the target distribution. Therefore, we utilize DreamBooth primarily as a general fine-tuning method to adjust our model using images from a specific domain. Specifically, we fine-tune two SD U-Nets within the source and target domains, which enables them to generate images in the styles of GTA5 and Cityscapes independently, as depicted in Figure 3.4. The objective here is to adapt the SD model from an initial domain into a specific one to capture the distinctive appearance of this particular domain. In doing so, the model should acquire the capacity to consistently produce images that embody the unique features of a given domain. It is worth noting that no masks are required in this process.

3.4.2 U-Net Swap

With two independently pretrained SD U-Nets, subsequently, we perform U-Net swap during the training phase and inference phase. The idea behind this is to allow the fine-tuned SD U-Net weights to govern the specific domain style, while ControlNet ensures that the generation adheres to the conditioning layout. This approach is illustrated in Figure 3.5.

Training Phase. During the training phase, we utilize the U-Net that has been fine-tuned on the source domain, with its weights frozen. The trainable ControlNet is then trained using the image-mask pairs dataset derived from the source domain. By doing this, we aim to have the ControlNet branch primarily focus on layout control, thereby placing less emphasis on appearance styling, as the left part of the SD U-Net is already generating images in the style of the source domain.

Inference Phase. During the inference phase, the trained ControlNet is frozen, while we replace the source domain’s SD U-Net with the one that has been trained on the target domain. This approach essentially allows us to maintain the semantic control that the ControlNet branch has learned from the source domain, while the interchangeable U-Nets facilitate the transfer of style to the target domain. As a result, when provided with a mask from the GTA dataset, our model can generate images that mimic the style of the Cityscapes dataset while adhering to the structural guidance from the source domain.

With DreamBooth fine-tuning and U-Net swap, we illustrate a complete process to achieve aligned and realistic labeled dataset generation. This approach eliminates the need for target labels, underscoring the potential of our method in assisting with complex UDA tasks.

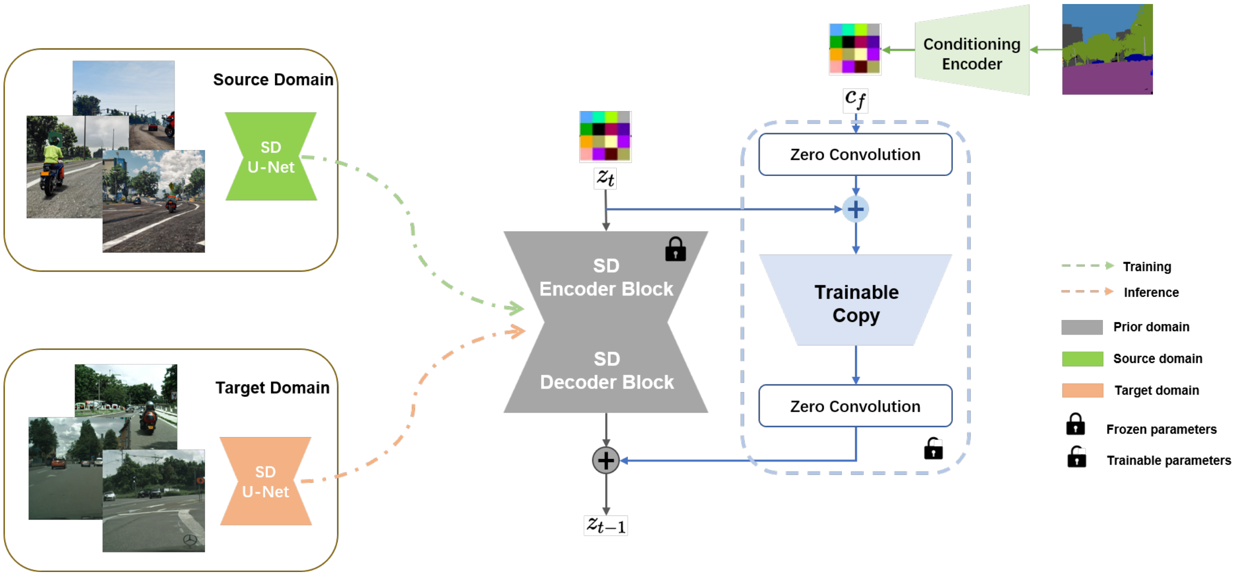


Figure 3.5: The overview of U-Net swap. Using a source domain pretrained SD U-Net during ControlNet training phase and a target-domain pretrained SD U-Net during inference phase.

3.5 Generation Quality Improvement

Like most datasets, the GTA dataset encounters the issue of imbalanced class distribution. This imbalance presents challenges for ControlNet training, as it may struggle to recognize rare classes such as trains and motorcycles. The proportion of pixels for each class in the GTA dataset is shown in Figure 3.6. Another challenge faced by the model is the accurate generation of small objects. Since the SD model operates in latent space, it can be particularly challenging to generate these small objects accurately. To mitigate these issues, we further apply Rare Class Sampling (RCS) [31] and Small Components Refinement (SCR) strategies to enhance the quality of image generation.

3.5.1 Rare Class Sampling (RCS)

The imbalanced class distributions in the dataset can cause difficulties, resulting in a biased model toward common classes. The Rare Class Sampling (RCS) technique is applied to address this issue by increasing the sampling probability for less frequent classes, thereby offering a more balanced training process.

To implement RCS, we begin by calculating the frequency f_c of each class c in the source dataset, based on the number of pixels that belong to a particular class c:

$$f_c = \frac{\sum_{i=1}^{N_S} \sum_{j=1}^{H \times W} \left[y_S^{(i,j,c)} \right]}{N_S \cdot H \cdot W} \tag{3.3}$$

Here, N_S represents the total number of images in the source dataset, and H and W denote the height and width of the images, respectively. The notation $y_S^{(i,j,c)}$ stands for the class label of the

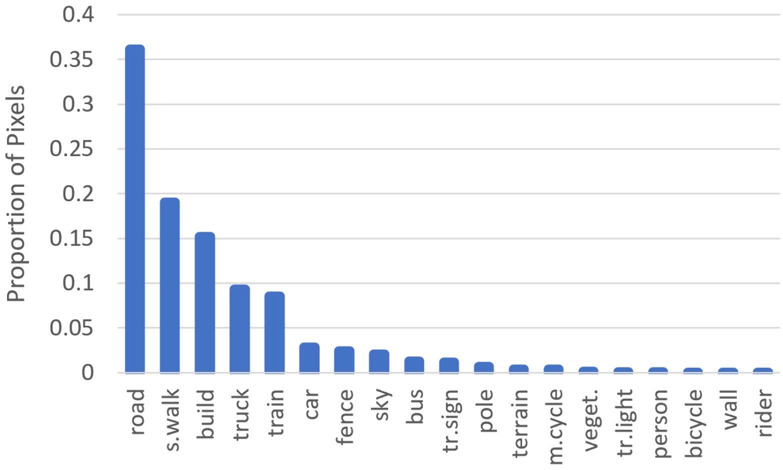


Figure 3.6: Class statistics of the GTA dataset.

pixel at position (i, j) in the image. The sampling probability $P(c)$ of a certain class c is then defined as a function of its frequency f_c:

$$P(c) = \frac{e^{(1-f_c)/T}}{\sum_{c'=1}^{C} e^{(1-f_{c'})/T}} \tag{3.4}$$

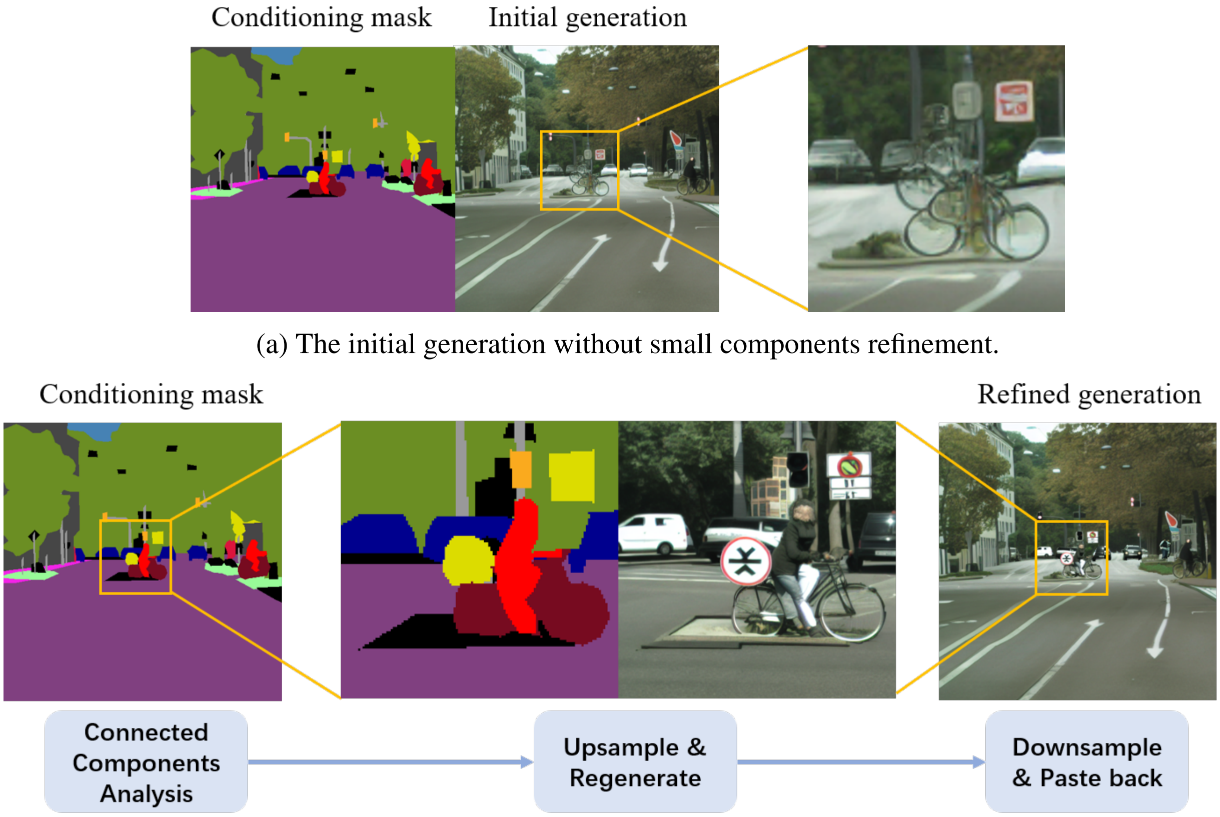
where C denotes the total number of classes and the hyperparameter T controls the smoothness of the distribution. A higher T leads to a more uniform distribution across classes, while a lower T results in a distribution that focuses more on the rare classes.

By applying the RCS technique, we increase the model's attention towards rare classes during training, thereby enhancing its performance on these particular classes.

3.5.2 Small Components Refinement (SCR)

The presence of small components in an image can pose challenges for the generation process. These components can be either overlooked or inaccurately generated, affecting the overall image quality. For example, as shown in Figure 3.7a, the model fails to generate distant objects like the rider and the round traffic sign in the first generation. To address this issue, we adopt a straightforward refinement strategy, and the process is illustrated in Figure 3.7b.

Firstly, we perform a connected components analysis on the input condition masks from the source domain. We then select those components that contain a number of pixel values lower than a certain threshold - these are what we consider 'small components'. Next, we take a small crop from the mask that includes the small components, and then directly upsample this cropped mask using a nearest neighbor resampling filter. The resulting enlarged conditioning mask is used as input to regenerate the image and offers better control over the previously overlooked objects. Upon regeneration of the small components, we downscale these regenerated components to match the original size of the image crop. We then integrate these refined parts into the initially generated image. The result of this process is a refined image that offers a higher degree of alignment between the input mask and the generated image. This method allows us to more accurately generate and represent small components within the final image, enhancing the overall quality and realism of our generation process.



(a) The initial generation without small components refinement.

(b) The small components refinement process and the refined generation result.

Figure 3.7: The overview of small components refinement (SCR) strategy.

Chapter 4

Experiments and Results

In this chapter, we first provide an introduction to the experimental setup, detailing our data sources and the metrics used. We then elaborate on the specific implementations used for the various components of our process. In the results section, we draw comparisons between multiple labeled data generation methods within a single domain setting and also discuss the performance of our downstream segmentation task within this context. Further, we present experiments designed to validate the advantages of our proposed methods in the UDA setting. We also carry out an ablation study to further evaluate the effectiveness of our proposed strategies.

4.1 Experimental Setup

4.1.1 In-domain Experiments

To evaluate the capabilities of diffusion models in generating labeled data, our initial experiments are conducted in-domain. This means that we have complete access to images and labels from the target domain (Cityscapes), which are utilized for training the diffusion models. We compare all three labeled data generation methods in this context, choosing the most promising for subsequent cross-domain experiments.

We additionally conduct a downstream semantic segmentation task using the most promising generation approach. This provides valuable insights into the extent to which our generated dataset can accurately mimic the target distribution.

4.1.2 Cross-Domain Experiments

In order to verify the validity of our proposed method in aiding the domain adaptation of semantic segmentation models, we carry out experiments in a cross-domain (UDA) setting. In this scenario, we use labels from the source domain (GTA5) to generate images that resemble those from the target domain (Cityscapes). These generated images and corresponding labels are then used to train a segmentation model from scratch. We evaluate the performance of the trained model on the validation dataset from the target domain to assess the effectiveness of our method.

4.1.3 Datasets and Metrics

We utilize two datasets in our experiments. The target dataset, **Cityscapes** [13], comprises 2,975 training images and 500 validation images. These images, captured from a car in urban settings, have a resolution of 2048x1024 and are labeled across 19 classes. On the other hand, the source dataset, **GTA5** [48], includes 24,966 synthetic images with a resolution of 1914x1052. The classes in the GTA5 dataset are matched to those in the Cityscapes dataset.

For performance evaluation, we report the Intersection over Union (IoU) for each class and the mean Intersection over Union (mIoU) over all classes- these are standard metrics in semantic segmentation tasks. For the labeled data generation methods, we calculate the mIoU between the image-label pairs within the generated dataset. For downstream semantic segmentation tasks, the mIoU is computed between the labels predicted by the model (which has been trained on the generated dataset) and the ground truth labels on the target validation dataset.

4.2 Implementation Details

Here, we present the general setting utilized throughout all steps of our study. Unless otherwise specified, all the pre-trained SD models used in our experiments are the RunwayML-v1-5 version. Furthermore, all the generated images maintain a resolution of 512×512 pixels. In terms of the reverse denoising process of diffusion models, we use 20 denoising steps. More denoising steps usually lead to a higher quality image at the expense of slower inference, yet we find that there is no discernible difference in visual results between 50 and 20 denoising steps, so we adopt a smaller number of steps for faster inference.

4.2.1 In-domain Labeled Image Generation

Grounding-based generation

The architecture of the grounding module is based on the approach proposed by Li *et al.*'s work [40]. However, unlike the original study that outputs single-class masks and employs a binary cross-entropy loss, we have extended the output channels to accommodate 19 classes from Cityscapes and have implemented a multi-class cross-entropy loss. The pseudo ground truth masks are obtained from an off-the-shelf model, SegFormer, which has been pre-trained solely on the Cityscapes dataset, and we benefit from the implementation from the work [31]. We train the grounding module on an RTX 2080 Ti for 10 epochs, which consumes approximately 20 GPU hours. The initial learning rate is set to 1e-4 and the weight decay is 1e-4.

During inference time, We generate 1000 images using a hand-designed prompt generator (as discussed in Section 3.3.4) as a training dataset.

Guidance-based generation

The implementation is based on the work [3]. The off-the-shelf model and the approach used for text prompts generation are the same as the ones used for grounding-based generation. We set the guidance scalar to m=400. The number of DDIM sampling steps used is 500, and the iterations are repeated 10 times. In practice, we only experimented with a single class of cars in this method, as the optimization process is relatively costly, taking 90 minutes.

Condition-based generation

We employ ControlNet [72] as the additional SD adapter for the condition-based generation. It replicates 12 encoder blocks and 1 middle block from the original SD model. Each encoder block comprises downsampling convolutional layers, ResNet layers, and Vision Transformers (ViTs). The ViTs incorporate multiple cross-attention and/or self-attention mechanisms. The text prompt, described as a list of class names that appear in the mask, is encoded using OpenAI's CLIP [45]. The diffusion time steps are encoded via positional encoding.

SD works in latent space, which requires ControlNet to adjust the image-based conditions to match the convolution size. To achieve this, we utilize a tiny conditioning embedding network to convert the one-hot encoded condition (of size 512×512×20, where 20 signifies 19 classes from Cityscapes and one background class) into a 64×64×4 feature space.

We train an oracle model using Cityscapes training images and labels, randomly cropped to a resolution of 512. The model is trained with a learning rate of 1e-5 and a batch size of 8 on an RTX 3090 GPU for one day. During inference, the DDIM scheduler uses 20 sampling steps, and we use labels from the Cityscapes training dataset as input conditions to create our generated dataset.

4.2.2 Cross-domain Labeled Image Generation

DreamBooth Fine-tuning

We fine-tune two separate SD models using the Cityscapes training dataset and the full GTA5 dataset. These models are trained with images that have been randomly resized and cropped to a resolution of 512. We employ a constant learning rate scheduler with a learning rate of 2e-6. The number of training iterations is set to 4500 for the target domain fine-tuning and 10000 for the source domain fine-tuning. We train on an RTX 3090 GPU for around 2 hours. Unlike the methodology presented in [52], we keep the text encoder frozen in our experiments.

U-Net Swap for ControlNet

In this cross-domain experiment, we train ControlNet on the source domain, namely full GTA5 images and labels, with the same hyperparameters that were used in the oracle model training mentioned in section 4.2.1. However, here we replace the original SD model with the one that has been fine-tuned on the source domain.

For the generation of images, we make another swap in the SD model, this time substituting it with the one fine-tuned on the target domain. Using masks derived from the GTA5 dataset as input, we proceed to generate 3000 images that closely emulate the visual style of Cityscapes.

We incorporate the Rare Class Sampling (RCS) method during both training and generation stages, setting a temperature scalar of 0.01. As for the Small Components Refinement (SCR) strategy, we employ a minimum area threshold of 3000 to select small components. The refinement process can be repeated up to a maximum of 20 times to ensure better alignment with the input mask.

4.2.3 Semantic Segmentation Model Evaluation

We validate the generated datasets by utilizing them in downstream semantic segmentation tasks. More specifically, we train a semantic segmentation model, SegFormer [68], in both in-domain and cross-domain scenarios. For both of these settings, the segmentation model is trained using the generated datasets and subsequently validated on the real-world Cityscapes validation dataset. The training process is performed on an RTX 2080 TI GPU and consists of 40,000 iterations with a batch size of 2. This process takes approximately 7 hours to complete.

For the validation phase, we resize the Cityscapes validation dataset to a resolution of 1024x512. The inference process is carried out in a sliding window fashion, where we utilize patches of size 512 with a stride of 256.

4.3 Results

4.3.1 In-domain Experiments

Labeled Image Generation

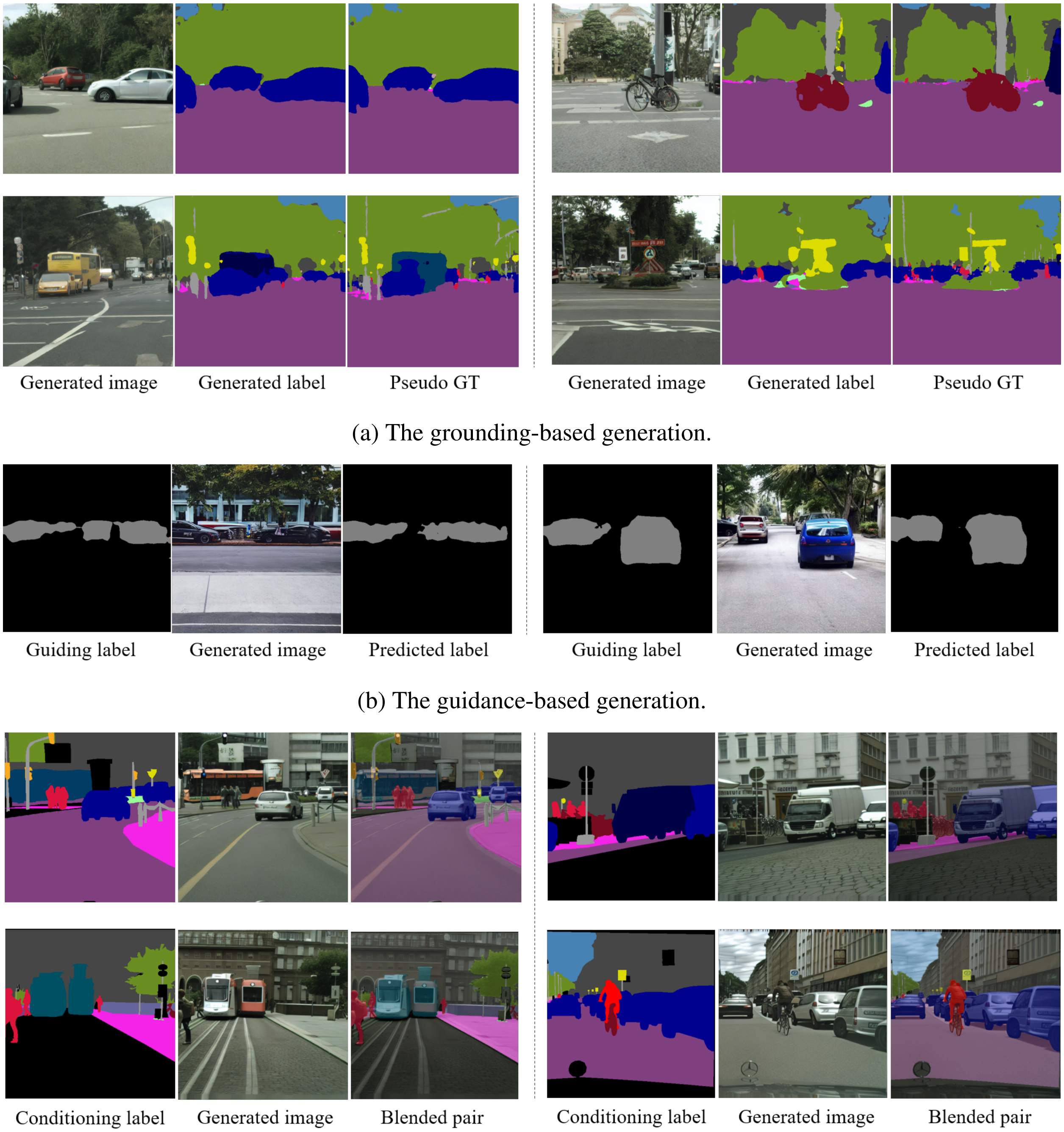
In this section, we evaluate three different methods for generating labeled images within the domain, both qualitatively and quantitatively. In addition, we investigate the strengths and weaknesses of each of these methods.

The qualitative results of the three approaches are shown in Figure 4.1.

Grounding-based Generation In Figure 4.1a, the leftmost two columns display our generated image and label pairs based on the grounding-based generation approach, while the third column displays the pseudo-ground-truth masks produced by the off-the-shelf model. The first row of images shows that this method is good at generating high-quality images and corresponding masks for common classes and large objects. The presence of cars, roads, and trees in these images suggests that the model can reliably handle these categories and generate realistic representations of them. However, as we move to the more challenging categories, such as shown in the second row, both our generative model and the oracle model start to exhibit difficulties. For instance, the oracle model can struggle with differentiating between the car and bus classes. These challenges are naturally passed down to the generative model, which bases its output on the guidance of the oracle model. Additionally, the generated image is solely guided by texts, which can hardly provide structure guidance. Consequently, this leads to uncontrollable and unstable generation results.

Guidance-based Generation As for guidance-based generation, we only conducted tests on a single common class car, as illustrated in Figure 4.1b. The input label used to guide the image generation is shown in the first column, while the second and third columns display the generated result and the predicted label by an off-the-shelf segmentation model, respectively. The generated image can produce classes that approximately correspond to the guiding mask. However, in general, the accuracy is not satisfying. Furthermore, it took approximately 90 minutes to sample such an image; thus, we refrained from conducting further experiments in this study.

Condition-based Generation As shown in Figure 4.1c, the left two columns present the input conditioning mask and the generated image respectively. In the third column, we have blended the image and the mask to enhance visualization. The condition-based generation exhibits a substantial improvement in the alignment between the generated images and masks, almost achieving



(a) The grounding-based generation.

(b) The guidance-based generation.

(c) The condition-based generation.

Figure 4.1: Qualitative results for labeled data generation.

pixel-level accuracy. This highlights the superior precision of the condition-based approach when compared to the guidance-based or grounding-based methods. Notably, the model can also distinguish between similar classes like trucks and cars, as shown in the first row. Furthermore, the condition-based generation method shows a significant advantage in terms of controllability, thus making it a promising solution for labeled data generation and subsequent research.

Quantitative Comparison To further quantify the results, we use the pretrained segmentation model SegFormer to check the alignment between the image and label pairs in our generated dataset. In this process, we use the segmentation model to infer the generated images to obtain the predicted masks, and then calculate the mIoU between the predicted masks and the generated/conditioning labels in the dataset. Given the inefficiency of the guidance-based methods, we limit our comparison to the grounding-based and condition-based generation methods. Additionally, we draw a comparison with a concurrent work [17] based on a GAN framework, providing a more comprehensive evaluation of our approach. The results are shown in Table 4.1, which further showcases the superior alignment and high level of control of condition-based generation over other approaches. Therefore, in the following sections, we will focus on the condition-based generation for our subsequent analyses.

	Grounding-based generation	Condition-based generation (ours)	PairSIS-GAN [17]
mIoU↑	50.53	**60.37**	40.6

Table 4.1: Quantitative results for the labeled image generation.

Semantic Segmentation Evaluation We train ControlNet with the labeled Cityscapes training dataset and utilize Cityscapes masks that are randomly cropped from the Cityscapes training dataset to generate 6,000 images. We subsequently compared the performance of a semantic segmentation model, which is trained on our generated dataset, to the performance of models trained on the real Cityscapes training dataset and the GTA5 dataset. The results are presented in Table 4.2. Though the performance of our model does not match that of the oracle model, we observe a substantial advantage in performance over the model trained on the GTA5 dataset. This suggests that our generated dataset has imitated the distribution of the target domain to a considerable degree, showing a promising direction for the UDA setting.

	GTA→Cityscapes	Gen→Cityscapes	Cityscapes→Cityscaps (oracle)
mIoU↑	47.67	**64.97**	76.81

Table 4.2: In-domain experiments results. *→*: the dataset to the left of the arrow indicates the dataset on which the model is trained exclusively, while the dataset to the right of the arrow indicates the dataset on which the model is evaluated.

4.3.2 Cross-domain Experiments

In the UDA scenario, given conditioning labels from the GTA5 dataset, we generate Cityscapes-like images. we present qualitative results to illustrate the visual alignment and realism of our

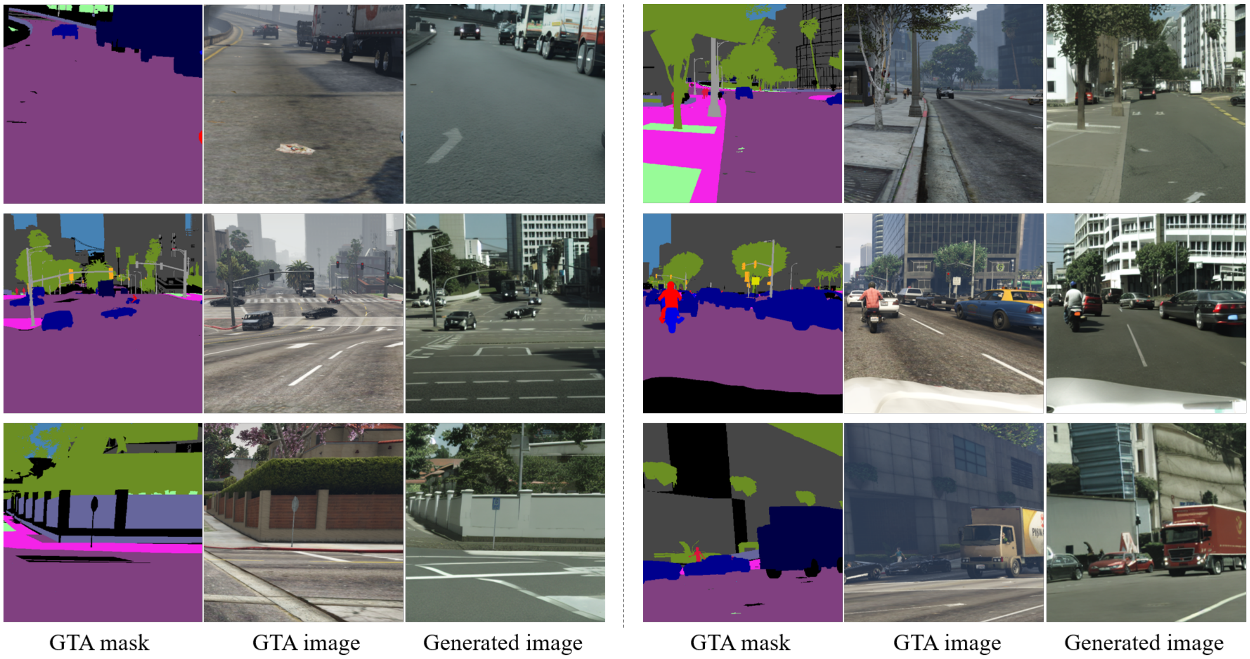


Figure 4.2: The generation results by conducting the U-Net swap. The paired GTA masks and images are only used for training, while during inference time, given GTA masks as guidance, we could generate Cityscapes-like images.

generated images, and also conduct quantitative analysis by evaluating the performance of models trained on our generated datasets in downstream tasks.

Qualitative results Our generated image-label pairs are shown in Figure 4.2. The leftmost column displays the conditioning label cropped from the GTA5 dataset. The middle column shows the original corresponding GTA5 image, which is used during the training of ControlNet. The right columns present the generated image, which adheres to the structural control of the source domain, while also transferring the style to mimic the target domain.

Semantic Segmentation Evaluation In order to validate the effectiveness of our generated dataset in the unsupervised domain adaptation (UDA) setting, we train a semantic segmentation model on these datasets. We then compare its performance with a model trained on the synthetic GTA5 dataset. The results of this comparison are illustrated in Table 4.3. Our model exhibits an improvement of 3.4 % in performance over the model trained on the synthetic GTA5 dataset, which underscores the value and effectiveness of our generated dataset.

	Road	S.walk	Build.	Wall	Fence	Pole	Tr.Light	Sign	Veget.	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	M.bike	Bike	mIoU↑
GTA5	65.0	19.8	85.5	41.0	35.2	31.1	44.9	26.5	87.3	38.9	88.0	63.0	20.3	82.4	52.0	54.6	12.0	33.5	24.8	47.7
Ours	**93.0**	**57.3**	**86.2**	**50.2**	29.2	28.0	31.5	**29.1**	86.8	**42.8**	82.7	**65.2**	**34.6**	**87.7**	42.3	50.9	1.8	**37.8**	**33.1**	**51.1**

Table 4.3: Cross-domain semantic segmentation evaluation results. As can be seen, the model trained on our generated dataset outperforms the one trained on GTA5 across multiple classes and achieves a higher overall mIoU.

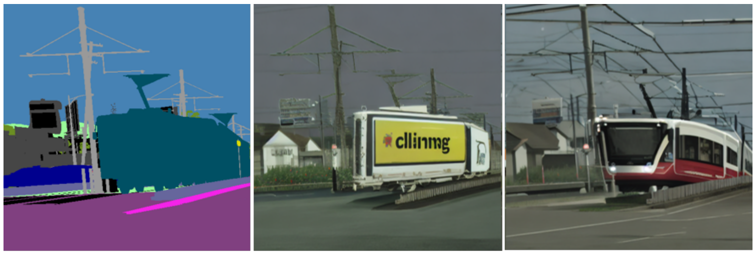


Figure 4.3: The generation results when utilizing RCS.

By generating images that bridge the domain gap, diffusion models have proven their utility for semantic segmentation tasks, and provide an innovative perspective to tackle the challenges of unsupervised domain adaptation. Rather than adhering to traditional UDA methodologies, these results demonstrate the potential for novel approaches that utilize the capabilities of diffusion models in generating meaningful, high-quality synthetic data that closely mimic the characteristics of the target domain.

4.3.3 Ablation Study

We investigate the efficacy of the Rare Class Sampling (RCS) and Small Class Refinement (SCR) strategies within a cross-domain setting.

RCS enables the model to encounter rare classes earlier and more frequently, thereby reducing the model's bias. An example of this is depicted in Figure 4.3. In the absence of RCS, the model is unable to accurately recognize the class 'train'. However, with the application of RCS, the model's representation of the class, train, significantly improves. The utility of RCS is also reflected quantitatively in Table 4.4, where it is seen to enhance the model's performance by an increase of +3.8 mIoU.

We also confirm the effectiveness of the SCR strategy. Through refined generation, we provide a more accurate condition, which helps reduce confusion for the model. The application of this strategy leads to a further improvement in the mIoU score by 3.5, resulting in a final score of 51.1 when both these strategies are employed.

Methods		mIoU↑
RCS	SCR	(cross-domain)
✘	✘	43.8
✔	✘	47.6
✔	✔	**51.1**

Table 4.4: Ablation study on RCS and SCR.

4.3.4 Training with Traditional Domain Adaptation Methods

To investigate if traditional domain adaptation (DA) methods can help with our generated dataset, we train the SOTA UDA method DAFormer [31] with our generated dataset. Additionally, we compare the performance of the model trained on GTA5 with DAFormer as well. The results are displayed in Table 4.5.

	Road	S.walk	Build.	Wall	Fence	Pole	Tr.Light	Sign	Veget.	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	M.bike	Bike	mIoU↑
Gen(w/o. DAFormer)	93.0	57.3	86.2	50.2	29.2	28.0	31.5	29.1	86.8	42.8	82.7	65.2	34.6	87.7	42.3	50.9	1.8	37.8	33.1	51.1
Gen(w/. DAFormer)	96.0	72.1	88.1	55.8	38.2	40.2	38.8	41.3	87.6	44.9	86.8	67.5	42.0	90.1	63.8	61.7	8.9	53.6	60.7	59.9
GTA5(w/. DAFormer)	95.7	70.2	89.4	53.5	48.1	49.6	55.8	59.4	89.9	47.9	92.5	72.2	44.7	92.3	74.5	37.71	65.1	55.9	61.8	68.3

Table 4.5: The performance with DAFormer trained on the generated dataset.

We observe that traditional domain adaptation techniques can help with our generated dataset for each of the classes. However, when compared with GTA5 to Cityscapes using DAFormer, trained on the DAformer, our dataset hasn't achieved comparable results. The potential reason for this could be as follows: while we generate the same number of training images as GTA5, our images have a lower resolution of 512×512. Additionally, since we also apply RCS for dataset generation, some images with rare classes may be sampled multiple times, while some images with common classes may not be sampled at all. These factors could result in the model utilizing relatively less information compared to the GTA5 dataset.

Chapter 5

Discussion

5.1 Trade-off between Style and Alignment

Throughout the ControlNet oracle model's training phase, we notice that regardless of the pretrained weights of the SD backbone model (whether on the prior domain or the target domain), the ControlNet branch quickly absorbs the style of the training images. Moreover, this ability to mimic the image's style becomes even more potent as the alignment improves during training. This can be challenging for us since we aim to separate the structure from the style. Consequently, we have decided to investigate a simplified conditioning encoder branch. Instead of fine-tuning the existing deep encoder (the copied blocks), our aim is to train a lightweight encoder, with the hope of using fewer parameters to limit its ability to capture the style. As shown in Figure 5.1, we have developed a lightweight version of ControlNet by exclusively training the zero convolution layers indicated by the orange dashed line. The rest of the model's weights are retained as frozen copies from the full version of ControlNet.

The qualitative differences are depicted in Figure 5.2. Regarding the image style, we observe that the lightweight version exhibits fewer artifacts and appears more natural, resembling real-world images. However, the lightweight encoder's capability to recognize contents in input control maps is comparatively less satisfying. As highlighted in the red box, the full deep encoder outperforms in producing aligned content.

In our project, we believe that aligned image-label pairs are more crucial for model training in semantic segmentation tasks than styles. Consequently, we have chosen to adhere to the full version of ControlNet.

5.2 Limitations and Future Work

In this section, we address the limitations of our research and outline potential directions for future work.

Prompt Engineering Prompt engineering has always been a crucial aspect in text-to-image generation. In this thesis, we utilized a list of class names as prompts for ControlNet training, which worked well. However, we acknowledge that the full capability of the language model might not have been fully explored. One promising avenue for further exploration is to design more diverse

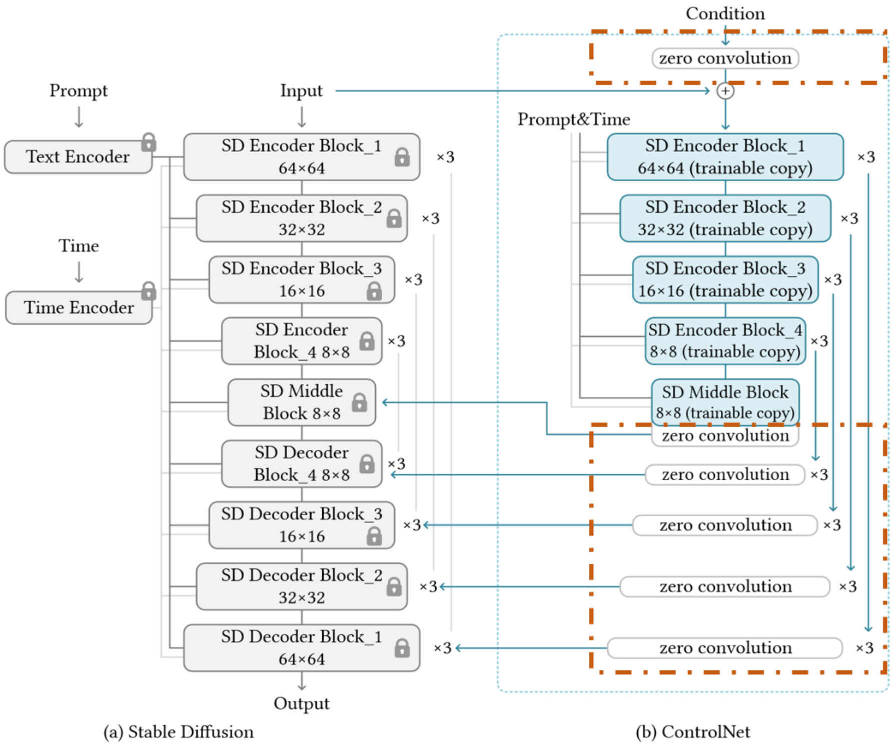


Figure 5.1: The architecture of ControlNet. Besides training the full ControlNet, we also train a lightweight version with only zero convolutions being trainable, as indicated by the orange dashed line. The figure is taken from [72].

prompts that can provide additional information, such as adverse weather conditions. This could potentially enhance the image generation process and lead to more robust and contextually relevant outputs.

Domain Generalization In this work, we limit our examination to the Cityscapes dataset. It would indeed be worthwhile to delve into the model's domain generalization ability by applying it to other real-world datasets.

Few-shot Condition Our experiments are conducted in an unsupervised setting where no target labels are available. However, in a practical scenario, it could be beneficial to annotate a handful of labels within the training dataset to enhance the performance of the model. As indicated in Figure 5.3, even 10% of the training data can already yield a relatively high mIoU score. This suggests that it could be worthwhile to experiment with scenarios involving few-shot learning, which can potentially lead to substantial performance improvement even with limited data. This remains a promising area to explore in future work.

Style and Content Disentangle While our method of interchanging two parallel U-Nets for the source and target domains has achieved a certain degree of style transfer, the disentanglement of style and content hasn't been maximized. We attribute this limitation to the absence of explicit separation and the lack of joint training for these two elements. One potential solution to address

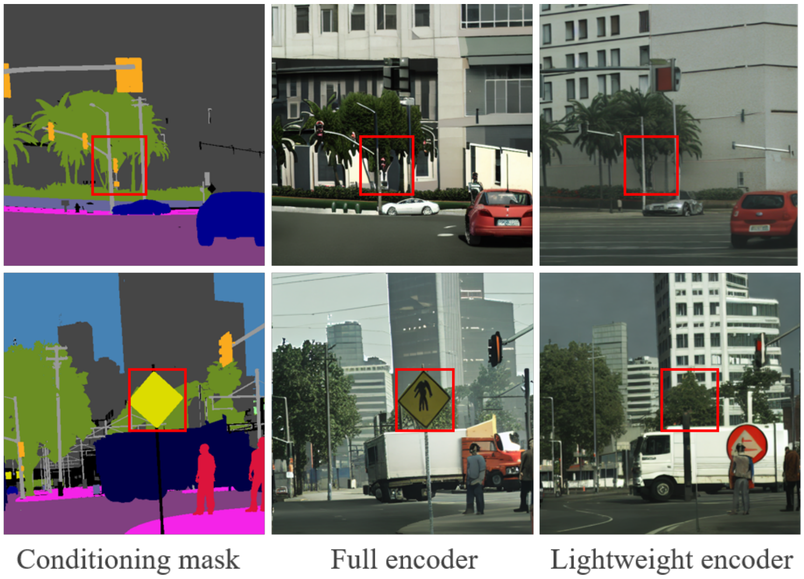


Figure 5.2: Comparison with lightweight version of ControlNet.

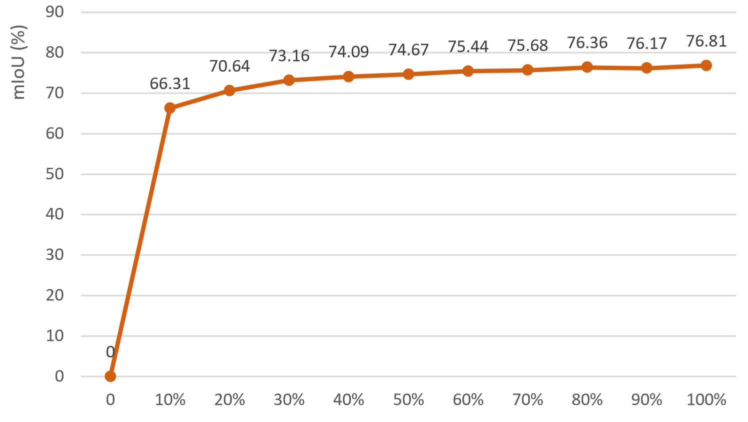


Figure 5.3: The oracle model performance in relation to the portion of training data.

this issue would be to explicitly represent these two elements and train local structure and global style in a joint manner. By incorporating such an approach, we believe that we can enhance the disentanglement of style and content, leading to more effective and accurate style transfer results.

Chapter 6

Conclusion

In this thesis, we aim to investigate the generation of a realistic and aligned labeled dataset for semantic segmentation by employing conditional text-to-image diffusion models. Additionally, we conduct a closed-loop evaluation of our generated dataset in the context of unsupervised domain adaptation.

Our initial efforts are centered on the target domain, examining the ability of diffusion models to synthesize labeled data. We delve into three distinct methods for handling semantic segmentation labels: grounding-based generation, condition-based generation, and guidance-based generation. It's found that the condition-based generation technique yields the best generation quality, and consequently, we further tailor this method for an unsupervised domain adaptation scenario. In the UDA setting, we implement the U-Net swap technique, which utilizes DreamBooth to fine-tune two domain-specific SD U-Nets and swaps them during the training and inference phase. With two additional proposed strategies, we achieve a reasonable image fidelity and alignment for labeled data generation. We validate our generated dataset, particularly in the UDA setting. Here, our dataset exceeds the performance of the GTA5 dataset, effectively highlighting its value for unsupervised semantic segmentation domain adaptation.

Bibliography

[1] Nikita Araslanov and Stefan Roth. Self-supervised augmentation consistency for adapting semantic segmentation, 2021.

[2] Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J. Fleet. Synthetic data from diffusion models improves imagenet classification, 2023.

[3] Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Soumyadip Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models, 2023.

[4] Dina Bashkirova, Jose Lezama, Kihyuk Sohn, Kate Saenko, and Irfan Essa. Masksketch: Unpaired structure-guided masked image generation, 2023.

[5] Manuel Brack, Felix Friedrich, Dominik Hintersdorf, Lukas Struppek, Patrick Schramowski, and Kristian Kersting. Sega: Instructing diffusion using semantic dimensions, 2023.

[6] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis, 2019.

[7] Dar-Yen Chen. Artfusion: Controllable arbitrary style transfer using dual conditional latent diffusion models, 2023.

[8] Minghao Chen, Iro Laina, and Andrea Vedaldi. Training-free layout control with cross-attention guidance, 2023.

[9] Mu Chen, Zhedong Zheng, Yi Yang, and Tat-Seng Chua. Pipa: Pixel- and patch-wise self-supervised learning for domain adaptative semantic segmentation, 2022.

[10] Weifeng Chen, Jie Wu, Pan Xie, Hefeng Wu, Jiashi Li, Xin Xia, Xuefeng Xiao, and Liang Lin. Control-a-video: Controllable text-to-video generation with diffusion models, 2023.

[11] Jaehoon Choi, Taekyung Kim, and Changick Kim. Self-ensembling with gan-based data augmentation for domain adaptation in semantic segmentation, 2019.

[12] Ernie Chu, Shuo-Yen Lin, and Jun-Cheng Chen. Video controlnet: Towards temporally consistent synthetic-to-real video translation using conditional image diffusion models, 2023.

[13] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[15] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021.

[16] Lisa Dunlap, Alyssa Umino, Han Zhang, Jiezhi Yang, Joseph E. Gonzalez, and Trevor Darrell. Diversify your vision datasets with automatic diffusion-based augmentation, 2023.

[17] George Eskandar, Diandian Guo, Karim Guirguis, and Bin Yang. Towards pragmatic semantic image synthesis for urban scenes, 2023.

[18] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion, 2022.

[19] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. A neural algorithm of artistic style, 2015.

[20] Vidit Goel, Elia Peruzzo, Yifan Jiang, Dejia Xu, Nicu Sebe, Trevor Darrell, Zhangyang Wang, and Humphrey Shi. Pair-diffusion: Object-level image editing with structure-and-appearance paired diffusion models, 2023.

[21] Rui Gong, Martin Danelljan, Han Sun, Julio Delgado Mangas, and Luc Van Gool. Prompting diffusion representations for cross-domain semantic segmentation, 2023.

[22] Rui Gong, Wen Li, Yuhua Chen, and Luc Van Gool. Dlow: Domain flow for adaptation and generalization, 2019.

[23] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[24] Cusuh Ham, James Hays, Jingwan Lu, Krishna Kumar Singh, Zhifei Zhang, and Tobias Hinz. Modulating pretrained diffusion models for multimodal image synthesis, 2023.

[25] Ligong Han, Yinxiao Li, Han Zhang, Peyman Milanfar, Dimitris Metaxas, and Feng Yang. Svdiff: Compact parameter space for diffusion fine-tuning, 2023.

[26] Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition?, 2023.

[27] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control, 2022.

[28] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[29] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022.

[30] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Hrda: Context-aware high-resolution domain-adaptive semantic segmentation, 2022.

[31] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9924–9935, 2022.

[32] Lukas Hoyer, Dengxin Dai, Haoran Wang, and Luc Van Gool. Mic: Masked image consistency for context-enhanced domain adaptation, 2023.

[33] Lianghua Huang, Di Chen, Yu Liu, Yujun Shen, Deli Zhao, and Jingren Zhou. Composer: Creative and controllable image synthesis with composable conditions, 2023.

[34] Nisha Huang, Yuxin Zhang, and Weiming Dong. Style-a-video: Agile diffusion for arbitrary text-based video style transfer, 2023.

[35] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models, 2023.

[36] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022.

[37] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion, 2023.

[38] Gihyun Kwon and Jong Chul Ye. Diffusion-based image translation using disentangled style and content representation, 2023.

[39] Zheng Li, Yuxuan Li, Penghai Zhao, Renjie Song, Xiang Li, and Jian Yang. Is synthetic data from diffusion models ready for knowledge distillation?, 2023.

[40] Ziyi Li, Qinye Zhou, Xiaoyun Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. Guiding text-to-image diffusion model towards grounded generation. *arXiv preprint arXiv:2301.05221*, 2023.

[41] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models, 2023.

[42] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE international conference on computer vision*, pages 4990–4999, 2017.

[43] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models, 2022.

[44] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. Classmix: Segmentation-based data augmentation for semi-supervised learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1369–1378, 2021.

[45] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[46] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022.

[47] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows, 2016.

[48] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 102–118. Springer, 2016.

[49] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[50] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[51] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3234–3243, 2016.

[52] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation, 2023.

[53] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022.

[54] Mert Bulent Sariyildiz, Karteek Alahari, Diane Larlus, and Yannis Kalantidis. Fake it till you make it: Learning transferable representations from synthetic imagenet clones, 2023.

[55] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022.

[56] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.

[57] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics, 2015.

[58] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022.

[59] Han Sun, Rui Gong, Konrad Schindler, and Luc Van Gool. Sf-fsda: Source-free few-shot domain adaptive object detection with efficient labeled data factory, 2023.

[60] Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models, 2023.

[61] Wilhelm Tranheden, Viktor Olsson, Juliano Pinto, and Lennart Svensson. Dacs: Domain adaptation via cross-domain mixed sampling. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1379–1389, 2021.

[62] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker. Learning to adapt structured output space for semantic segmentation, 2020.

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

[64] Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space, 2021.

[65] Roy Voetman, Maya Aghaei, and Klaas Dijkstra. The big data myth: Using diffusion models for dataset generation to train deep detection models, 2023.

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

[67] Qin Wang, Dengxin Dai, Lukas Hoyer, Luc Van Gool, and Olga Fink. Domain adaptive semantic segmentation with self-supervised depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8515–8525, 2021.

[68] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers, 2021.

[69] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models, 2023.

[70] Han Xue, Zhiwu Huang, Qianru Sun, Li Song, and Wenjun Zhang. Freestyle layout-to-image synthesis, 2023.

[71] Jiwen Yu, Yinhuai Wang, Chen Zhao, Bernard Ghanem, and Jian Zhang. Freedom: Training-free energy-guided conditional diffusion model, 2023.

[72] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023.

[73] Yuxuan Zhang, Huan Ling, Jun Gao, Kangxue Yin, Jean-Francois Lafleche, Adela Barriuso, Antonio Torralba, and Sanja Fidler. Datasetgan: Efficient labeled data factory with minimal human effort, 2021.

[74] Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K. Wong. Uni-controlnet: All-in-one control to text-to-image diffusion models, 2023.

[75] Wenliang Zhao, Yongming Rao, Zuyan Liu, Benlin Liu, Jie Zhou, and Jiwen Lu. Unleashing text-to-image diffusion models for visual perception, 2023.

[76] Qianyu Zhou, Zhengyang Feng, Qiqi Gu, Jiangmiao Pang, Guangliang Cheng, Xuequan Lu, Jianping Shi, and Lizhuang Ma. Context-aware mixup for domain adaptive semantic segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(2):804–817, 2022.

[77] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European conference on computer vision (ECCV)*, pages 289–305, 2018.

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

Towards Exploiting the Generative Image Diffusion Models for Unsupervised Semantic Segmentation Domain Adaptation

Authored by (in block letters):

For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**
Jia	Yuru

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date	**Signature(s)**
Zurich, 31st July 2023	Yuru Jia

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.