MaskFactory: Towards High-quality Synthetic Data Generation for Dichotomous Image Segmentation

Haotian Qian[1][*] **YD Chen**[1][*] **Shengtao Lou**[1] **Fahad Shahbaz Khan**[3,4]
Xiaogang Jin[1][†] **Deng-Ping Fan**[2,3]
[1]State Key Lab of CAD&CG, Zhejiang University   [2]VCIP&CS, Nankai University
[3]MBZUAI   [4]Linköping University

Abstract

Dichotomous Image Segmentation (DIS) tasks require highly precise annotations, and traditional dataset creation methods are labor intensive, costly, and require extensive domain expertise. Although using synthetic data for DIS is a promising solution to these challenges, current generative models and techniques struggle with the issues of scene deviations, noise-induced errors, and limited training sample variability. To address these issues, we introduce a novel approach, **MaskFactory**, which provides a scalable solution for generating diverse and precise datasets, markedly reducing preparation time and costs. We first introduce a general mask editing method that combines rigid and non-rigid editing techniques to generate high-quality synthetic masks. Specially, rigid editing leverages geometric priors from diffusion models to achieve precise viewpoint transformations under zero-shot conditions, while non-rigid editing employs adversarial training and self-attention mechanisms for complex, topologically consistent modifications. Then, we generate pairs of high-resolution image and accurate segmentation mask using a multi-conditional control generation method. Finally, our experiments on the widely-used DIS5K dataset benchmark demonstrate superior performance in quality and efficiency compared to existing methods. The code is available at `https://qian-hao-tian.github.io/MaskFactory/`.

1 Introduction

Dichotomous image segmentation (DIS) aims to accurately segment objects from natural images [1, 2], a critical task in various computer vision applications, including medical imaging [3, 4, 5], autonomous driving [6] and connectomics research [7, 8]. However, traditional methods for collecting datasets for DIS tasks are labor-intensive, costly, and require extensive domain expertise. Recently, synthetic data has emerged as a promising solution for generating diverse and precise datasets at scale, offering a scalable and cost-effective means for model training. Given the importance of high-quality synthetic data for training models, developing methods to generate such datasets for DIS tasks is crucial. Although generative models have been employed to assist in synthetic dataset production, they face significant limitations in terms of controllability, precision, and diversity.

Recent synthetic methods [9, 10, 11] utilize diffusion models [12] to synthesize image-mask pairs from textual cues and attention maps. Despite progress, they still encounter three major challenges: (1) *Controllability:* Text-guided generation scheme may deviate from real-world scenes, especially with fine-grained labels, leading to uncontrollable generated images. (2) *Precision:* The use of attention maps can introduce noise, which degrades mask fidelity and poses challenges for applications

---

[*]Equal contribution. Deng-Ping Fan served as the project leader for this work.
[†]Corresponding author. Contact the author at jin@cad.zju.edu.cn

Original Source		Rigid Mask Editing		Non-rigid Mask Editing	
Real Image	Source Mask	Synthetic Image	Synthetic mask	Synthetic Image	Synthetic mask

Figure 1: Shows the edited masks from the first stage and the corresponding images generated in the second stage. In the examples, we transformed the viewpoint of park benches and tables from a frontal view to a top-down view and edited their shapes, changing park benches from curved to square edges and tables from square to circular shapes.

requiring precise pixel-level segmentation, such as DIS [2]. 3. *Diversity:* Previous works [13, 14, 15] are relatively uniform, with limited ability to generate diverse training samples, which makes them unsuitable for providing sufficient variability in training data.

Generating diverse human images often poses challenge due to the complexity of prompts, which limits their applicability in tasks requiring extensive variability. It's worthy noting that the expansion of datasets with fine granularity, such as DIS datasets, currently lacks an effective solution. In fact, we observe that existing DIS datasets predominantly consist of single-source orthophotos featuring limited shape diversity among similar objects. The generation of more diverse DIS images that accurately reflect real-world distributions remains an unresolved challenge.

To address these challenges, we propose MaskFactory, a novel two-stage method that efficiently generates high-quality synthetic datasets for DIS tasks. Based on the geometric characteristics of objects, our approach simultaneously considers both rigid and non-rigid transformations of target objects. This means that when generating synthetic datasets, we take into account not only changes in viewpoints (rigid transformations) but also deformations (non-rigid transformations), as illustrated in Figure 1. In the first stage, rigid transformations are driven by geometric priors learned from large-scale diffusion models, enabling precise viewpoint changes and simulating diverse variations in observation angles and scales. Non-rigid transformations leverage prompts provided by large language models to accurately alter the shape of target objects via attention-based editing, ensuring topological consistency before and after editing through topology-preserving adversarial training. Consequently, even for masks with complex geometries, high-quality and diverse synthetic masks can be generated for DIS tasks. In the second stage, we utilize multiple control inputs, including masks, canny edges, and prompts representing class information, to guide the generation process. These inputs are fed into diffusion models to produce high-resolution images that match the prepared segmentation masks. This approach ensures high consistency between images and masks while enhancing the realism and diversity of the dataset.

Our method significantly enhances the realism and diversity of generated datasets. We validate the effectiveness of MaskFactory on the DIS5K dataset [2], specifically designed to evaluate DIS performance. Experimental results show that our approach outperforms existing dataset synthesis methods in terms of structural preservation and error metrics, achieving an average performance gain of 8.8%. These findings highlight the potential of MaskFactory to provide the data diversity and annotation precision required for DIS tasks while significantly reducing the time and costs associated with dataset preparation.

In summary, the contributions of our work are listed as follows:

- We introduce MaskFactory, a novel approach that generate high-quality datasets for DIS task in terms of quality, precision, and efficiency.
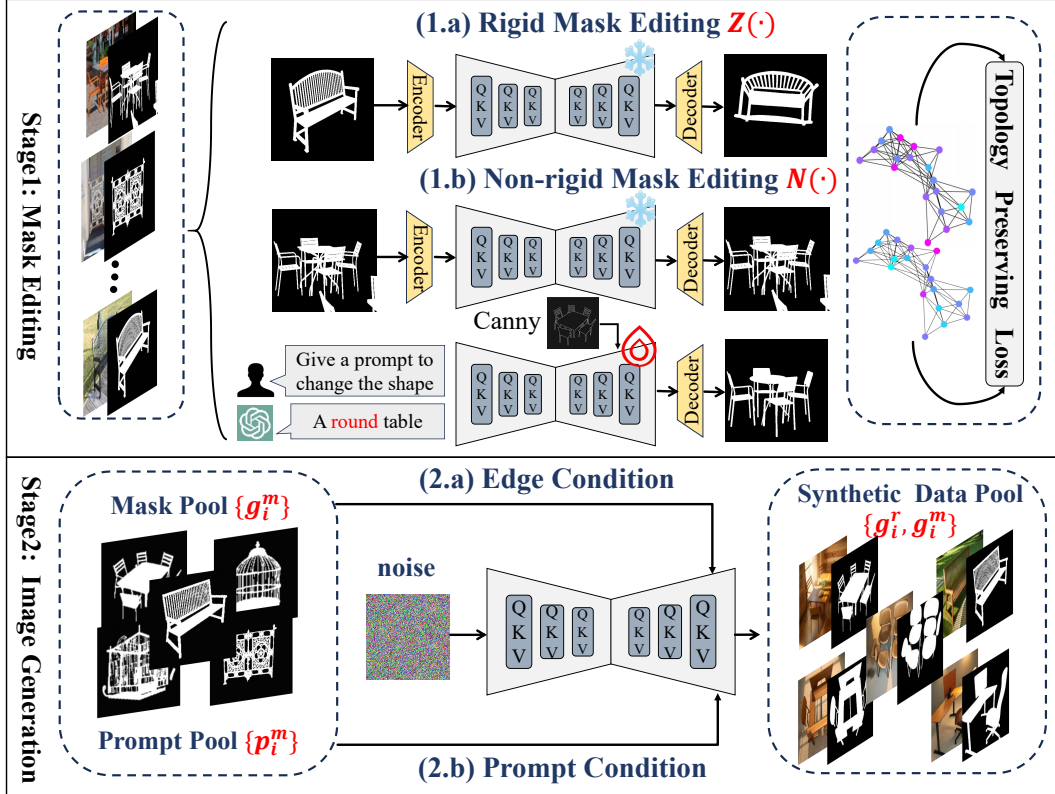
Figure 2: Workflow of MaskFactory. In the first stage, we generate new masks by applying rigid and non-rigid editing to the existing ground truth masks. In the second stage, we use the generated masks and their corresponding extracted Canny edges as conditions, along with a prompt representing the category, to generate RGB images. This process forms paired data for our generative model.

- We propose a two-step method that synthesizes high-quality and diverse object masks via masking editing and generates corresponding high-resolution images using a multi-conditional control generation method.
- Experimental results on the DIS5K dataset demonstrate the superior performance of MaskFactory, compared to existing dataset synthesizing methods.

2 Related work

Synthetic data. Synthetic data has garnered significant attention in various machine learning and computer vision tasks, such as natural language processing [16], object detection [17], and image segmentation [10]. For instance, Lin *et al.* [17] demonstrated that synthetic data can enhance performance in object detection, especially in scenarios with limited access to real-world data. Generative adversarial networks (GANs) and variational autoencoders (VAEs) are two popular deep learning-based methods for generating synthetic data [18]. Recent studies have focused on improving the quality and diversity of synthetic data [19] and exploring its use in few-shot learning [20]. Additionally, synthetic data has been shown to enhance the interpretability and explainability of machine learning models [21]. Benefiting from recent advancements in diffusion models [12, 22], methods such as DatasetDM [9] and Dataset Diffusion [10] can generate high-quality synthetic data for various computer vision tasks. However, these methods may introduce additional errors due to the inclusion of pseudo-labels. These errors arise because pseudo-labels can be noisy and inaccurate, leading to suboptimal training data. Although some research [23] employs controlnet-like control schemes [24] to mitigate these issues, the generated synthetic data may still suffer from noise and errors, making it unsuitable for the DIS task. Therefore, in this paper, we focus on generating high-quality synthetic data for the DIS task, where the generated image-mask pairs need to be highly precise and accurate.

Dichotomous image segmentation. DIS has seen substantial progress with the advent of high-resolution imaging technologies. Representative approaches include the intermediate supervision strategy in IS-Net [2], the frequency prior method [1], and the unite-divide-unite strategy by [25]. However, these methods have enhanced segmentation accuracy but often struggle to capture extremely fine details. Thus, recent progressive refinement strategies, *e.g.,* BASNet [26], emphasize the importance of auxiliary information such as gradient maps and multi-scale inputs. These methods propose to use gradient features and ground truth supervision to enhance the learning of weak features in complex regions. Then, BiRefNet [1] maintains high-resolution inputs and employs a bilateral reference framework to better capture intricate details. The latest works, such as the multi-view aggregation network by Yu et al. [27] and the interactive segmentation approach by Liu et al. [28], further advance the field by integrating diverse prompts and enhancing feature representation for high-quality segmentation.

3 Method

3.1 Overview of MaskFactory

Current image segmentation methods are significantly constrained by their dependence on limited manually annotated data, which hampers both performance and generalization. To mitigate this, pseudo-label generation is often employed to augment training datasets. However, in the context of DIS, these methods frequently introduce artifacts that degrade segmentation quality. Additionally, existing image editing techniques often fail to preserve the topological structure of binary masks, resulting in discontinuities and overlaps within target regions.

To address these challenges, we introduce the MaskFactory framework, designed to generate a large number of high-quality synthetic image-mask pairs $\mathcal{G} = \{(g_i^r, g_i^m)\}_{i=1}^M$ from an original dataset $\mathcal{D} = \{(I_i^r, I_i^m)\}_{i=1}^N$. This approach aims to enhance the performance of DIS models. As illustrated in Figure 2, our framework comprises two main stages: mask editing and image generation.

In the mask editing stage, source masks from the original dataset are transformed using both rigid and non-rigid editing methods, resulting in a set of high-precision synthetic masks $\mathcal{G}^m = \{g_i^m\}_{i=1}^M$. Rigid mask editing generates synthetic masks from various perspectives, while non-rigid mask editing employs a topology-preserving adversarial training mechanism to edit masks according to semantic prompts while retaining the structural integrity of the source masks.

In the image generation stage, a multi-conditional control generation method is utilized to produce realistic RGB images $\mathcal{G}^r = \{g_i^r\}_{i=1}^M$ that correspond precisely to the synthetic masks, using the latter as conditioning constraints.

3.2 Mask Editing Stage

3.2.1 Rigid Mask Editing

Rigid mask editing aims to preserve detailed information from the source masks through rigid transformations. We leverage the Zero123 [29] method, which employs a viewpoint-conditioned diffusion model ψ_θ to manipulate masks' perspectives. Given the relative camera rotation and translation $\mathbf{T}_i$ for the desired viewpoint, ψ_θ synthesizes a new mask g_i^m based on the source mask I_i^m, such that $g_i^m = \psi_\theta(I_i^{m'}, \mathbf{T}_i)$, where $I_i^{m'}$ is the inverted image of the source mask I_i^m to ensure the main component is non-zero.

3.2.2 Non-Rigid Mask Editing

Non-rigid mask editing, inspired by MasaCtrl [30], is a critical component of MaskFactory. Unlike MasaCtrl, which directly manipulates the source mask I_i^m using a textual prompt P_i to generate a synthetic mask g_i^m, we introduce a topology-preserving adversarial training mechanism to mitigate artifacts and structural degradation in binary mask editing. The textual prompt P_i is derived from a pool of prompts $\{p_i^m\}$ that are generated using GPT-4 based on the original images. These prompts provide detailed descriptions that guide the mask editing process. This module consists of a generator G_θ and a discriminator D_ϕ. The generator transforms noise $\mathbf{z}$ into a synthetic mask g_i^m under the guidance of a textual prompt P_i and the source mask I_i^m. A mutual attention mechanism aligns the

query features $\mathbf{Q}_t$ of g_i^m with the key and value features $\mathbf{K}_s, \mathbf{V}_s$ of I_i^m, ensuring consistency during editing. To avoid foreground-background confusion, a mask $\mathbf{M}$ is extracted from the cross-attention maps to guide the model's focus.

Topology-Preserving Adversarial Training. To maintain the structural information of the source mask, we first extract an edge map $E_s = \mathcal{E}(I_i^m)$ using an edge detection operator $\mathcal{E}$, obtaining key points $\mathcal{V} = \{v_j\}_{j=1}^{N_v}$. We then construct a graph $\mathcal{T} = (\mathcal{V}, \mathcal{E}_s)$ based on these key points. The discriminator D_ϕ performs adversarial training on the structural graphs $\mathcal{T}_g$ and $\mathcal{T}_s$ of the synthetic mask g_i^m and the source mask, respectively, ensuring topological consistency.

The training objective of the discriminator D_ϕ is to maximize:

$$\max_\phi \mathbb{E}_{\mathcal{T}_s \sim p_{\text{data}}(\mathcal{T}_s)}[\log D_\phi(\mathcal{T}_s)] + \mathbb{E}_{\mathcal{T}_g \sim p_{\text{gen}}(\mathcal{T}_g)}[\log(1 - D_\phi(\mathcal{T}_g))], \tag{1}$$

where $p_{\text{data}}(\mathcal{T}_s)$ and $p_{\text{gen}}(\mathcal{T}_g)$ represent the distributions of the structural graphs of the source masks and the synthetic masks, respectively. Conversely, the training objective of the generator G_θ is to minimize the discriminative power of the discriminator:

$$\min_\theta \mathbb{E}_{\mathcal{T}_g \sim p_{\text{gen}}(\mathcal{T}_g)}[\log(1 - D_\phi(\mathcal{T}_g))]. \tag{2}$$

Through topology-preserving adversarial training, the non-rigid editing module effectively retains the structural information from the source masks during the editing process, generating high-quality, artifact-free synthetic masks.

The overall loss function $\mathcal{L}_{\text{total}}$ for non-rigid mask editing encompasses the adversarial loss $\mathcal{L}_{\text{GAN}}$, the content loss $\mathcal{L}_{\text{content}}$, and the structure preservation loss $\mathcal{L}_{\text{structure}}$:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{GAN}} + \lambda_1 \mathcal{L}_{\text{content}} + \lambda_2 \mathcal{L}_{\text{structure}}, \tag{3}$$

where λ_1 and λ_2 are balancing factors.

The adversarial loss $\mathcal{L}_{\text{GAN}}$ is defined as:

$$\mathcal{L}_{\text{GAN}} = \mathbb{E}_{\mathcal{T}_s \sim p_{\text{data}}(\mathcal{T}_s)}[\log D_\phi(\mathcal{T}_s)] + \mathbb{E}_{\mathcal{T}_g \sim p_{\text{gen}}(\mathcal{T}_g)}[\log(1 - D_\phi(\mathcal{T}_g))]. \tag{4}$$

The content loss $\mathcal{L}_{\text{content}}$ measures the semantic consistency between the synthetic mask and the textual prompt:

$$\mathcal{L}_{\text{content}} = \|g_i^m - I_i^m\|_1, \tag{5}$$

where g_i^m is the synthetic mask and I_i^m is the source mask.

The structure preservation loss $\mathcal{L}_{\text{structure}}$ evaluates the difference between the structural graphs of the synthetic mask and the source mask:

$$\mathcal{L}_{\text{structure}} = \|\mathcal{T}_g - \mathcal{T}_s\|_1. \tag{6}$$

These components ensure that the editing process maintains the structural and semantic consistency of the source masks.

3.3 Image Generation Stage

Following the mask editing stage, we obtain a synthetic mask pool $\{g_i^m\}_{i=0}^M$ comprising finely detailed synthetic masks generated through the aforementioned transformations. In the subsequent image generation stage, we accurately generate corresponding RGB images $\{g_i^r\}_{i=0}^M$ for the masks in the synthetic mask pool using a multi-condition control generation method. The primary segmentation of the RGB images aligns with the corresponding synthetic masks. Inspired by ControlNet [24], we introduce a multi-condition control generation method to achieve precise RGB image generation. This method simultaneously injects the segmentation condition c_i^s and the canny condition c_i^y to steer the denoising process of the random Gaussian noise, ensuring the generated RGB images $\{g_i^r\}_{i=0}^M$ correspond accurately to the synthetic masks $\{g_i^m\}_{i=0}^M$.

Since the synthetic mask itself serves as the segmentation condition c_i^s, when a synthetic mask g_i^m is provided, we only need to extract the canny condition c_i^y using the canny operator:
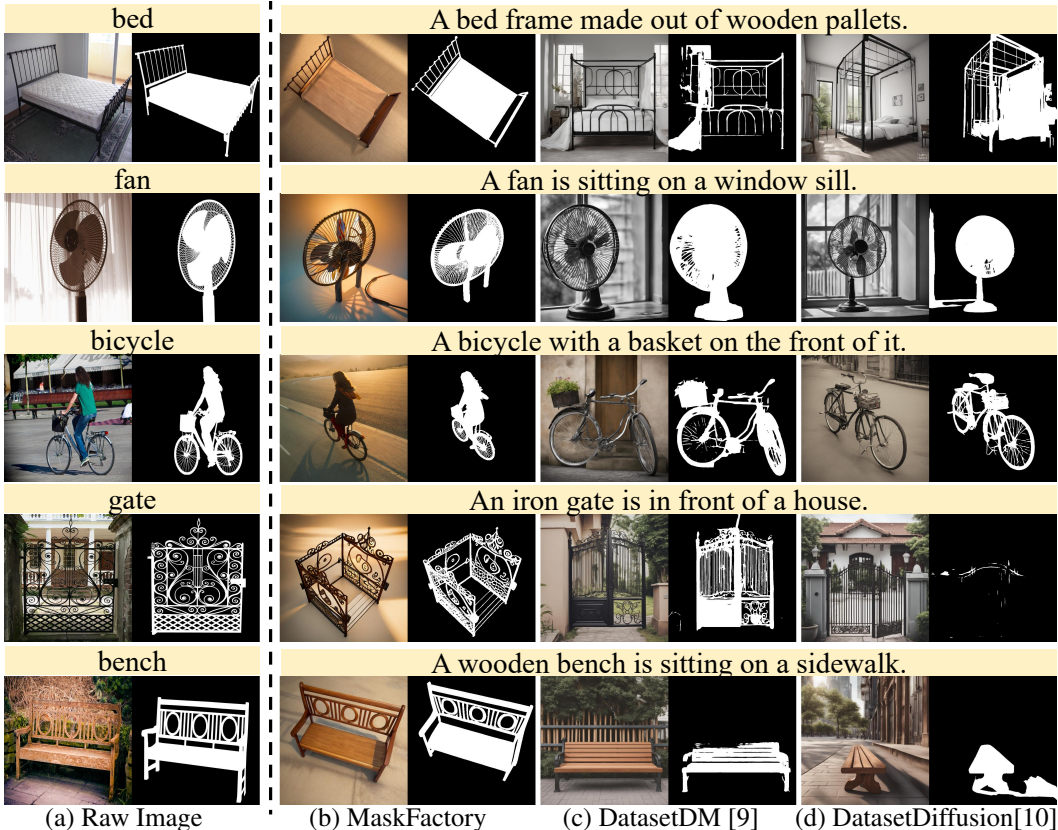
$$c_i^s = g_i^m, \quad c_i^y = Canny(g_i^m) \tag{7}$$

Figure 3: compared with baseline methods

After obtaining the canny and segmentation conditions, we input them into block B_θ, which consists of a set of neural layers. Finally, these conditions are injected into the pre-trained diffusion model M_θ, controlling the noise z denoising process to generate the corresponding RGB image. We refer readers to [24] for more details.

$$g_i^r = M_\theta(z, B_\theta(c_i^s), B_\theta(c_i^y)) \tag{8}$$

4 Experiment and Results

4.1 Dataset & Metrics

We conduct our experiments on the DIS5K dataset, which comprises 5,479 high-resolution images featuring camouflaged, salient, or meticulous objects in various backgrounds. The DIS5K dataset is divided into three subsets: DIS-TR (3,000 images) for training, DIS-VD (470 images) for validation, and DIS-TE (2,000 images) for testing. For data augmentation, we utilize the mask portion of the training subset (DIS-TR).

To evaluate our models, we employ a diverse set of metrics to ensure comprehensive performance assessment. These metrics include max F_1[31], which balances precision and recall, providing a harmonic mean that is indicative of overall accuracy; F_β^ω[32], a weighted F-measure that compensates for class imbalances, with values ranging from 0 to 1, where higher values denote superior performance; M (Mean Absolute Error)[33], which calculates the average absolute difference between the predicted and ground truth masks, with lower values signifying better accuracy; S_α[34], a structural similarity measure that evaluates the preservation of significant structures within the image, with values closer to 1 indicating better performance; and E_M^ϕ[35], an enhanced measure that considers both pixel-level and image-level information for a more holistic evaluation, where higher values represent better performance. Collectively, these metrics provide a robust framework for assessing the effectiveness and reliability of our segmentation models.

Table 1: Comparison of experimental results with different generation methods, illustrating the impact of using various numbers of generated images on segmentation task performance.

Dataset	IS-Net	DatasetDM [9]				Dataset Diffusion [10]				MaskFactory			
		2500	5000	7500	10000	2500	5000	7500	10000	2500	5000	7500	10000
DIS-VD $maxF_1\uparrow$	0.791	0.792	0.791	0.788	0.785	0.793	0.780	0.771	0.767	0.831	0.833	0.834	**0.835**
$F_\beta^\omega\uparrow$	0.717	0.720	0.719	0.712	0.710	0.726	0.726	0.716	0.710	0.725	0.754	0.757	**0.759**
$M\downarrow$	0.074	0.076	0.076	0.075	0.077	0.074	0.077	0.076	0.077	0.073	0.073	**0.071**	0.072
$S_\alpha\uparrow$	0.813	0.814	0.810	0.807	0.805	0.826	0.821	0.804	0.790	0.832	0.855	0.860	**0.866**
$E_M^\phi\uparrow$	0.856	0.869	0.864	0.864	0.860	0.868	0.859	0.852	0.838	0.880	0.911	0.914	**0.923**
DIS-TE1 $maxF_1\uparrow$	0.740	0.744	0.744	0.740	0.736	0.741	0.727	0.721	0.719	0.776	0.777	0.779	**0.784**
$F_\beta^\omega\uparrow$	0.662	0.670	0.668	0.663	0.659	0.675	0.669	0.667	0.654	0.677	0.700	0.703	**0.705**
$M\downarrow$	0.074	0.075	0.077	0.075	0.076	0.076	0.074	0.077	0.078	0.073	**0.071**	**0.071**	0.073
$S_\alpha\uparrow$	0.787	0.791	0.786	0.781	0.778	0.790	0.776	0.760	0.754	0.803	0.822	0.826	**0.829**
$E_M^\phi\uparrow$	0.820	0.825	0.819	0.813	0.810	0.827	0.816	0.802	0.798	0.853	0.866	0.868	**0.875**
DIS-TE2 $maxF_1\uparrow$	0.799	0.810	0.807	0.801	0.793	0.801	0.799	0.782	0.776	0.808	0.814	0.819	**0.822**
$F_\beta^\omega\uparrow$	0.728	0.741	0.736	0.736	0.734	0.739	0.737	0.721	0.712	0.739	0.764	0.765	**0.772**
$M\downarrow$	0.070	0.072	0.070	0.071	0.071	0.072	0.071	0.073	0.071	0.068	0.067	0.068	**0.067**
$S_\alpha\uparrow$	0.826	0.833	0.828	0.823	0.821	0.833	0.832	0.811	0.795	0.849	0.855	0.861	**0.865**
$E_M^\phi\uparrow$	0.858	0.870	0.868	0.866	0.865	0.861	0.860	0.850	0.834	0.868	0.895	0.901	**0.903**
DIS-TE3 $maxF_1\uparrow$	0.830	0.846	0.845	0.839	0.833	0.838	0.834	0.831	0.812	0.841	0.850	0.867	**0.870**
$F_\beta^\omega\uparrow$	0.758	0.770	0.766	0.763	0.757	0.769	0.752	0.733	0.728	0.761	0.782	0.783	**0.785**
$M\downarrow$	0.064	0.066	0.066	0.065	0.066	0.066	0.065	0.066	0.065	**0.062**	**0.062**	**0.062**	0.063
$S_\alpha\uparrow$	0.836	0.848	0.844	0.839	0.832	0.844	0.837	0.823	0.819	0.842	0.859	0.866	**0.878**
$E_M^\phi\uparrow$	0.883	0.894	0.887	0.883	0.878	0.900	0.896	0.878	0.858	0.911	0.915	0.918	**0.926**
DIS-TE4 $maxF_1\uparrow$	0.827	0.833	0.826	0.820	0.816	0.830	0.818	0.804	0.801	0.856	0.876	0.876	**0.879**
$F_\beta^\omega\uparrow$	0.753	0.759	0.754	0.750	0.746	0.760	0.759	0.758	0.747	0.789	0.824	0.824	**0.830**
$M\downarrow$	0.072	0.074	0.073	0.073	0.075	0.074	0.072	0.076	0.074	0.071	**0.069**	0.071	0.072
$S_\alpha\uparrow$	0.830	0.846	0.839	0.838	0.833	0.841	0.830	0.822	0.815	0.842	0.852	0.860	**0.862**
$E_M^\phi\uparrow$	0.870	0.885	0.881	0.873	0.868	0.873	0.866	0.845	0.836	0.891	0.917	0.917	**0.923**

4.2 Impletement Details

Our image editing framework is implemented using PyTorch and trained on 8 NVIDIA GeForce RTX 3090 GPUs. We employ both non-rigid and rigid editing approaches to manipulate images in the DIS-TR dataset, utilizing spatial transformers and affine transformations to model non-rigid and rigid deformations, respectively. The hyperparameters used in our model are as follows: a batch size of 16, an image size of 512x512, 5 editing iterations, a learning rate of 0.001, a weight decay of 0.0001, 1000 diffusion steps, and a diffusion step size of 0.1. The hyperparameters in Equ 3 are set to $\lambda_1 = 0.8$ and $\lambda_2 = 0.5$.

We train the segmentation model using the DIS-TR subset of the DIS5K dataset, utilizing 2 NVIDIA GeForce RTX 3090 GPUs. The input size to the network is 512x512, with a learning rate of 0.0001 and a batch size of 48. The model is optimized using the Adam optimizer over a total of 800 epochs. The edited images generated by our model are combined with the original training set to form a new training set, enabling the model to learn from both original and edited images.

We evaluate our model on the DIS5K datasets, using the test sets from each dataset. For a comprehensive comparison, we also reproduce two state-of-the-art diffusion-based image generation methods, DatasetDM and Dataset Diffusion, using their default parameters. The generated images are compared with those from our approach, with the visualization of the results shown in Figure 3.

4.3 Results

Results by Generated Image Count.   We evaluate the performance of our proposed method, MaskFactory, against two state-of-the-art baselines, DatasetDM [9] and Dataset Diffusion [10], on the DIS5k dataset. The DIS5k dataset comprises five sub-datasets: DIS-VD, DIS-TE1, DIS-TE2, DIS-TE3, and DIS-TE4. We use the robust segmentation method IS-Net [2] as a baseline for the DIS task. For the two baselines, DatasetDM and Dataset Diffusion, we use the default parameters provided in their open-source implementations.

Table 2: Comparison of experimental results using different segmentation schemes, demonstrating performance gains with a fixed number of generated images.

	Method	IS-Net [2]		FP-DIS [36]		UDUN [25]		BiRefNet [1]		SAM-HQ [37]	
		w/o Ours	w Ours	w/o Ours	w Ours	w/o Ours	w Ours	w/o Ours	w Ours	w/o Ours	w Ours
DIS-TE1	$maxF_1 \uparrow$	0.740	0.784	0.784	0.805	0.784	0.799	0.866	0.882	0.897	0.905
	$M \downarrow$	0.074	0.073	0.060	0.063	0.059	0.057	0.036	0.033	0.019	0.018
	$S_\alpha \uparrow$	0.787	0.829	0.821	0.859	0.817	0.830	0.889	0.900	0.907	0.911
	$E_M^\phi \uparrow$	0.820	0.875	0.855	0.885	0.846	0.849	0.915	0.916	0.943	0.949
DIS-TE2	$maxF_1 \uparrow$	0.799	0.822	0.827	0.849	0.829	0.849	0.906	0.910	0.889	0.894
	$M \downarrow$	0.070	0.067	0.059	0.061	0.058	0.055	0.031	0.029	0.029	0.030
	$S_\alpha \uparrow$	0.823	0.865	0.845	0.862	0.843	0.866	0.913	0.921	0.883	0.889
	$E_M^\phi \uparrow$	0.858	0.903	0.889	0.893	0.886	0.894	0.947	0.957	0.928	0.937
DIS-TE3	$maxF_1 \uparrow$	0.830	0.870	0.868	0.911	0.865	0.888	0.920	0.937	0.851	0.853
	$M \downarrow$	0.064	0.063	0.049	0.051	0.050	0.047	0.029	0.027	0.045	0.043
	$S_\alpha \uparrow$	0.836	0.878	0.871	0.892	0.865	0.885	0.918	0.918	0.851	0.854
	$E_M^\phi \uparrow$	0.883	0.926	0.903	0.924	0.913	0.931	0.951	0.957	0.897	0.906
DIS-TE4	$maxF_1 \uparrow$	0.827	0.879	0.846	0.882	0.846	0.851	0.906	0.916	0.763	0.764
	$M \downarrow$	0.072	0.072	0.061	0.063	0.059	0.053	0.038	0.035	0.088	0.084
	$S_\alpha \uparrow$	0.830	0.862	0.852	0.856	0.849	0.873	0.901	0.911	0.799	0.806
	$E_M^\phi \uparrow$	0.870	0.923	0.891	0.935	0.891	0.895	0.933	0.941	0.843	0.850
DIS-VD	$maxF_1 \uparrow$	0.791	0.835	0.823	0.851	0.823	0.847	0.897	0.905	0.842	0.847
	$M \downarrow$	0.074	0.072	0.062	0.065	0.059	0.058	0.036	0.033	0.045	0.044
	$S_\alpha \uparrow$	0.813	0.866	0.843	0.873	0.838	0.857	0.905	0.909	0.848	0.850
	$E_M^\phi \uparrow$	0.856	0.923	0.873	0.880	0.876	0.901	0.931	0.941	0.896	0.903

First, we identify the best-performing model on the DIS-VD validation set and then evaluate its performance on the other sub-datasets. The models are trained on the DIS-TR dataset, which is augmented with generated datasets of varying sizes: 2500, 5000, 7500, and 10000 images.

The experimental results are presented in Table 1. Our proposed method, MaskFactory, consistently outperforms the baselines across all sub-datasets and evaluation metrics. As the number of generated images increases, the performance of MaskFactory improves, achieving the best results with 10000 generated images. Notably, MaskFactory attains the highest $maxF_1$ scores across all sub-datasets, with improvements ranging from 0.044 to 0.052 compared to the IS-Net baseline.

In contrast, while DatasetDM and Dataset Diffusion also show some performance gains, they encounter the issue of collapse when the generated data exceeds 5000 images. In these cases, the segmentation network's performance stagnates or even degrades. For the DIS segmentation task, the use of pseudo-labels could introduce additional errors, leading to performance declines.

Furthermore, MaskFactory demonstrates superior performance in terms of the weighted F-measure (F_β^ω), S-measure (S_α), and Enhanced-alignment Measure (E_M^ϕ). The Mean Absolute Error (M) is also consistently lower for MaskFactory compared to the baselines, indicating more accurate segmentation results.

Results by Segmentation Network. After achieving stable improvements in IS-Net's performance, we examined the generalizability of our approach by applying the same configuration to several state-of-the-art segmentation networks on the DIS5K dataset. Each network was trained using the DIS-TR dataset augmented with 10,000 generated image pairs. The networks considered in this study include FP-DIS [36], UDUN [25], BiRefNet [1], and SAMHQ [37], all implemented with their default parameters.

The experimental results, shown in Table 2, demonstrate that our proposed method consistently enhances the performance of all evaluated networks across the five sub-datasets of DIS5K. Notably, significant improvements were observed in the $maxF_1$ and E_M^ϕ metrics. For instance, on the DIS-TE1 sub-dataset, our method increased the $maxF_1$ score of IS-Net from 0.740 to 0.784, FP-DIS from 0.784 to 0.805, UDUN from 0.784 to 0.799, BiRefNet from 0.866 to 0.882, and SAMHQ from 0.897 to 0.905.
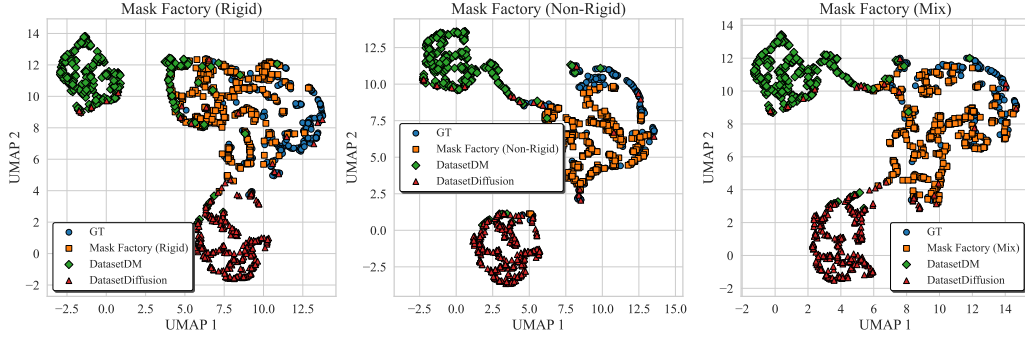
(a) Differences in UMAP distribution of generated mask images
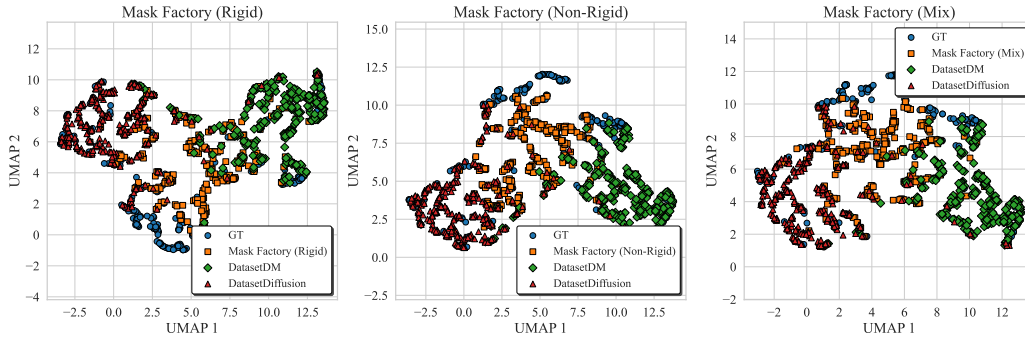
(b) Differences in UMAP distribution of generated corresponding RGB images

Figure 4: UMAP Distribution Differences

Additionally, the Mean Absolute Error (M) decreased for all networks when applying our method, indicating more accurate segmentation results. The S-measure (S_α) also consistently improved across all sub-datasets and networks, highlighting the effectiveness of our approach in capturing structural similarity.

On the DIS-VD sub-dataset, used for validation, our method boosted the $maxF_1$ score of IS-Net from 0.791 to 0.835, FP-DIS from 0.823 to 0.851, UDUN from 0.823 to 0.847, BiRefNet from 0.897 to 0.905, and SAMHQ from 0.842 to 0.847. These findings underscore the generalizability of our approach, as it enhances the performance of diverse segmentation networks without requiring any network-specific modifications.

Visual Results. As shown in Figure 5 (Appendix), our approach achieves precise results, comprising both rigid and non-rigid transformations. The non-rigid transformations, illustrated in columns (b) and (c), enable shape editing, such as removing a table corner or merging two backpacks into one. In contrast, the rigid transformations, demonstrated in columns (d), (e), and (f), primarily involve viewpoint changes, showcasing the original mask

Table 3: Similarity with original dataset.

Generate Type	CLIP	UMAP
DatasetDM [9]	0.6683	0.7017
Dataset Diffusion [10]	0.6217	0.6349
MaskFactory(rigid)	0.8791	0.8961
MaskFactory(non-rigid)	**0.9147**	**0.9346**
MaskFactory(All)	0.8967	0.9103

rotated in 3D space. Notably, our method effectively preserves the topological structure information of the original image, including the holes on the chair back. This allows for the low-cost generation of high-precision, diverse data pairs. To investigate differences between generated and real images, we analyzed image and mask distributions. Specifically, we used the CLIP model[38] to extract features from 300 real images, as well as images and masks generated by DatasetDM, DatasetDiffusion, and MaskFactory. We then applied UMAP[39] for dimensionality reduction on these 300 features. The feature distributions of masks and images are visualized in Figures 4(a) and 4(b), respectively.

In the mask editing domain, MaskFactory demonstrates superior mask fidelity compared to other generation methods, primarily due to the incorporation of topological consistency constraints. This results in a feature distribution that closely aligns with that of real images. Conversely, for RGB images, the prior from diffusion VAE introduces a larger disparity between the generated and real

image distributions. However, the distribution generated by MaskFactory shows a greater overlap with the real image distribution compared to the other two methods.

Furthermore, we quantified the differences using cosine similarity, as presented in Table 3. The results indicate that our method achieves the closest distribution to real images, further validating the effectiveness of MaskFactory in generating realistic masks and images.

5 Discussion

5.1 Ablation Study

Mask Generation Type Ablation. In our study, we implemented mask rigid editing, non-rigid editing, and mixed editing—each leveraging our novel mask control technology tailored for specific application scenarios. Rigid editing is designed for scenarios requiring precise geometric adjustments, primarily focusing on viewpoint and scale transformations. Non-rigid editing caters to applications

Table 4: Ablations on generation types.

Type	$maxF_1 \uparrow$	$M \downarrow$	$S_\alpha \uparrow$	$E_M^\phi \uparrow$
Rigid	0.768	0.074	0.807	0.867
Non-Rigid	0.771	0.074	0.796	0.858
Mix	0.784	0.073	0.829	0.875

needing high adaptability, handling topologically consistent deformations and complex, dynamic image edits. Mixed editing combines the advantages of both approaches, offering a comprehensive solution. We further evaluated the performance gains of each editing strategy. Our experimental results, as shown in Table 4.

From the table, we observe that mixed editing achieves the highest performance across most metrics. Specifically, it achieves the highest $maxF_1$ score and S_α, indicating superior structural fidelity and segmentation quality. The slight improvement in M and E_M^ϕ further underscores the versatility and effectiveness of mixed editing in creating diverse and realistic image-mask combinations.

Loss Function Ablation. We introduce content and structure losses into our model. The Discriminative Loss, implemented via a discriminator, evaluates the differences between generated and real images, aiming to enhance the realism and quality of the outputs. The Edge Constraint Loss focuses on maintaining edge coherence during image editing, which is critical for preserving detailed structural information. We conducted ablation experiments to evaluate the impact of each loss function. The experimental results, shown in Table 5.

From the table, we observe that the combination of all three loss functions ($\mathcal{L}_{\text{GAN}}$, $\mathcal{L}_{\text{content}}$, and $\mathcal{L}_{\text{structure}}$) achieves the highest $maxF_1$ score, indicating the best performance in terms of structural fidelity and realism.

5.2 Limitation

Despite the favorable outcomes achieved by our method, it still encounters significant issues. Although we experimented with different conditions, the results are shown in

Table 5: Ablations on loss functions.

Loss Function				
$\mathcal{L}_{\text{GAN}}$	$\mathcal{L}_{\text{content}}$	$\mathcal{L}_{\text{structure}}$	$maxF_1 \uparrow$	$M \downarrow$
✓			0.778	0.073
	✓		0.745	0.075
		✓	0.751	0.074
✓	✓	✓	0.784	0.073

Table 6. ControlNet sometimes produces unnatural images with stark foreground-background distinctions, necessitating additional harmonization. Complex scenarios can yield unrealistic elements, such as improperly positioned objects. Additionally, our method relies on pre-annotated image-mask pairs, limiting its ability to generate data autonomously and requiring high-quality initial annotations.

5.3 Conclusion

This paper introduces MaskFactory, a novel two-stage approach for generating high-quality synthetic datasets for DIS tasks. By combining rigid and non-rigid mask editing techniques and using multi-conditional control for image generation, MaskFactory produces diverse and precise synthetic image-mask pairs, significantly reducing dataset preparation time and costs. Experiments on the DIS5K benchmark demonstrate the superior performance

Table 6: Ablations on conditions.

Mask	Prompt	Canny	$maxF_1 \uparrow$	$M \downarrow$
✓			0.778	0.075
✓	✓		0.782	0.073
✓		✓	0.764	0.080
✓	✓	✓	0.784	0.073

of MaskFactory compared to existing methods in terms of quality and efficiency.

Acknowledgements

References

[1] Peng Zheng, Dehong Gao, Deng-Ping Fan, Li Liu, Jorma Laaksonen, Wanli Ouyang, and Nicu Sebe. Bilateral reference for high-resolution dichotomous image segmentation. *CAAI AIR*, 3:9150038, 2024.

[2] Xuebin Qin, Hang Dai, Xiaobin Hu, Deng-Ping Fan, Ling Shao, and Luc Van Gool. Highly accurate dichotomous image segmentation. In *ECCV*, 2022.

[3] Lian Liu, Han Zhou, Jiongquan Chen, Sijing Liu, Wenlong Shi, Dong Ni, Deng-Ping Fan, and Xin Yang. Instructive feature enhancement for dichotomous medical image segmentation. In *MICCAI*, 2023.

[4] Yinda Chen, Che Liu, Xiaoyu Liu, Rossella Arcucci, and Zhiwei Xiong. Bimcv-r: A landmark dataset for 3d ct text-image retrieval. In *MICCAI*, 2024.

[5] Yinda Chen, Wei Huang, Xiaoyu Liu, Shiyu Deng, Qi Chen, and Zhiwei Xiong. Learning multiscale consistency for self-supervised electron microscopy instance segmentation. In *ICASSP*, 2024.

[6] Chenfeng Xu, Bichen Wu, Zining Wang, Wei Zhan, Peter Vajda, Kurt Keutzer, and Masayoshi Tomizuka. Squeezesegv3: Spatially-adaptive convolution for efficient point-cloud segmentation. In *ECCV*, 2020.

[7] Xiaoyu Liu, Miaomiao Cai, Yinda Chen, Yueyi Zhang, Te Shi, Ruobing Zhang, Xuejin Chen, and Zhiwei Xiong. Cross-dimension affinity distillation for 3d em neuron segmentation. In *CVPR*, 2024.

[8] Yinda Chen, Wei Huang, Shenglong Zhou, Qi Chen, and Zhiwei Xiong. Self-supervised neuron segmentation with multi-agent reinforcement learning. In *IJCAI*, 2023.

[9] Weijia Wu, Yuzhong Zhao, Hao Chen, Yuchao Gu, Rui Zhao, Yefei He, Hong Zhou, Mike Zheng Shou, and Chunhua Shen. Datasetdm: Synthesizing data with perception annotations using diffusion models. *NeurIPS*, 2023.

[10] Quang Nguyen, Truong Vu, Anh Tran, and Khoi Nguyen. Dataset diffusion: Diffusion-based synthetic data generation for pixel-level semantic segmentation. *NeurIPS*, 2023.

[11] Yinda Chen, Che Liu, Wei Huang, Sibo Cheng, Rossella Arcucci, and Zhiwei Xiong. Generative text-guided 3d vision-language pretraining for unified medical image segmentation. *arXiv preprint arXiv:2306.04811*, 2023.

[12] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[13] Aysim Toker, Marvin Eisenberger, Daniel Cremers, and Laura Leal-Taixé. Satsynth: Augmenting image-mask pairs through diffusion models for aerial semantic segmentation. In *CVPR*, 2024.

[14] Lihe Yang, Xiaogang Xu, Bingyi Kang, Yinghuan Shi, and Hengshuang Zhao. Freemask: Synthetic images with dense annotations make stronger segmentation models. *NeurIPS*, 2023.

[15] Yinda Chen, Haoyuan Shi, Xiaoyu Liu, Te Shi, Ruobing Zhang, Dong Liu, Zhiwei Xiong, and Feng Wu. Tokenunify: Scalable autoregressive visual pre-training with mixture token prediction. *arXiv preprint arXiv:2405.16847*, 2024.

[16] Dingquan Wang and Jason Eisner. Synthetic data made to order: The case of parsing. In *EMNLP*, 2018.

[17] Shaobo Lin, Kun Wang, Xingyu Zeng, and Rui Zhao. Explore the power of synthetic data on few-shot object detection. In *CVPR*, 2023.

[18] Alvaro Figueira and Bruno Vaz. Survey on synthetic data generation, evaluation methods and gans. *Math.*, 10(15):2733, 2022.

[19] Aman Kishore, Tae Eun Choe, Junghyun Kwon, Minwoo Park, Pengfei Hao, and Akshita Mittel. Synthetic data generation using imitation training. In *CVPR*, 2021.

[20] Tung Nguyen, Sudhanshu Agrawal, and Aditya Grover. Expt: Synthetic pretraining for few-shot experimental design. *NeurIPS*, 2023.

[21] Yang Liu, Sujay Khandagale, Colin White, and Willie Neiswanger. Synthetic benchmarks for scientific research in explainable machine learning. In *NeurIPS Dataset Track*, 2021.

[22] 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. *arXiv preprint arXiv:2307.01952*, 2023.

[23] Qi Chen, Xiaoxi Chen, Haorui Song, Zhiwei Xiong, Alan Yuille, Chen Wei, and Zongwei Zhou. Towards generalizable tumor synthesis. *CVPR*, 2024.

[24] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023.

[25] Jialun Pei, Zhangjun Zhou, Yueming Jin, He Tang, and Pheng-Ann Heng. Unite-divide-unite: Joint boosting trunk and structure for high-accuracy dichotomous image segmentation. In *ACM MM*, 2023.

[26] Xuebin Qin, Zichen Zhang, Chenyang Huang, Chao Gao, Masood Dehghan, and Martin Jagersand. Basnet: Boundary-aware salient object detection. In *CVPR*, 2019.

[27] Qian Yu, Xiaoqi Zhao, Youwei Pang, Lihe Zhang, and Huchuan Lu. Multi-view aggregation network for dichotomous image segmentation. In *CVPR*, 2024.

[28] Qin Liu, Jaemin Cho, Mohit Bansal, and Marc Niethammer. Rethinking interactive image segmentation with low latency high quality and diverse prompts. In *CVPR*, 2024.

[29] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *CVPR*, 2023.

[30] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *CVPR*, 2023.

[31] Radhakrishna Achanta, Sheila Hemami, Francisco Estrada, and Sabine Susstrunk. Frequency-tuned salient region detection. In *CVPR*, 2009.

[32] Ran Margolin, Lihi Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps? In *CVPR*, 2014.

[33] Federico Perazzi, Philipp Krähenbühl, Yael Pritch, and Alexander Hornung. Saliency filters: Contrast based filtering for salient region detection. In *CVPR*, 2012.

[34] Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *ICCV*, 2017.

[35] Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment measure for binary foreground map evaluation. In *IJCAI*, 2018.

[36] Yan Zhou, Bo Dong, Yuanfeng Wu, Wentao Zhu, Geng Chen, and Yanning Zhang. Dichotomous image segmentation with frequency priors. In *IJCAI*, 2023.

[37] Lei Ke, Mingqiao Ye, Martin Danelljan, Yu-Wing Tai, Chi-Keung Tang, Fisher Yu, et al. Segment anything in high quality. *NeurIPS*, 36, 2023.

[38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[39] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[40] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Appendix

A Pseudocode for the MaskFactory Algorithm

MaskFactory is a two-stage approach for generating high-quality synthetic datasets for DIS tasks. In the first stage, existing ground truth masks undergo rigid and non-rigid editing to generate diverse synthetic masks. Rigid editing uses geometric priors from diffusion models for precise viewpoint transformations, while non-rigid editing employs adversarial training and self-attention mechanisms for complex shape modifications while preserving topology.

In the second stage, the generated masks and their corresponding Canny edges serve as conditions, along with category prompts, to guide the generation of high-resolution RGB images using a multi-conditional control generation method. This process ensures consistency between the generated images and masks while enhancing dataset realism and diversity. Our pseudocode visible Algorithm 1

Algorithm 1: MaskFactory Algorithm

Input : $D = \{(I_i^r, I_i^m)\}_{i=1}^N$ - Original dataset
Output: $G = \{(g_i^r, g_i^m)\}_{i=1}^M$ - Synthetic dataset

1 **Stage 1: Mask Editing**
2 **for** $i \leftarrow 1$ **to** N **do**
3   **Step 1.1: Rigid Mask Editing**
4      $I_i^{m'} \leftarrow \text{Invert}(I_i^m)$ // Invert source mask
5      $g_i^m \leftarrow \psi_\theta(I_i^{m'}, T_i)$ // Apply viewpoint transformation
6   **Step 1.2: Non-Rigid Mask Editing**
7      $E_s \leftarrow E(I_i^m)$ // Extract edge map from source mask
8      $\mathcal{V} \leftarrow \{v_j\}_{j=1}^{N_v}$ // Obtain key points from edge map
9      $\mathcal{T}_s \leftarrow (V, E_s)$ // Construct source mask structural graph
10     $g_i^m \leftarrow G_\theta(z, P_i, I_i^m)$ // Generate synthetic mask
11     $T_g = (\mathcal{V}, \mathcal{E}_s)$ // Construct synthetic mask structural graph
12     $\mathcal{L}_{GAN} \leftarrow \mathbb{E}_{T_s \sim p_{data}(T_s)}[\log D_\phi(T_s)] + \mathbb{E}_{T_g \sim p_{gen}(T_g)}[\log(1 - D_\phi(T_g))]$
13     $\mathcal{L}_{content} \leftarrow \|g_i^m - I_i^m\|_1$
14     $\mathcal{L}_{structure} \leftarrow \|T_g - T_s\|_1$
15     $\mathcal{L}_{total} \leftarrow \mathcal{L}_{GAN} + \lambda_1 \mathcal{L}_{content} + \lambda_2 \mathcal{L}_{structure}$
16   Update G_θ and D_ϕ to minimize $\mathcal{L}_{total}$

17 **Stage 2: Image Generation**
18 **for** $i \leftarrow 1$ **to** M **do**
19     $c_i^s \leftarrow g_i^m$ // Segmentation condition
20     $c_i^y \leftarrow \text{Canny}(g_i^m)$ // Canny condition
21     $z \sim \mathcal{N}(0, 1)$ // Sample Gaussian noise
22     $g_i^r \leftarrow M_\theta(z, B_\theta(c_i^s), B_\theta(c_i^y))$ // Generate RGB image

23 **return** $G = \{(g_i^r, g_i^m)\}_{i=1}^M$

B Dataset Details

We conduct our experiments on the DIS5K dataset, which comprises 5,479 high-resolution images featuring camouflaged, salient, or meticulous objects in various backgrounds. The DIS5K dataset is divided into three subsets: DIS-TR (3,000 images) for training, DIS-VD (470 images) for validation, and DIS-TE (2,000 images) for testing. For data augmentation, we utilize the mask portion of the training subset (DIS-TR).

To evaluate our models, we employ a diverse set of metrics to ensure comprehensive performance assessment. These metrics include max F_1, which balances precision and recall, providing a harmonic mean that is indicative of overall accuracy; F_β^ω, a weighted F-measure that compensates for class

imbalances, with values ranging from 0 to 1, where higher values denote superior performance; M (Mean Absolute Error), which calculates the average absolute difference between the predicted and ground truth masks, with lower values signifying better accuracy; S_α, a structural similarity measure that evaluates the preservation of significant structures within the image, with values closer to 1 indicating better performance; and E_M^ϕ, an enhanced measure that considers both pixel-level and image-level information for a more holistic evaluation, where higher values represent better performance. Collectively, these metrics provide a robust framework for assessing the effectiveness and reliability of our segmentation models.

C Mathematical Details

C.1 Diffusion-Based Image Generation

The diffusion model employed in MaskFactory for image generation follows the formulation introduced by Ho et al. [40]. Given a data distribution $x_0 \sim q(x_0)$, the forward diffusion process is defined as a Markov chain that gradually adds Gaussian noise to the data:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t \mathbf{I}), \tag{9}$$

where $\beta_t \in (0, 1)$ is a variance schedule. The reverse process is learned by a neural network ϵ_θ that predicts the noise added at each step:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma_t^2 \mathbf{I}), \tag{10}$$

where $\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(x_t, t) \right)$ and $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$.

The objective is to maximize the **variational lower bound**:

$$\mathcal{L} = \mathbb{E}_{q(x_0)} \mathbb{E}_{q(x_1,...,x_T|x_0)} \left[\sum_{t=1}^T \log \frac{p_\theta(x_{t-1}|x_t)}{q(x_{t-1}|x_t, x_0)} \right]. \tag{11}$$

C.2 Topology-Preserving Adversarial Training

The **topology-preserving adversarial training** in **MaskFactory** involves a generator G_θ and a discriminator D_ϕ. The generator aims to minimize the adversarial loss:

$$\mathcal{L}_{\text{GAN}}(G) = \mathbb{E}_{T_g \sim p_{\text{gen}}(T_g)}[\log(1 - D_\phi(T_g))], \tag{12}$$

while the discriminator tries to maximize the adversarial loss:

$$\mathcal{L}_{\text{GAN}}(D) = \mathbb{E}_{T_s \sim p_{\text{data}}(T_s)}[\log D_\phi(T_s)] + \mathbb{E}_{T_g \sim p_{\text{gen}}(T_g)}[\log(1 - D_\phi(T_g))]. \tag{13}$$

The **generator** and **discriminator** are updated alternately to reach an equilibrium.

C.3 Structural Graph Construction

To preserve the topological structure of the source masks during editing, **MaskFactory** constructs structural graphs T_s and T_g for the source and synthetic masks, respectively. The structural graph $T = (V, E)$ consists of a set of vertices $V = \{v_j\}_{j=1}^{N_v}$ representing key points and a set of edges E representing their connectivity.

The structure preservation loss is defined as the L1 distance between the structural graphs:

$$\mathcal{L}_{\text{structure}} = \|T_g - T_s\|_1 = \sum_{(i,j) \in E} \|T_g(i,j) - T_s(i,j)\|, \tag{14}$$

where $T_g(i, j)$ and $T_s(i, j)$ denote the edge weights between vertices i and j' in the synthetic and source mask structural graphs, respectively.

By minimizing the **structure preservation loss** along with the adversarial and content losses, **MaskFactory** ensures that the edited masks maintain the topological structure of the source masks.

D Visualization of the results using two editing methods from MaskFactory.

In this section, we demonstrate MaskFactory's ability to edit masks for common and fine-grained objects.

D.1 Visualization of Common Object Mask Editing

We selected common household items such as tables, chairs, bags, and musical instruments for editing using MaskFactory. Non-rigid edits were performed with different prompts, and rigid edits were performed from different viewpoints. The editing results and corresponding RGB image generations are shown in Figure 5. MaskFactory exhibits strong topological structure preservation and produces diverse editing outcomes, as demonstrated by the variety of modifications made to both rigid and non-rigid objects.

D.2 Visualization of Fine-Grained Object Mask Editing

We selected fine-grained objects from MaskFactory's generated masks, such as ornate European chandeliers, iron gates, birdcages, and seahorses, to showcase MaskFactory's detailed editing capabilities. Even with complex geometries, MaskFactory can perform topology-preserving edits without losing the original mask's semantic information. These intricate structures play a crucial role in segmentation metrics. As illustrated in Figure 6, the visual results demonstrate the model's ability to handle fine-grained details while maintaining the integrity of the original mask's structure.

D.3 Visualization Results with Canny Constraints

After incorporating Canny edge detection as a constraint, the visual results of MaskFactory show a significant improvement in boundary precision. The Canny edges effectively guide the generation process, resulting in images with clearer and more accurate boundary details, avoiding vague or ambiguous transition areas. The visualizations demonstrate that the Canny edges not only better constrain the contours of the generated images but also enhance the overall fidelity and visual quality of the output. Compared to models without edge constraints, our approach produces more detailed and structurally coherent images, as shown in Figure 7.

D.4 Topological Structure Visualization

We performed topological structure visualizations on selected samples to assess the model's ability to preserve topology during the editing process. These visualizations clearly show how our method retains the topological structure of the original data while allowing for effective manipulation when necessary. Whether dealing with complex geometric shapes or subtle structural modifications, the topological visualizations illustrate that our approach reliably preserves geometric consistency and topological features throughout the editing process, as demonstrated in Figure 8.

(a) raw image   (b) rigid 1   (c) rigid 2   (d) non-rigid 1   (e) non-rigid 2   (f) non-rigid 3

Figure 5: Visual results of common object mask editing. The model demonstrates strong topological structure preservation and diverse editing outcomes with both rigid and non-rigid edits.
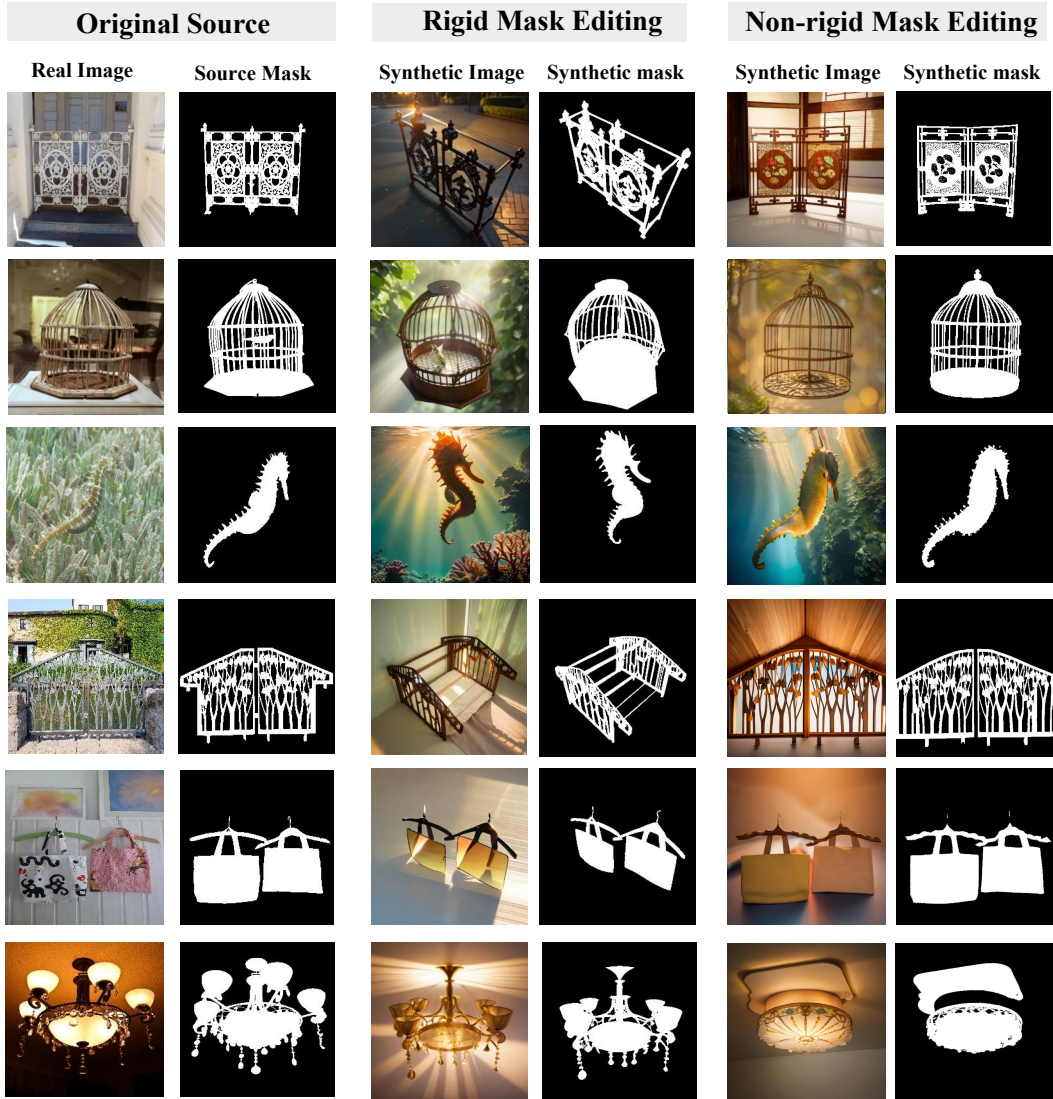
	Original Source		Rigid Mask Editing		Non-rigid Mask Editing	
	Real Image	Source Mask	Synthetic Image	Synthetic mask	Synthetic Image	Synthetic mask

Figure 6: Visual results of fine-grained object mask editing. The model successfully edits complex structures without compromising the original mask's semantic information.
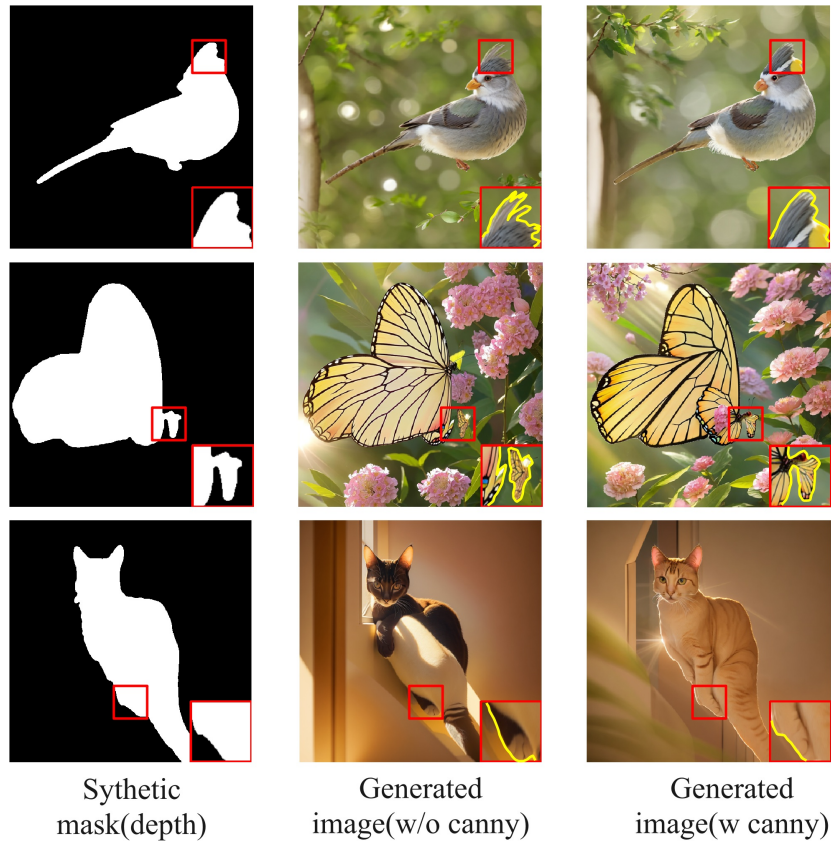
| Sythetic
mask(depth) | Generated
image(w/o canny) | Generated
image(w canny) |

Figure 7: Canny condition visual results. The generated images show improved boundary precision and better structural coherence.

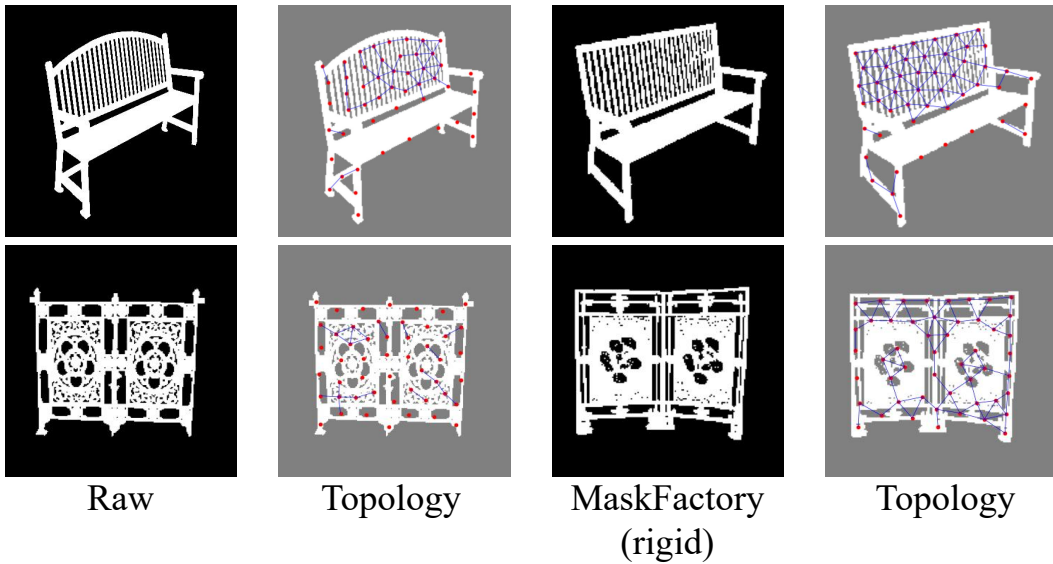| Raw | Topology | MaskFactory
(rigid) | Topology |

Figure 8: Topological Structure Visualization. The visualizations demonstrate the model's ability to maintain and manipulate topology during the editing process.

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: This paper presents a method for generating and editing images with high-precision masks, and announces the open-sourcing of a DIS generation dataset.

   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: Please refer to Section 5.2.

   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: This paper's experiments and methods do not introduce new theories; the theoretical foundations used are detailed in Section C.

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: Please refer to Section 4.2.

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: [Yes]

Justification: This paper includes a link to the anonymized code in the abstract.

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 run 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, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Please refer to Section 4.2.

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: [Yes]

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: Please refer to Section 4.2.

   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: please check in our project page.

   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: [NA]

    Justification: please check in our project page.

    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: [NA]

Justification: please check in our project page.

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: please check in our project page.

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: please check in our project page.

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: please check in our project page.

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: please check in our project page.

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.