
SemFlow: Binding Semantic Segmentation and Image Synthesis via Rectified Flow

Chaoyang Wang¹ Xiangtai Li¹ Lu Qi² Henghui Ding³
Yunhai Tong¹ Ming-Hsuan Yang²

¹School of Intelligence Science and Technology, Peking University

²UC, Merced ³Institute of Big Data, Fudan University

Project page: <https://wang-chaoyang.github.io/project/semflow>
cywang@stu.pku.edu.cn, qqlu1992@gmail.com, xiangtai94@gmail.com

Abstract

Semantic segmentation and semantic image synthesis are two representative tasks in visual perception and generation. While existing methods consider them as two distinct tasks, we propose a unified framework (SemFlow) and model them as a pair of reverse problems. Specifically, motivated by rectified flow theory, we train an ordinary differential equation (ODE) model to transport between the distributions of real images and semantic masks. As the training object is symmetric, samples belonging to the two distributions, images and semantic masks, can be effortlessly transferred reversibly. For semantic segmentation, our approach solves the contradiction between the randomness of diffusion outputs and the uniqueness of segmentation results. For image synthesis, we propose a finite perturbation approach to enhance the diversity of generated results without changing the semantic categories. Experiments show that our SemFlow achieves competitive results on semantic segmentation and semantic image synthesis tasks. We hope this simple framework will motivate people to rethink the unification of low-level and high-level vision.

1 Introduction

Understanding semantic content and creating images from semantic conditions are fundamental research topics in computer vision. Semantic segmentation [45; 7; 59; 85; 11; 10] and image synthesis [8; 32; 28; 72; 90] are two representative dense prediction tasks and inspires various downstream applications, including autonomous driving [5] and medical image analysis [89]. The former aims to assign a category label to each pixel in the image, while the latter aims to generate realistic images given semantic layouts.

Although semantic segmentation and image synthesis constitute a pair of reverse problems, existing works typically solve them using two distinct methodologies. On the one hand, segmentation models mostly follow the spirit of discriminative models. Specifically, a pre-trained backbone is employed to extract multi-scale features, and then task-specific decoders are used for dense prediction. On the other hand, semantic image synthesis frameworks are mainly built upon generative adversarial networks (GAN) [19; 52; 57; 62] or diffusion models (DM) [63; 64; 25]. GAN-based methods [90; 72; 28] typically take semantic layouts as inputs and adopt a discriminator for adversarial training. Meanwhile, DM-based methods [36] generate from noise with semantic layouts functioning as control signals.

An intuitive solution is to represent the segmentation mask as a colormap and model it as a conditional image generation task [56]. However, there are several implementation challenges. Overall, most of

Corresponding author: Lu Qi and Xiangtai Li.

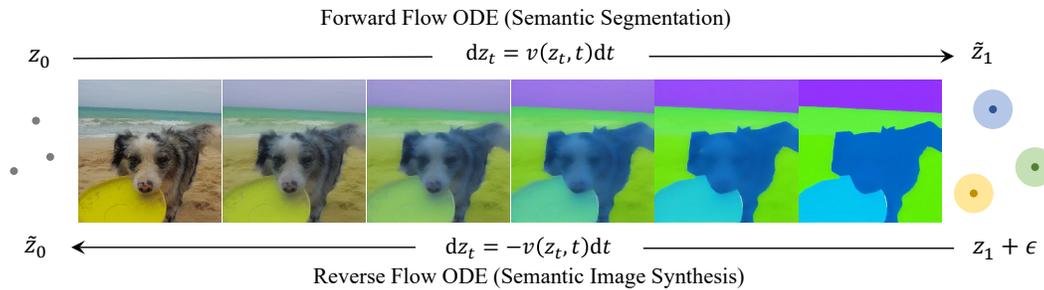


Figure 1: **Rectified flow bridges semantic segmentation (SS) and semantic image synthesis (SIS).** SS and SIS are modeled as a pair of transportation problems between the distributions of images and masks. They share the same ODE and only differ in the direction of the velocity field. We propose a finite perturbation operation on the mask to enable multi-modal generation without changing the semantic labels. *Grey dots* represent data samples. *Colored dots* represent semantic centroids, also known as anchors in Eq. 7. *Colored bubbles* represent the scale of perturbation.

the discriminative segmentation models do not apply to this problem due to their irreversibility. For generative adversarial networks, their generators are typically unidirectional. Jointly training multiple unidirectional models to achieve bidirectional generation [90] is not the concern of this paper. Latent diffusion models (LDMs) have recently demonstrated great potential in generative tasks. Beyond generative tasks, several works [56; 69] attempt to apply diffusion models for segmentation, with images functioning as conditions, but their applicable tasks are limited to be class-agnostic. There are three main problems for existing LDM-based segmentation frameworks: 1) The contradiction between the randomness of generation outputs and the certainty of segmentation labels. 2) The huge cost brought by multiple inference steps. 3) The irreversibility between semantic masks and images.

In this paper, we use rectified flow [40; 42; 17] to enable LDM as a unified framework for semantic segmentation and semantic image synthesis. Our key idea is summarized in Fig. 1. Starting from an LDM [58] framework, we solve the above problems with three methodologies. First, we redefine the mapping function to address the randomness. Previous works [56] aim to learn the mapping from the joint distribution of Gaussian noise and images to the segmentation masks. We argue that Gaussian noise in this mapping function is redundant and negatively affects the determinism of semantic segmentation results. Instead, our model *directly* learns the mapping from images to masks. Secondly, we make the mapping reversible via rectified flow. Rectified flow is an ordinary differential equation (ODE) framework with a time-symmetric training object. This feature allows the model to be trained once to obtain bi-directional transmission capabilities and can be solved numerically using simple ODE solvers such as Euler. Finally, we propose a finite perturbation method to enable multi-modal image synthesis, as the mapping is one-to-one in semantic segmentation but one-to-many in semantic synthesis. We add amplitude-limited noise, enabling masks to be sampled from a collection of semantic-invariant distributions rather than a fixed value, and further improving the quality of synthesized results. Moreover, our model needs fewer inference steps than traditional diffusion models because the transport trajectory is straight, significantly reducing the gap between LDM and traditional discriminative models in segmentation.

The main contributions are as follows: 1) We introduce SemFlow, a new unified framework that binds semantic segmentation and image synthesis with rectified flow, effectively leveraging the length of the generative models. 2) We propose specialized designs of SemFlow, including pseudo mask modeling, bi-directional training of segmentation and generation, and a finite perturbation strategy. 3) We validate SemFlow on several popular benchmarks. For semantic segmentation, SemFlow dramatically narrows the gap between diffusion models and discriminative models in terms of accuracy and inference speed with a more elegant framework. Meanwhile, SemFlow also performs decently in semantic image synthesis tasks.

2 Related Work

Diffusion Models and Rectified Flow. Diffusion models [25; 63; 64] have shown impressive results in the field of generation, such as image generation [58; 55; 83; 27; 13; 54], video generation [80],

image editing [4; 46; 60; 51; 23], image super resolution [61; 26] and point cloud [47; 50; 88; 82; 48]. Most of these methods are based on stochastic differential equations (SDEs) and need multiple steps for generation. Recently, some works [40; 39; 38; 1; 22] propose to model with probability flow ordinary differential equations (ODEs) to reduce the inference steps. Specifically, rectified flow [40; 42] defines the forward process as straight paths and uses reflow to minimize the sampling steps to one. Although rectified flow has demonstrated decent results on image generation and image transfer tasks, they are limited to low-level vision and lack an exploration of the unification of segmentation and generation tasks.

Semantic Segmentation is one of the core tasks in visual perception, aiming to assign each pixel of the given image a semantic category. Previous semantic segmentation approaches are typically built upon discriminative modeling, consisting of a strong backbone [21; 16; 43; 44; 33] for feature extraction and a task-specific decoder head [7; 59; 85; 11; 10; 87; 65; 81; 15; 14] for mask prediction. Recently, some works [34; 86; 18; 77; 75; 3; 70; 9; 20; 2; 29; 76; 71] exploit diffusion models for segmentation. They typically follow the spirit of discriminative models and employ the diffusion model as a feature extractor. Although UniGS [56] and LDMSeg [69] attempt to use a plain Stable Diffusion framework for segmentation, their applicable tasks are limited to be class-agnostic. In practice, reconciling the stochastic outputs of diffusion models with the deterministic results of semantic segmentation is difficult. We rethink this problem and re-model it with rectified flow.

Semantic Image Synthesis is the reverse problem of semantic segmentation, aiming to generate realistic images given semantic layouts [8; 53; 41; 74; 91; 66; 67; 79; 32; 35; 49]. Several studies have delved into semantic synthesis through two methodologies. One methodology [90; 28; 72; 6; 68] is based on GAN and trained with adversarial loss and reconstruction loss. However, some of these methods can only generate unimodal outputs. Although some methods [67; 92] have been developed to address the diversity issues, GAN-based frameworks typically suffer from unstable training and need careful parameter tuning. Another methodology [73] employs diffusion models and regards it as a conditional image generation task, where semantic masks function as control signals. Some works further add a greater variety of control signals to enhance the consistency of synthesized results, like textual prompts [78] and bounding box [36]. Despite these methods achieves good synthetic results, their architecture is usually asymmetric, and the generator is usually unidirectional. This hinders the exploration of the unification of semantic segmentation and image synthesis.

3 Method

In this section, we first review diffusion models and the differential equations of diffusion-based segmentation models (DSMs) and then analyze the disadvantages of existing approaches. Afterward, we propose our SemFlow, which is inspired by rectified flow theory. It solves the randomness problem in the existing DSM and unifies semantic segmentation and image synthesis with *one* model. We will elaborate on each phase of our method.

3.1 Diffusion Model and Segmentation Modeling.

Diffusion models are a class of likelihood-based models that define a Markov chain of forward and backward processes. In the forward process, the noise is gradually added to the real data x_0 to form the noisy data x_t :

$$q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\alpha_t}x_0, (1 - \alpha_t)I), \quad t \in [0..T], \quad (1)$$

where α_t is functions of t . During training, the model ϵ_θ learns to estimate the noise by minimizing the objective with L2 loss:

$$\mathcal{L} = \mathbb{E}_{x_0 \sim q(x_0), \epsilon \sim \mathcal{N}(0, I), t} \left[\|\epsilon_\theta(\sqrt{\alpha_t}x_0 + \sqrt{1 - \alpha_t}\epsilon_t) - \epsilon_t\|_2^2 \right], \quad (2)$$

In the backward process, the model starts from Gaussian noise and gradually denoises to generate realistic data [63] via,

$$x_{t-1} = \sqrt{\alpha_{t-1}} \left(\frac{x_t - \sqrt{1 - \alpha_t}\epsilon_\theta(x_t, t)}{\sqrt{\alpha_t}} \right) + \sqrt{1 - \alpha_{t-1} - \sigma_t^2}\epsilon_\theta(x_t, t) + \sigma_t\epsilon_t. \quad (3)$$

Luckily, Eq. 3 provides a unified solution for DDPM and DDIM, where the former belongs to an SDE modeling and the latter is an ODE modeling. We parameterized it as

$$x_0 = f(x_T, \sigma), \quad (4)$$

where σ controls the way of modeling:

$$\sigma_t = \begin{cases} \sqrt{(1 - \alpha_{t-1})(1 - \alpha_t)}\sqrt{1 - \alpha_t/\alpha_{t-1}} & \text{DDPM} \\ 0 & \text{DDIM} \end{cases} \quad (5)$$

In the image generation task, x_0 represents the real images, and x_T represents the Gaussian noise. We extend the boundaries of Eq. 4 and apply it to semantic segmentation problems. An intuitive way is to encode the segmentation mask into colormaps and model segmentation as a conditional image generation task. Given image I , we rewrite the projection f into a formulation of conditional generation,

$$x_0 = f(x_T, \sigma, I). \quad (6)$$

Eq. 6 formulates the existing diffusion-based segmentation models with generative modeling. However, this modeling approach suffers from the following problems: **1)** The randomness of initial noise and the determinism of the segmentation mask are contradictory. **2)** From a transmission point of view, the transfer from noise to the segmentation mask does not conform to the paradigm of semantic segmentation task, where noise is redundant. **3)** The images in this approach only serve as the condition, which is non-causal in reverse transfer. Thus, the reversed transfer is a separate problem.

3.2 Unify Segmentation and Synthesis with Rectified Flow

Task-agnostic Framework. We employ the *standard Stable Diffusion* [58] (SD) framework for our task without any task-specific decoder head or well-designed text prompts. To this end, we design the network architecture using the following three steps. First, we convert the semantic segmentation masks to 3-channel pseudo mask $M = (m_0, m_1, m_2)$ to align with the images I . Assume that the valid region $[0, 255]$ (for real images) is divided into k parts with a spacing of s , the pseudo mask corresponding to category index c is formulated as:

$$m'_0 = \lfloor c/k^2 \rfloor, \quad m'_1 = \lfloor (c - m'_0 * k^2)/k \rfloor, \quad m'_2 = c - m'_0 * k^2 - m'_1 * k, \quad M = s * (m'_0, m'_1, m'_2), \quad (7)$$

where $s * (k - 1) < 255$ and $\lfloor \cdot \rfloor$ means the floor operator. The (m_0^i, m_1^i, m_2^i) are called anchors.

After the transformation, we adopt a VAE encoder \mathcal{E} to compress the images and pseudo masks into the latent space. The corresponding VAE decoder \mathcal{D} restores the latent variables to the pixel space.

$$z_0 = \mathcal{E}(I), \quad z_1 = \mathcal{E}(M), \quad \tilde{I} = \mathcal{D}(\tilde{z}_0), \quad \tilde{M} = \mathcal{D}(\tilde{z}_1), \quad (8)$$

where \tilde{z} means the output of the UNet.

Discussion. Although previous works [69] argue that segmentation masks are lower in entropy and specifically re-train a lighter network, we still adopt the off-the-shelf VAE from SD. The reasons are as follows: 1) The number of parameters in VAE is negligible compared to the UNet (84M vs. 860M). 2) The VAE specifically trained for segmentation masks can not be applied to images, thus destroying the bi-directional transportation. 3) Specific training creates spatial priors, such as clustering, which hinders our exploration of LDM's segmentation capability itself.

Note that we do not use image captions or image features as prompts. As our model unifies semantic segmentation and image synthesis with *one* model, the usage of captions contradicts the definition of semantic segmentation task while the features are non-causal for image synthesis. Thus, in this work, we set the prompt as empty.

Bi-directional Training and Inference. Contrary to the conventional approach, we propose modeling the segmentation task with rectified flow. It is an ODE framework that aims to learn the mapping between two distributions through straight trajectories.

Denote π_0 and π_1 represent the distribution of the latent variables of images and masks, respectively. Denote $z_0 \sim \pi_0$ and $z_1 \sim \pi_1$, the trajectory from z_0 to z_1 can be formulated as $z_t = \varphi_t(z_0, z_1)$:

$$\frac{dz_t}{dt} = \frac{\partial \varphi_t(z_0, z_1)}{\partial t}. \quad (9)$$

When φ is the trajectory of rectified flow [40], z_t can be reformulated as the linear interpolation process between z_0 and z_1 as $z_t = (1 - t)z_0 + tz_1$. We aim to learn the velocity field using neural

networks $v_\theta(z_t, t)$ and solve it with optimization methods,

$$\begin{aligned} \mathcal{L} &= \int_0^1 \mathbb{E}_{(z_0, z_1) \sim \gamma} \left[\left\| v_\theta(z_t, t) - \frac{\partial \varphi_t(z_0, z_1)}{\partial t} \right\|^2 \right] dt, \\ &= \int_0^1 \mathbb{E}_{(z_0, z_1) \sim \gamma} \left[\|v_\theta(z_t, t) - (z_1 - z_0)\|^2 \right] dt, \end{aligned} \quad (10)$$

where γ indicates the coupling of images and their corresponding pseudo masks.

Eq. 10 is our training loss. Upon training completed, the transfer from z_0 to z_1 can be described via an ODE:

$$\frac{dz_t}{dt} = v_\theta(z_t, t), \quad t \in [0, 1]. \quad (11)$$

So far, we have constructed a mapping from the distribution of images to that of masks. Compared with DSMs in Eq. 6, our approach avoids the interference of randomness, enabling the application of diffusion models to semantic segmentation tasks.

Moreover, Eq. 10 has a time-symmetric form, which results in an equivalent problem by exchanging z_0 and z_1 and flipping the sign of v_θ . Interestingly, the transportation problem from π_1 to π_0 indicates the semantic image synthesis task. This means that semantic segmentation and semantic image synthesis essentially become a pair of mutually reverse problems that share the same ODE (Eq. 11) and have solutions with opposite signs.

In the inference stage, we can obtain the approximate results by numerical solvers like the forward Euler method, which can be formulated as follows,

Semantic segmentation is regarded as the transportation from z_0 to z_1 , which we call *forward flow*. The ODE is Eq. 11 and the numerical solution is,

$$z_{t+\frac{1}{N}} = z_t + \frac{1}{N} v_\theta(z_t, t), \quad t \in \{0, 1, \dots, N-1\}/N. \quad (12)$$

After \tilde{M} is restored with Eq. 8, we calculate the L2 distance between \tilde{M} and anchors and obtain the segmentation mask.

Correspondingly, the semantic image synthesis task is considered to transfer in a reverse direction, which we call the *reverse flow*. Specifically, this transfer can be described as $\frac{dz_t}{dt} = -v_\theta(z_t, t)$, and the solution is,

$$z_{t-\frac{1}{N}} = z_t - \frac{1}{N} v_\theta(z_t, t), \quad t \in \{N, N-1, \dots, 1\}/N. \quad (13)$$

3.3 Finite Perturbation

In Eq. 10, the model is configured to learn the one-to-one mapping between images and masks. However, assigning a fixed mask to each image brings about several problems. First, we find that constant z_1 in Eq. 13 hinders the multi-modal generation for the image synthesis task. Moreover, the pseudo masks are low in entropy. We hypothesize that the low-entropy distribution of masks hinders the training process and may finally spoil the quality of synthesis results.

To this end, we propose to add finite perturbation on the pseudo masks. Specifically, given pseudo masks M which has a spacing of s , we add a noise with a limited amplitude on M ,

$$M' = M + \epsilon, \quad \epsilon \sim \mathcal{U}(-\beta, \beta), \quad (14)$$

where \mathcal{U} is a uniform distribution. We set $0 < \beta < s/2$ to ensure that the semantic label of each pixel does not change. Therefore, we propose to replace z_1 with $z'_1 = \mathcal{E}(M')$ in Eq. 10 and Eq. 13. We show the effectiveness of this design in Sec. 4.3.

4 Experiments

4.1 Experimental Settings

Dataset and Metrics. We study SemFlow using three popular datasets: COCO-Stuff [37], CelebAMask-HQ [30], and Cityscapes [12]. They contain 171, 19, and 19 categories, respectively. For semantic segmentation, we evaluate with mean intersection over union (mIoU). For

Table 1: **Semantic segmentation results on COCO-Stuff dataset.** **SS** and **SIS** represents semantic segmentation and semantic image synthesis, respectively. **Sampler-N** means the usage of a specific sampler with N inference steps.

| Method | Task | | Sampler | COCO-Stuff mIoU (SS) | CelebAMask-HQ | |
|-----------------|------|-----|----------|-------------------------|---------------|-------------|
| | SS | SIS | | | FID (SIS) | LPIPS (SIS) |
| MaskFormer [11] | ✓ | | - | 41.9 | - | - |
| DSM [58] | ✓ | | DDIM-200 | 16.1 | - | - |
| DSM [58] | ✓ | | DDPM-200 | 20.2 | - | - |
| pix2pixHD [72] | | ✓ | - | - | 54.7 | 0.529 |
| SPADE [53] | | ✓ | - | - | 42.2 | 0.487 |
| SC-GAN [74] | | ✓ | - | - | 19.2 | 0.395 |
| BBDM [31] | | ✓ | BBDM-200 | - | 21.4 | 0.370 |
| SemFlow (ours) | ✓ | ✓ | Euler-25 | 38.6 | 32.6 | 0.393 |

semantic image synthesis, we assess with the Fréchet inception distance (FID) [24] and learned perceptual image patch similarity (LPIPS) [84].

Implementation Details. The SemFlow model is built upon Stable Diffusion UNet and initialized with weights from the pre-trained SD 1.5. Images and semantic masks in COCO-Stuff, CelebAMask-HQ, and Cityscapes are resized and cropped into 512×512 , 512×512 , and 512×1024 , respectively. We use the off-the-shelf VAE of the corresponding Stable Diffusion model. The spacing s is set as 50, and the division coefficient k is 6. The amplitude of perturbation is 6. Note that there is no need to train two models for segmentation and synthesis separately since the optimization target in Eq. 10 is time-symmetric.

Baselines. We seek to unify semantic segmentation (SS) and semantic image synthesis (SIS) into a pair of reverse problems and train *one* model for a solution. As few previous works achieve this goal, we thus take different baselines for SS and SIS, respectively. For SS, we design two baseline models, DSMs [58], which follow the principle of diffusion-based conditional generation modeling in Sec. 3.1. The network structure and hyperparameters of DSM follow SemFlow, except that the inputs of UNet are eight channels. Specifically, the noise and images are concatenated in the channel dimension. DSM-DDIM and DSM-DDPM differ in differential equation modeling (ODE vs. SDE). For SIS, we take pix2pixHD [72], SPADE [53], SC-GAN [74], BBDM [31] and CycleGAN [90] as the baselines.

4.2 Main Results

Semantic Segmentation. Fig. 2 shows the qualitative comparison between SemFlow and DSMs. Our SemFlow demonstrates satisfactory performance in a range of scenarios and exhibits high accuracy in classifying the semantic labels of targets. In contrast, DSM fails in semantic segmentation tasks, regardless of SDE or ODE modeling. First, DSM is inferior to SemFlow in discriminating different semantic categories. Moreover, the outputs of DSM-DDIM and DSM-DDPM change dramatically with different random seeds. As discussed in Sec. 3.1, DSM is susceptible to the randomness inherent in diffusion models, making it unable to produce deterministic results.

Tab. 1 shows the quantitative results on the COCO-Stuff dataset. We compare SemFlow with two variants of DSM and MaskFormer, which is a classical discriminative segmentation model. Regarding accuracy, our SemFlow achieves 38.6 mIoU and outperforms DSMs by a remarkable margin. Moreover, SemFlow only uses a simple sampler, forward Euler method, and fewer inference steps than DSMs. Further reducing the inference steps brings about faster generation but witnesses a slight drop in performance, which is analyzed in Sec. 4.3.

Semantic Image Synthesis. We compare our approach with other specialist models on semantic image synthesis tasks in Tab. 1. On CelebAMask-HQ, SemFlow achieves a performance of 32.6 FID and 0.393 LPIPS.

Beyond specialist models, we also qualitatively compare our approach with CycleGAN [90] in Fig. 3 to demonstrate the *overall* performance on SS and SIS tasks. All results of SemFlow are generated with *one* model. In other words, the ODE modeling of SS and SIS share the same velocity

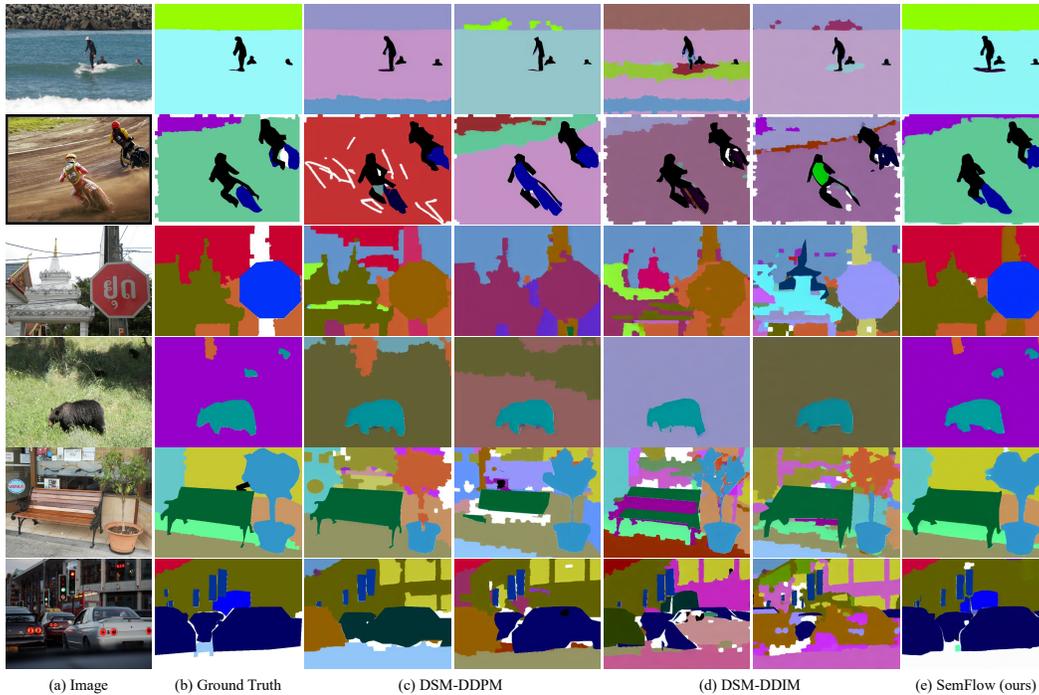


Figure 2: **Semantic segmentation results on COCO-Stuff dataset.** For the ground truth, each color reflects the value of anchors (Eq. 7), which corresponds to one semantic category, and the color white indicates the ignored regions. The predictions of DSM vary considerably under different random seeds.

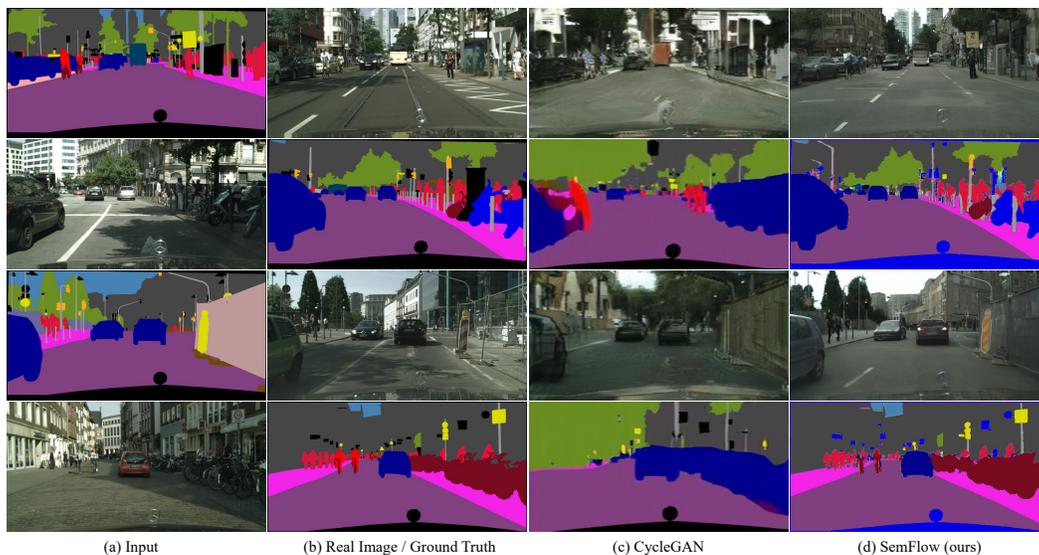


Figure 3: **Semantic segmentation and semantic image synthesis results on Cityscapes dataset.** The color black in the ground truth indicates the ignored region. The segmentation results of SemFlow are colored following [12].

field and only differ in the sign. The first and third rows show the image synthesis results given semantic layouts, while the second and fourth row shows the segmentation results. Our synthesis and segmentation results are inspiring and significantly outperform CycleGAN. Note that CycleGAN essentially trains two unidirectional generators while we employ only *one* model for the two tasks.

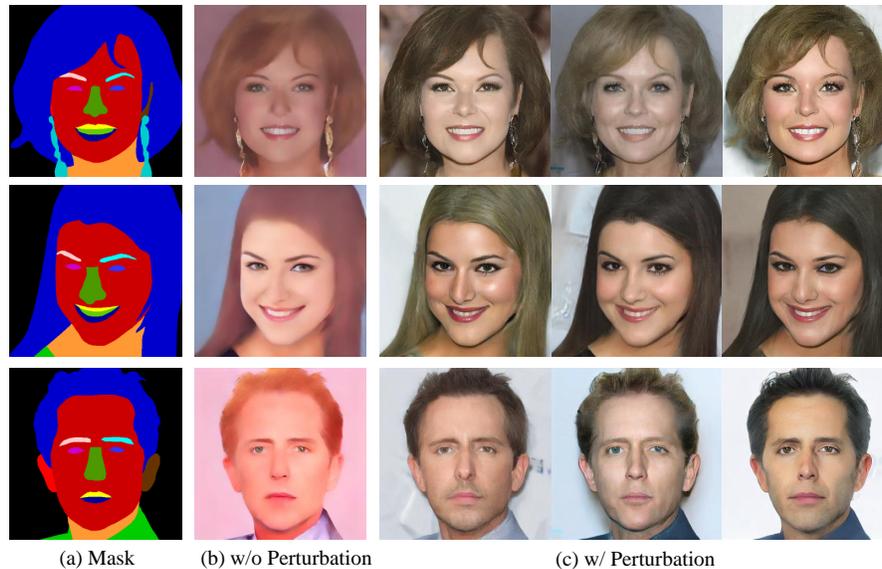


Figure 4: **Semantic image synthesis results on CelebAMask-HQ dataset.** Semantic masks are colored to show different semantic components. SemFlow w/ Perturbation indicates the finite perturbation operation in Eq. 14.

Discussion. Although the performance gap exists compared with specialist models, our approach solves the two major issues of diffusion-based segmentation approaches: 1) Randomness of outputs and 2) Large inference steps. We believe it significantly reduces the gap between discriminative models and diffusion models.

4.3 Ablation Studies

Perturbation. Fig. 4 shows the synthesized results of SemFlow on the CelebAMask-HQ dataset and demonstrates the importance of perturbation operation. First, perturbation enables our approach to generate multi-modal results from one semantic layout using different noises. The synthesized results show diverse appearances, such as skin, hair color, and the background, and exhibit good consistency with the masks. Moreover, we also show the synthesis results of the model without perturbation in Fig. 4. It is witnessed that the model without perturbation is only able to generate uni-modal results, and the synthesized images exhibit a loss of detail and suffer from severe over-smoothing problems.

Straight Trajectory. Our SemFlow aims to establish linear trajectories between the distribution of images and the masks, which enables it to sample with fewer steps.

Tab. 2 compares the performance of DSM and SemFlow on segmentation tasks under different inference steps, where DSM sampled with DDPM and DDIM represent SDE and ODE modeling, respectively. Overall, SemFlow outperforms DSM by a remarkable margin regardless of the inference steps. Specifically, the DSMs' performance declines markedly as the number of steps decreases. With 5 steps, the DSM-DDPM achieves a mIoU of 12.3, while the DSM-DDIM achieves 9.5 mIoU. On the contrary, our SemFlow achieves 28.3 even with one step.

Fig. 5 provides semantic image synthesis results with different inference steps. Although fewer steps will result in some loss of detail and make the images appear smoother, the results of one-step synthesis are still competitive. This indicates that the trajectory between the two distributions is straightened. Fig. 6 and Fig. 7 visualizes the latent variable sampled from the trajectory. The transition between z_0 and z_1 is smooth.

5 Conclusion

In this work, we present SemFlow, a framework that employs a diffusion model to integrate semantic segmentation and semantic image synthesis as a pair of reverse problems. To overcome the con-

Table 2: **Semantic segmentation results with different inference steps on COCO-Stuff dataset.** mIoU is used as the metric.

| Method&Sampler | | 1 | 2 | 5 | 10 | 25 | 50 | 100 | 200 |
|----------------|-------|------|------|------|------|------|------|------|------|
| DSM | DDIM | - | 0.1 | 4.0 | 9.5 | 13.6 | 14.9 | 15.7 | 16.1 |
| | DDPM | - | 0.1 | 5.3 | 12.3 | 17.5 | 18.9 | 19.6 | 20.2 |
| SemFlow | Euler | 28.3 | 31.0 | 36.9 | 38.4 | 38.6 | 38.4 | 38.3 | 38.3 |

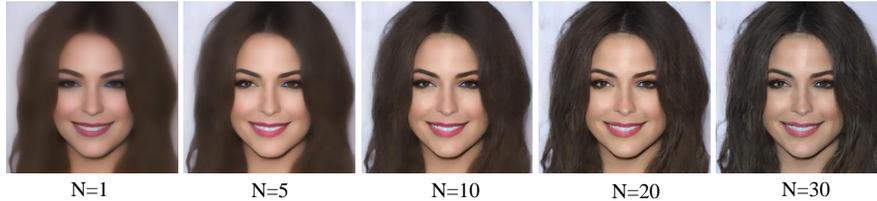


Figure 5: **Image synthesis results with different inference steps.** We use the forward Euler method to get numerical solutions. Our approach obtains competitive results even with only one inference step.



Figure 6: **Visualization of latent variables on the trajectory from z_1 to z_0 (Semantic image synthesis).** Top row: COCO-Stuff. Bottom row: Cityscapes.

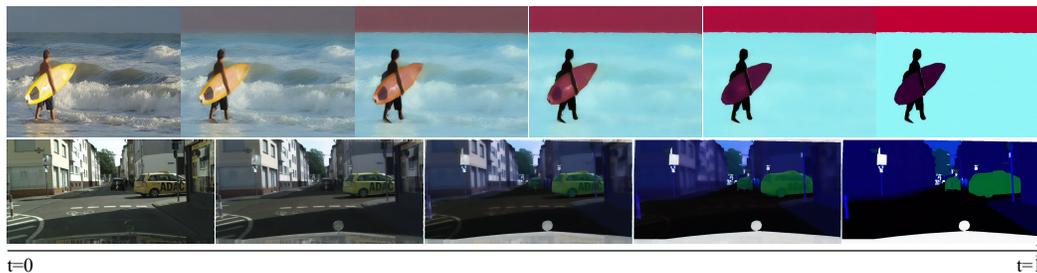


Figure 7: **Visualization of latent variables on the trajectory from z_0 to z_1 (Semantic segmentation).** Top row: COCO-Stuff. Bottom row: Cityscapes.

tradition between the randomness of diffusion models and the certainty of semantic segmentation results, we propose to model semantic segmentation as a transport problem between image and mask distributions. We then employ rectified flow to learn the transfer function, which brings the benefits of reversible transportation. We propose a finite perturbation method to enable multi-modal generation, which also greatly improves the quality of synthesized results. With straight trajectory modeling, our model can sample with much fewer steps. Experimental results show that even with a weak sampler, our model still achieves comparable or even better results than specialist models. We hope our research can inspire the findings on unified generative model design for the community.

Acknowledgement. This work was supported by the National Key Research and Development Program of China (No. 2023YFC3807600).

References

- [1] Michael S Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *arXiv preprint arXiv:2303.08797*, 2023. 3
- [2] Tomer Amit, Tal Shaharabany, Eliya Nachmani, and Lior Wolf. Segdiff: Image segmentation with diffusion probabilistic models. *arXiv preprint arXiv:2112.00390*, 2021. 3
- [3] Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, Valentin Khruikov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. In *ICLR*, 2022. 3
- [4] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023. 3
- [5] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020. 1
- [6] JungWoo Chae, Hyunin Cho, Sooyeon Go, Kyungmook Choi, and Youngjung Uh. Semantic image synthesis with unconditional generator. In *NeurIPS*, 2023. 3
- [7] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *TPAMI*, 2017. 1, 3
- [8] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *ICCV*, 2017. 1, 3
- [9] Ting Chen, Lala Li, Saurabh Saxena, Geoffrey Hinton, and David J Fleet. A generalist framework for panoptic segmentation of images and videos. In *ICCV*, 2023. 3
- [10] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 1, 3
- [11] Bowen Cheng, Alex Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *NeurIPS*, 2021. 1, 3, 6
- [12] 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 *CVPR*, 2016. 5, 7
- [13] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021. 2
- [14] Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, and Chen Change Loy. MeViS: A large-scale benchmark for video segmentation with motion expressions. In *ICCV*, 2023. 3
- [15] Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, Philip HS Torr, and Song Bai. MOSE: A new dataset for video object segmentation in complex scenes. In *ICCV*, 2023. 3
- [16] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 3
- [17] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. *arXiv preprint arXiv:2403.03206*, 2024. 2
- [18] Zigang Geng, Binxin Yang, Tiankai Hang, Chen Li, Shuyang Gu, Ting Zhang, Jianmin Bao, Zheng Zhang, Han Hu, Dong Chen, et al. Instructdiffusion: A generalist modeling interface for vision tasks. In *CVPR*, 2024. 3
- [19] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 2020. 1
- [20] Zhangxuan Gu, Haoxing Chen, Zhuoer Xu, Jun Lan, Changhua Meng, and Weiqiang Wang. Diffusioninst: Diffusion model for instance segmentation. In *ICASSP*, 2024. 3
- [21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3
- [22] Eric Heitz, Laurent Belcour, and Thomas Chambon. Iterative α -(de) blending: A minimalist deterministic diffusion model. In *SIGGRAPH*, 2023. 3
- [23] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. In *ICLR*, 2023. 3
- [24] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 2017. 6
- [25] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 1, 2
- [26] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *JMLR*, 2022. 3
- [27] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS Workshops*, 2021. 2
- [28] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 1, 3
- [29] Minh-Quan Le, Tam V Nguyen, Trung-Nghia Le, Thanh-Toan Do, Minh N Do, and Minh-Triet Tran. Maskdiff: Modeling mask distribution with diffusion probabilistic model for few-shot instance segmentation. In *AAAI*, 2024. 3
- [30] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *CVPR*, 2020. 5

- [31] Bo Li, Kaitao Xue, Bin Liu, and Yu-Kun Lai. Bbdm: Image-to-image translation with brownian bridge diffusion models. In *CVPR*, 2023. 6
- [32] Ke Li, Tianhao Zhang, and Jitendra Malik. Diverse image synthesis from semantic layouts via conditional imle. In *ICCV*, 2019. 1, 3
- [33] Xiangtai Li, Zhao Houlong, Han Lei, Tong Yunhai, and Yang Kuiyuan. Gff: Gated fully fusion for semantic segmentation. In *AAAI*, 2020. 3
- [34] Xinghui Li, Jingyi Lu, Kai Han, and Victor Prisacariu. Sd4match: Learning to prompt stable diffusion model for semantic matching. *arXiv preprint arXiv:2310.17569*, 2023. 3
- [35] Yuheng Li, Yijun Li, Jingwan Lu, Eli Shechtman, Yong Jae Lee, and Krishna Kumar Singh. Collaging class-specific gans for semantic image synthesis. In *ICCV*, 2021. 3
- [36] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *CVPR*, 2023. 1, 3
- [37] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 5
- [38] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *ICLR*, 2023. 3
- [39] Qiang Liu. Rectified flow: A marginal preserving approach to optimal transport. *arXiv preprint arXiv:2209.14577*, 2022. 3
- [40] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023. 2, 3, 4
- [41] Xihui Liu, Guojun Yin, Jing Shao, Xiaogang Wang, et al. Learning to predict layout-to-image conditional convolutions for semantic image synthesis. In *NeurIPS*, 2019. 3
- [42] Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, et al. InstafLOW: One step is enough for high-quality diffusion-based text-to-image generation. In *ICLR*, 2024. 2, 3
- [43] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 3
- [44] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *CVPR*, 2022. 3
- [45] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1
- [46] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *CVPR*, 2022. 3
- [47] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *CVPR*, 2021. 3
- [48] Shitong Luo and Wei Hu. Score-based point cloud denoising. In *ICCV*, 2021. 3
- [49] Zhengyao Lv, Xiaoming Li, Zhenxing Niu, Bing Cao, and Wangmeng Zuo. Semantic-shape adaptive feature modulation for semantic image synthesis. In *CVPR*, 2022. 3
- [50] Zhaoyang Lyu, Zhifeng Kong, Xudong Xu, Liang Pan, and Dahua Lin. A conditional point diffusion-refinement paradigm for 3d point cloud completion. In *ICLR*, 2022. 3
- [51] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *ICLR*, 2022. 3
- [52] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 1
- [53] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*, 2019. 3, 6
- [54] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 2
- [55] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *ICLR*, 2024. 2
- [56] Lu Qi, Lehan Yang, Weidong Guo, Yu Xu, Bo Du, Varun Jampani, and Ming-Hsuan Yang. Unigs: Unified representation for image generation and segmentation. In *CVPR*, 2024. 1, 2, 3
- [57] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*, 2016. 1
- [58] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2, 4, 6
- [59] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 1, 3
- [60] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *SIGGRAPH*, 2022. 3
- [61] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *TPAMI*, 2022. 3
- [62] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *NeurIPS*, 2016. 1
- [63] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 1, 2, 3
- [64] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021. 1, 2
- [65] Yiran Song, Qianyu Zhou, Xiangtai Li, Deng-Ping Fan, Xuequan Lu, and Lizhuang Ma. Ba-sam: Scalable bias-mode attention mask for segment anything model. *arXiv preprint arXiv:2401.02317*, 2024. 3

- [66] Vadim Sushko, Edgar Schönfeld, Dan Zhang, Juergen Gall, Bernt Schiele, and Anna Khoreva. You only need adversarial supervision for semantic image synthesis. In *ICLR*, 2021. 3
- [67] Zhentao Tan, Menglei Chai, Dongdong Chen, Jing Liao, Qi Chu, Bin Liu, Gang Hua, and Nenghai Yu. Diverse semantic image synthesis via probability distribution modeling. In *CVPR*, 2021. 3
- [68] Hao Tang, Song Bai, and Nicu Sebe. Dual attention gans for semantic image synthesis. In *ACM MM*, 2020. 3
- [69] Wouter Van Gansbeke and Bert De Brabandere. A simple latent diffusion approach for panoptic segmentation and mask inpainting. *arXiv preprint arXiv:2401.10227*, 2024. 2, 3, 4
- [70] Qiang Wan, Zilong Huang, Bingyi Kang, Jiashi Feng, and Li Zhang. Harnessing diffusion models for visual perception with meta prompts. *arXiv preprint arXiv:2312.14733*, 2023. 3
- [71] Chaoyang Wang, Xiangtai Li, Henghui Ding, Lu Qi, Jiangning Zhang, Yunhai Tong, Chen Change Loy, and Shuicheng Yan. Explore in-context segmentation via latent diffusion models. *arXiv preprint arXiv:2403.09616*, 2024. 3
- [72] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*, 2018. 1, 3, 6
- [73] Weilun Wang, Jianmin Bao, Wengang Zhou, Dongdong Chen, Dong Chen, Lu Yuan, and Houqiang Li. Semantic image synthesis via diffusion models. *arXiv preprint arXiv:2207.00050*, 2022. 3
- [74] Yi Wang, Lu Qi, Ying-Cong Chen, Xiangyu Zhang, and Jiaya Jia. Image synthesis via semantic composition. In *ICCV*, 2021. 3, 6
- [75] Jiahao Xie, Wei Li, Xiangtai Li, Ziwei Liu, Yew Soon Ong, and Chen Change Loy. Mosaicfusion: Diffusion models as data augmenters for large vocabulary instance segmentation. *arXiv preprint arXiv:2309.13042*, 2023. 3
- [76] Guangkai Xu, Yongtao Ge, Mingyu Liu, Chengxiang Fan, Kangyang Xie, Zhiyue Zhao, Hao Chen, and Chunhua Shen. Diffusion models trained with large data are transferable visual models. *arXiv preprint arXiv:2403.06090*, 2024. 3
- [77] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *CVPR*, 2023. 3
- [78] Han Xue, Zhiwu Huang, Qianru Sun, Li Song, and Wenjun Zhang. Freestyle layout-to-image synthesis. In *CVPR*, 2023. 3
- [79] Dingdong Yang, Seunghoon Hong, Yunseok Jang, Tianchen Zhao, and Honglak Lee. Diversity-sensitive conditional generative adversarial networks. In *ICLR*, 2019. 3
- [80] Ruihan Yang, Prakhar Srivastava, and Stephan Mandt. Diffusion probabilistic modeling for video generation. *Entropy*, 2023. 2
- [81] Haobo Yuan, Xiangtai Li, Chong Zhou, Yining Li, Kai Chen, and Chen Change Loy. Open-vocabulary sam: Segment and recognize twenty-thousand classes interactively. *arXiv preprint*, 2024. 3
- [82] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. Lion: Latent point diffusion models for 3d shape generation. In *NeurIPS*, 2022. 3
- [83] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023. 2
- [84] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 6
- [85] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017. 1, 3
- [86] Wenliang Zhao, Yongming Rao, Zuyan Liu, Benlin Liu, Jie Zhou, and Jiwen Lu. Unleashing text-to-image diffusion models for visual perception. In *ICCV*, 2023. 3
- [87] Chong Zhou, Xiangtai Li, Chen Change Loy, and Bo Dai. Edgesam: Prompt-in-the-loop distillation for on-device deployment of sam. *arXiv preprint arXiv:2312.06660*, 2023. 3
- [88] Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion. In *ICCV*, 2021. 3
- [89] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *MICCAI*, 2018. 1
- [90] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 1, 2, 3, 6
- [91] Peihao Zhu, Rameen Abdal, Yipeng Qin, and Peter Wonka. Sean: Image synthesis with semantic region-adaptive normalization. In *CVPR*, 2020. 3
- [92] Zhen Zhu, Zhiliang Xu, Ansheng You, and Xiang Bai. Semantically multi-modal image synthesis. In *CVPR*, 2020. 3

Overview. In this supplementary, we present more results and details:

- **A.** More details on our method.
- **B.** More empirical studies and visual results.
- **C.** Discussion of the limitations, broader impacts and safeguards.

A More Details on Method

Amplitude of Perturbation. In the main paper, the perturbation follows the uniform distribution $\epsilon \sim U(-\beta, \beta)$. We set $0 < \beta < s/2$ to prevent the semantic labels from changing. Here is the proof.

Let $M = (m_0, m_1, m_2)$ be the anchor for category c_1 , $M' = M + \epsilon$ be the anchor with perturbation, and $P = (p_0, p_1, p_2)$ be another anchor for category $c_2, c_1 \neq c_2$. First, it is obvious that $E(M') = E(M) = M$, which means that adding perturbation will not change the estimation of original anchors. Then we only need to prove $\|M' - M\|_2 < \|M' - P\|_2$, which is equivalent to $\sum_{i \in [3]} \epsilon_i^2 < \sum_{i \in [3]} (\epsilon_i + m_i - p_i)^2$, and can be further written as $0 < \sum_{i \in [3]} (m_i - p_i)(m_i - p_i + 2\epsilon_i)$. Since $-s < 2\epsilon_i < s$ and $m_i - p_i$ can only take the values of $\{\dots, -2s, -s, 0, s, 2s, \dots\}$, the two terms must have the same sign or one of them equals 0. The proof is completed.

Implementation Details. We train CelebAMask-HQ and Cityscapes with a batch size of 256 with AdamW optimizer for 80K and 8K steps, respectively. The initial learning rate is set as 2×10^{-5} and 5×10^{-5} . Linear learning rate scheduler is adopted. For COCO-Stuff dataset, we use a constant learning rate of 1×10^{-5} with a batch size of 128 for 320K steps. The COCO-Stuff indicates the version of COCO-Stuff-164k.

B More Empirical Studies and Visual Results

Sampling from Other Distributions. Fig. 8 shows the synthesized images which are sampled with different perturbations $\epsilon \sim U(-\beta', \beta')$. It can be observed that the model can obtain high-quality synthetic results only when the perturbations in the inference stage and the training stage follow the same distribution. Specifically, the synthesized images are dim when $\beta' < \beta$, and overexposed when $\beta' > \beta$.

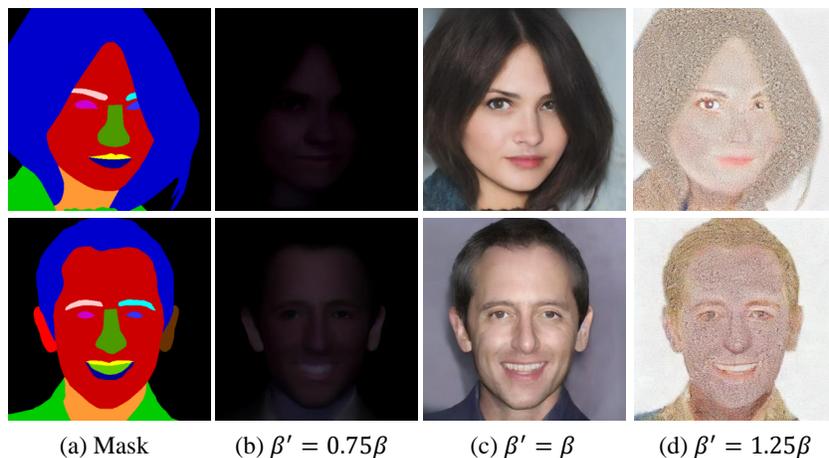


Figure 8: **Image synthesis results sampled from other distributions.** We sample with different perturbation $\epsilon \sim U(-\beta', \beta')$. In the training stage, the perturbation follows $U(-\beta, \beta)$.

More Visual Results on One-step Generation. Fig. 9 shows that our model can generate competitive images with only one inference step.

Selection of ODE solvers. Fig. 10 provides visualization results on different ODE solvers. A stronger solver has the potential to bring about better results.



Figure 9: **Synthesized images with one inference step.**



Figure 10: **The influences of ODE solvers.** (a) Euler indicates sampling with euler-25 solver. (b) RK45 indicates the Runge-Kutta method of order 5(4).

More Visual Results on Cityscapes. Fig. 11 shows the results of multi-modal image synthesis. The appearance of the synthesized scene varies according to different random seeds. Moreover, our model can learn the laws of shadows and light angles (the third column), demonstrating its great potential.

Fig. 12 shows visual results of semantic segmentation. Our model accurately segments the objects and assigns the correct semantic labels.

C Discussion

Limitations. Some previous work, such as Stable Diffusion, SDXL show that scaling up the training data is essential to exploit the potential of diffusion models fully. Due to the limited computational resources and the number of image-mask pairs in the existing dataset, we leave it to future work. In addition, although this paper does not focus on the network structure and the design of ODE solvers, we believe there is room for exploration to further improve performance.

Broader Impacts. Our SemFlow extends the diffusion model from the generative domain to the segmentation task and narrows the gap with traditional discriminative models. It also functions as a unified framework to bind semantic segmentation and image synthesis. We hope this will motivate people to explore the unification of low-level and high-level vision. In terms of social impact, it will encourage artistic content creation, but at the same time may face the problem of fake content.

Safeguards. The content produced by a generative model is highly correlated with its training data. Ensuring fair and clean training data can effectively prevent models from generating harmful content.

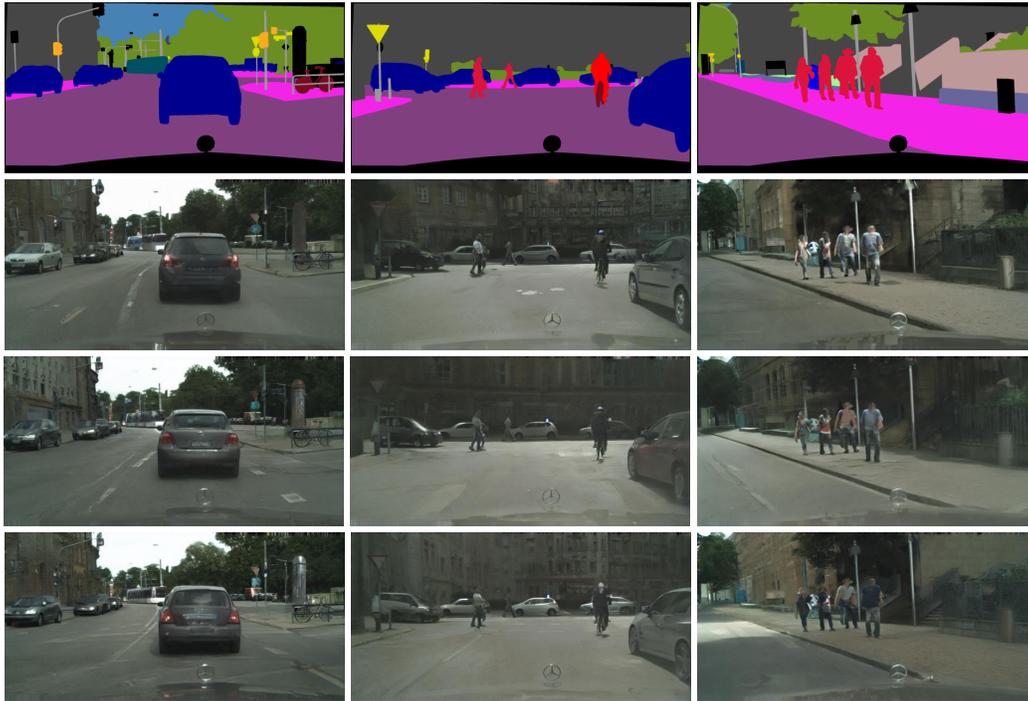


Figure 11: **Image synthesis results on Cityscapes.** We show the results under three random seeds for each semantic mask. The first row: semantic layouts. The second to the fourth row: synthesized results.

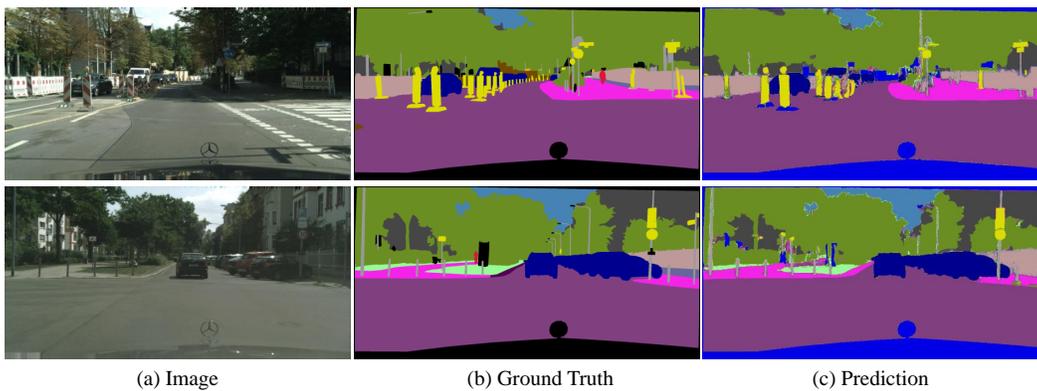


Figure 12: **Semantic segmentation results on Cityscapes.** The color black in the ground truth indicates the ignored regions.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification: We include the main claims in the abstract and introduction.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: We discuss the limitations in Appendix C.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory Assumptions and Proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Assumptions and proof are in Section 3 and Appendix A.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental Result Reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide the implementation details in Section 4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: We are not able to provide the code at submission time. We will definitely release the code in the future.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental Setting/Details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Experimental setting and details are in Section 4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment Statistical Significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: We did not conduct experiments with error bars.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments Compute Resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We provide the implementation details in Section 4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code Of Ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines?>

Answer: [Yes]

Justification: We conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics .

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader Impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We discuss broader impacts in Appendix C.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [Yes]

Justification: We describe the safeguards in Appendix C.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We use existing assets with properly credited.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: No new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: Not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: Not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.