FashionR2R: Texture-preserving Rendered-to-Real Image Translation with Diffusion Models

Rui Hu[1*†] , **Qian He**[2†] , **Gaofeng He**[2] , **Jiedong Zhuang**[1] ,
Huang Chen[2] , **Huafeng Liu**[1‡] , **Huamin Wang**[2]
[1]Zhejiang University, [2]Style3D Research
https://rickhh.github.io/FashionR2R/

Abstract

Modeling and producing lifelike clothed human images has attracted researchers' attention from different areas for decades, with the complexity from highly articulated and structured content. Rendering algorithms decompose and simulate the imaging process of a camera, while are limited by the accuracy of modeled variables and the efficiency of computation. Generative models can produce impressively vivid human images, however still lacking in controllability and editability. This paper studies photorealism enhancement of rendered images, leveraging generative power from diffusion models on the controlled basis of rendering. We introduce a novel framework to translate rendered images into their realistic counterparts, which consists of two stages: Domain Knowledge Injection (DKI) and Realistic Image Generation (RIG). In DKI, we adopt positive (real) domain finetuning and negative (rendered) domain embedding to inject knowledge into a pretrained Text-to-image (T2I) diffusion model. In RIG, we generate the realistic image corresponding to the input rendered image, with a Texture-preserving Attention Control (TAC) to preserve fine-grained clothing textures, exploiting the decoupled features encoded in the UNet structure. Additionally, we introduce SynFashion dataset, featuring high-quality digital clothing images with diverse textures. Extensive experimental results demonstrate the superiority and effectiveness of our method in rendered-to-real image translation.

1 Introduction

Modeling and simulating digital humans and clothing has achieved significant progress [1, 2, 3, 4, 5], while leveraging these 3D assets for fashion e-commerce still remains a challenging problem. Due to the imperfection of 3D models and the approximation in rendering algorithms, rendered images cannot yet replace fashion photos taken by a camera, with deficiency in the realism of rendered human faces and skin, clothing shape and fabric, etc. This paper studies transferring rendered fashion images into their realistic counterparts, which is inherently an Image-to-Image (I2I) translation problem.

Existing works on improving the realism of rendered images mainly resort to retrieving and blending real image patches [6], or train a GAN-based network [7, 8, 9] due to lack of paired training data. Another line of works can tackle this problem as general I2I translation [10, 11, 12, 13]. However, these methods may still suffer from several limitations: Firstly, their image generation pipelines have limited power to utilize real image resources for highly-detailed enhancement and may suffer from instability and mode collapse from adversarial training. Moreover, they either focus on indoor/outdoor

[*] Work done during an internship at Style3D Research.

[†] These authors contributed equally to this work.

[‡] Corresponding author.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

scene enhancement while keeping coarse object-level semantic layout, or try to maintain face identity in training through loss constraints on sketches, and thus have difficulty in preserving fine-grained texture in clothing images.

In this paper, we propose a novel framework based on diffusion models for rendered-to-real fashion image translation to address above limitations. Our main idea consists of two aspects: Firstly, we propose to leverage abundant generative prior from pretrained Text-to-Image (T2I) diffusion models [14], and apply simple adaptation to realistic image generation under the guidance of distilled rendered prior. Secondly, we adopt a texture-preserving mechanism by extracting spatial image structure through attention from an inversion pipeline.

To achieve this, we design a diffusion-based method consisting of two stages: Domain Knowledge Injection (DKI) and Realistic Image Generation (RIG). During DKI, we first finetune a pretrained T2I diffusion model [14] on real fashion photos with derived captions from BLIP [15], to adapt its capability in generating high-quality images to our target domain. After this adaptation, we propose to guide the image generation towards the negative direction of rendered effect. Inspired by Textual Inversion [16], we distill a general rendered "concept" with thousands of rendered fashion images by training a negative domain embedding vector based on the adapted base model. During RIG, we employ a DDIM inversion [17] pipeline to first invert a rendered image into the latent noise map, and then generate its corresponding real image using the previous embedding as a negative guidance [18]. Similar to recent training-free controls in T2I generation method [19, 20, 21, 22], we discover that the attention map in the shallow layers of the UNet contains rich spatial image structure and can be used for fine-grained texture-preserving during the generation. Specifically, we inject query and key of the self-attention from the rendered image inversion and generation pipeline to the rendered-to-real image generation pipeline. This largely improves the consistency of intricate clothing texture details.

We evaluate our method on a public rendered Face Synthetics dataset [1] and our collected SynFashion Dataset with fine-grained digital clothing and abundant texture variations. Empirical results comparing to previous works and experimental analysis demonstrate the efficacy of our method. Our main contributions are three-folds:
(1) We propose a novel framework to address rendered-to-real fashion image translation by utilizing generative prior from pretrained diffusion models.
(2) We inject rendered-to-real domain knowledge into a pretrained T2I diffusion model through positive domain finetuning and negative domain embedding, and design a texture-preserving attention control to preserve fine-grained clothing textures during the translation.
(3) We collect a high-quality rendered fashion image dataset using the professional design software Style3D Studio, and plan to release the data with our paper to promote research in this important area.

2 Related Works

2.1 Rendered-to-real Image Translation

Improving the realism of rendered images has been a long-standing problem due to the inherent limitations of rendering pipelines and the rich potential for commercial applications. CG2Real [6] proposes to retrieve similar images from a large collection of real photos and then applies local style transfer to upgrade color, tone and texture of the CG image. Deep CG2Real [7] adopts a two-stage deep learning framework to first transfer OpenGL images to PBR (Physically-Based Rendering) images, and then translates PBR to real images, disentangling lighting and texture in a CycleGAN-like [23] framework. [8] enhances photorealism under the guidance of a set of input G-buffers and learns the network with a perceptual discriminator. [9] proposes to learn a rendered image generator for human faces, which can encode the same face identity but different "style" from a real face image generator, based on StyleGAN [24, 25]. These methods all utilize limited data for generative training, while we propose to adapt diffusion models pretrained on large datasets for better image generation quality. Besides, applying these methods to fashion images often leads to the failure to preserve fine-grained clothing textures.

2.2 Image-to-image Translation

Transferring a rendered fashion image into its realistic counterpart is inherently an image-to-image (I2I) translation problem, which has attracted wide interest in different realms of research [26, 27, 28,

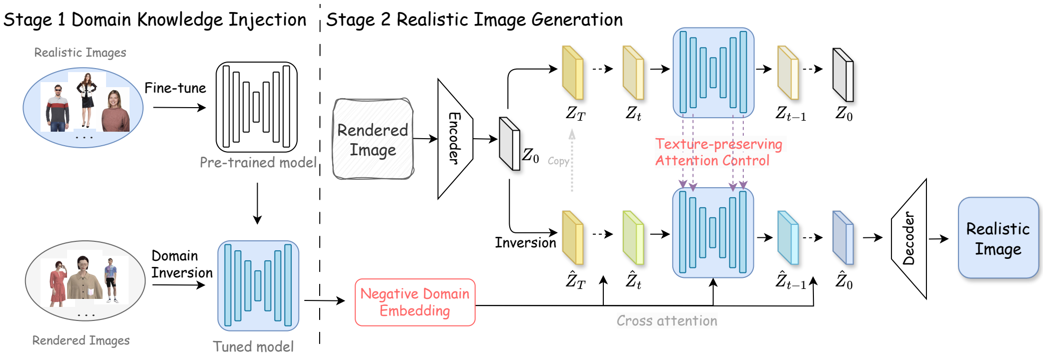

Figure 1: The overall pipeline of our proposed method.

29, 30, 31]. Pix2Pix [32] utilizes a conditional-GAN [33, 34] and applies pixel-wise regularization based on paired training data, which is unavailable in many problem settings. Cycle-GAN [23] proposes to utilize cycle consistency [35, 36, 37, 38] and optimizes a two-sided mapping between input source domain and output target domain. CUT [10] addresses the computational redundancy and over-restriction in this framework by simplifying it to one-sided [39, 40, 11] and introduces a patch-wise contrastive loss [41, 42, 43] for refined local constraints. UNSB [12] proposes an iterative refinement method based on Schrödinger bridge to overcome potential mode collapse in GAN generation, while still has difficulty in faithfully translating high-resolution images. Different from general I2I tasks and domain adaptation, our method focuses on photorealism enhancement and can utilize more target-domain real photos for high-quality generation training, and thus can deal with imbalanced source-target training set. Style transfer [44, 45, 46] is a specific type of I2I task and can manage to transfer input source image to an arbitrary style [13, 47, 48, 49, 50] given one/few-shot target domain images as reference. These methods mainly focus on transferring style attributes like semantics, brushstrokes, colors, or material, while rendered-to-real requires preserving and enhancing complicated fine-grained details. Human/portrait relighting [51, 52] modifies the nuanced lighting condition in the input image, while does not focus on enhancing realism and should leave geometry and materials untouched. Super-resolution methods [53, 54, 55, 56, 57, 58] address detail enhancement, while their success largely relies on synthesizing pseudo low-resolution images to obtain training pairs [59, 60], which is non-trivial for rendered-to-real problem.

2.3 Diffusion-based Image Synthesis

Recent progress in Text-to-Image (T2I) generation [14, 61, 62] based on diffusion models [63, 64, 65] opens up new opportunity for advancing rendered-to-real image translation. Many works have explored the possibility of utilizing abundant generative prior in pretrained diffusion models. Some [66, 16] apply the adaptation of generation for a new concept with a few images, through either finetuning the base model [66], or optimizing a text embedding [16]. Others [67, 68] leverage text as guidance to edit a given image. However, rendered-to-real translation lies in the nuance of changes, which is too subtle to define as a "concept" or to capture with a few images. [69, 70] leverage diffusion models for texture estimation or PBR synthesis, while mainly focusing on the generation of certain variables for the rendering pipeline, rather than subtle modification of preset variables in a given input image. Additionally, [19, 20, 21, 22] discover that the attention in the SD UNet captures rich image features and can apply to content preservation and modification. In our work, we utilize self-attention in shallow layers from the rendered image inversion, to impose the consistency of fine-grained texture in image translation.

3 Method

3.1 Preliminaries

Latent Diffusion Models. In diffusion framework, the forward diffusion process begins by generating noisy images x_t from clean images x_0 sampled from a specified data distribution, accompanied by their respective noise labels ϵ. These pairs are used to train a score estimator [71] ϵ_θ usually based on the UNet architecture. The score estimator can serve as an effective approximation of the score function $\nabla \log p(x)$ which directs the inverse denoising process to generate new data samples.

With distinguished capabilities in synthesizing images, the Latent Diffusion Model (LDM) [14] is selected as the backbone of our method. The LDM employs a pre-trained AutoEncoder to transform the diffusion process from pixel space to latent space and integrates a conditional branch, facilitating faster training and more flexible embedding of conditions. Specifically, the pre-trained Encoder $\mathcal{E}(\cdot)$first encodes images into latent space $z = \mathcal{E}(x)$. Following this, the score estimator network ϵ_θ is trained by taking the latent z, step t and conditions c as input to predict the noise labels:

$$\min_\theta \mathbb{E}_{z=\mathcal{E}(x),\epsilon\sim\mathcal{N}(0,I),t\sim\mathrm{U}(1,T)} \left\|\epsilon - \epsilon_\theta\left(z_t, t, c\right)\right\|_2^2 \tag{1}$$

For text to image generation task, condition c is usually the text embedding generated from text prompt y through a tokenizer and a pretrained CLIP [72] model $c = \tau(y)$. The intermediate noisy latent z_t is generated through the formula [64]:

$$z_t = \sqrt{\bar{\alpha}(t)} z_0 + \sqrt{1 - \bar{\alpha}(t)}\epsilon, \epsilon \sim N(0, I) \tag{2}$$

$\bar{\alpha}$ is the cumulative product of the noise coefficients at each step. During the sampling process, the trained score estimator takes random Gaussian noise as input, along with text embedding as condition. It progressively predicts the noise added at each step, completing the denoising process to obtain $\hat{z}_0$. The final image is obtained by the pretrained decoder $\hat{x}_0 = \mathcal{D}(\hat{z}_0)$.

Textual Inversion. Textual inversion [16] introduces a new paradigm to T2I generation models, allowing the model to learn a new concept by setting a placeholder token "[C]" and obtaining the corresponding text embedding $\hat{v}$ as a learnable vector. This vector is then trained and optimized using a few images represent this new concept:

$$\hat{v} = \arg\min_v \mathbb{E}_{z=\mathcal{E}(x),\epsilon\sim\mathcal{N}(0,I),t\sim U(1,T)} \left\|\epsilon - \epsilon_\theta(z_t, t, v)\right\|_2^2 \tag{3}$$

During training, the network parameters are all fixed, only the embedding is optimized.

DDIM Sampling and Inversion. Inversion is an effective method for finding the corresponding noise map of an image and achieving training-free control during the generation process. DDIM inversion is widely used due to its clear principles and easy implementation. The DDIM sampling process is [17]:

$$z_{t-1} = \sqrt{\bar{\alpha}_{t-1}} \frac{z_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta\left(z_t, t, c\right)}{\sqrt{\bar{\alpha}_t}} + \sqrt{1-\bar{\alpha}_{t-1}}\epsilon_\theta\left(z_t, t, c\right) \tag{4}$$

By simply assuming $z_{t-1} \approx z_t$ and rewriting the sampling process in reverse direction, the following DDIM Inversion [17] formula is given:

$$z_t = \sqrt{\bar{\alpha}_t} \frac{z_{t-1} - \sqrt{1-\bar{\alpha}_{t-1}}\epsilon_\theta\left(z_{t-1}, t, c\right)}{\sqrt{\bar{\alpha}_{t-1}}} + \sqrt{1-\bar{\alpha}_t}\epsilon_\theta\left(z_{t-1}, t, c\right) \tag{5}$$

Unlike direct noise addition, the DDIM Inversion allows for the original information of the image to be well preserved, enhancing the stability in the subsequent generation process.

3.2 Overall Pipeline

Given a computer-rendered fashion image x_{cg}, the goal of our method is to transform it into a corresponding realistic image x_r while preserving the garment's detailed textures. Defining realism and helping model understand what is "realistic" remains an open question. The challenge can be divided into two sub-tasks: one is making the fashion image appear realistic by enhancing aspects like wrinkles, lighting and color, which reflect true-to-life expressions. Another one is to maintain the texture details of the garment to achieve fine-grained, controllable generation.

As shown in Fig. 1, our method comprises two stages: Domain Knowledge Injection (DKI) and Realistic Image Generation (RIG). During the DKI phase, we infuse the model with information from both the rendered and realistic domains through fine-tuning and domain inversion. In the subsequent generation phase, we utilize negative domain embedding v_{nd} to stimulate the model's potential for generating realistic images and employ self-attention control to preserve texture details. For a better understanding, details will be further elaborated in Section 3.3 and Section 3.4.

3.3 Domain Knowledge Injection

Target Domain Knowledge Injection To enhance the ability of the base SD model ϵ_θ to generate realistic images, especially concerning the appearance of garments and models, we use real studio-shot images x_{tr} to fine-tune the base model. This process injects real domain information into the model, thereby increasing its potential to generate authentic visual details, the fine-tuning process can be formulated as:

$$\epsilon_\theta^* = \arg\min_{\epsilon_\theta} \mathbb{E}_{z=\mathcal{E}(x_{tr}),\epsilon\sim\mathcal{N}(0,I),t\sim U(1,T)} \left\|\epsilon - \epsilon_\theta(z_t, t, v_{tr})\right\|_2^2 \tag{6}$$

where ϵ_θ is the pretrained SD model, v_{tr} is the embedding of the text description of the x_{tr}.

Source Domain Knowledge Injection For the source domain rendered data, we hope that the model can understand its characteristics and deviated from the rendered data manifold as much as possible during the generation process. After the first step fine-tuning, we assume that the model has already enhanced its representation of the real domain manifold. If we can make the model deviate from the rendered data manifold, it can better express the characteristics of realistic images. Inspired by the concepts of Textual Inversion and Classifier-Free Guidance (CFG) with negative prompts, we expand the concept of Textual Inversion to Domain Inversion. We train a negative domain embedding on a fine-tuned base model using a large number of rendered images. This negative domain embedding guides the model to avoid certain content, here is the rendered domain characteristics, during the generation process.

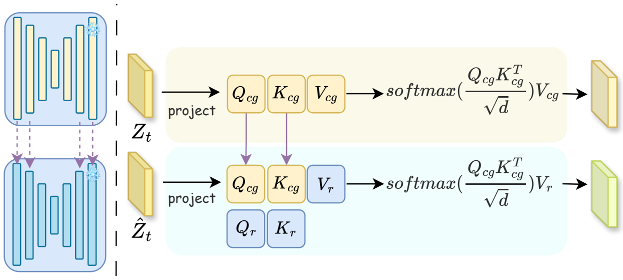
project
Q_{cg}
K_{cg}
V_{cg}
$softmax(\frac{Q_{cg}K_{cg}^T}{\sqrt{d}})V_{cg}$
Z_t
project
Q_{cg}
K_{cg}
V_r
$softmax(\frac{Q_{cg}K_{cg}^T}{\sqrt{d}})V_r$
Q_r
K_r
$\hat{Z}_t$

Figure 2: The diagram of Texture-preserving Attention Control (TAC).

Specifically, given that textual descriptions of what is real and rendered are limited, it is difficult to guide the model to generate images with satisfactory realism or to precisely direct it not to produce images with a rendered feel using text prompts only. Therefore, we consider using negative domain embedding v_{nd} trained on a large number of rendered images for guidance to inject the rendered domain knowledge to the model. It's worth nothing that unlike textual inversion, which typically optimizes a small embedding space with few images to represent a specific concept, such as a particular object in personalized generation or an easily expressible style. The concept of rendered domain in our task is much more general. Using a small embedding space corresponding to few images to represent this would easily lead to over-fitting to the content of the training images. We use the largest available embedding size to train the negative domain embedding, which is corresponding to the placeholder token size of 75:

$$\hat{v}_{nd} = \arg\min_{v} \mathbb{E}_{z=\mathcal{E}(x_{cg}),\epsilon\sim\mathcal{N}(0,I),t\sim U(1,T)} \left\|\epsilon - \epsilon_\theta^*(z_t, t, v)\right\|_2^2 \tag{7}$$

During the training of negative domain embedding, we freeze the parameters in the fine-tuned model ϵ_θ^*, and find the v_{nd} through direct optimization with a certain number of rendered images.

3.4 Realistic Image Generation

Negative Embedding Guidance After domain knowledge injection, we can use the negative domain embedding to guide the model in generating realistic images. During each denoising step, the negative domain embedding guidance is defined by:

$$\tilde{\epsilon}_\theta^*\left(z_t, t, v_{nd}\right) = w \cdot \epsilon_\theta^*\left(z_t, t, v_\varnothing\right) + (1-w) \cdot \epsilon_\theta^*\left(z_t, t, v_{nd}\right) \tag{8}$$

where $v_\varnothing$ denotes the embedding of Null text. With a guidance scale w larger than 1, the negative domain embedding becomes effective. Unlike traditional CFG guidance, here we do not use any positive prompts processed through CLIP to obtain the embedding as conditions. Instead, we directly

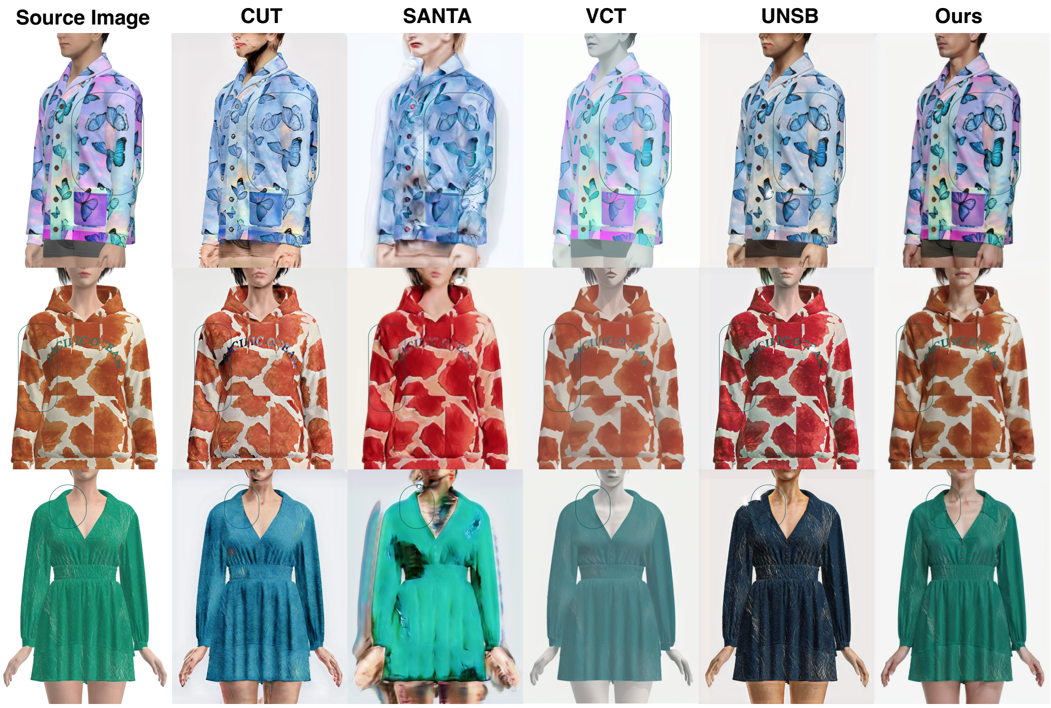

Figure 3: Results on our proposed SynFashion Dataset. (**Please zoom in for details.**)

employ a Null text embedding. The initial noise latent is obtained through DDIM inversion of the given rendered image. During the denoising process, we replace the CLIP conditioning branch, since the negative domain embedding is trained on a fine-tuned model, it can interact more effectively with the base model's latent space. This consistency allows for more precise adjustments in the latent manifold compared to embedding derived from text via CLIP.

Texture-preserving Attention Control (TAC) Inspired by previous work [19, 22], the attention features in the diffusion UNet, which includes both cross attention and self attention, hold rich information critical for generating the new images. Cross attention typically handles the attributes and semantics of the generated image, while self-attention maps play a crucial role in preserving geometric shapes and intricate details. The initial noise latent $\hat{Z}_t$ derived from the DDIM inversion of the original rendered image can be used in unconditional generation and extract the texture related attention features as shown in Fig. 2. However, directly replacing all self-attention maps can lead to a decrease in the realism of the generated images. We argue that this is because the attention map contains both the texture details of the garment and the general rendered domain characteristics. Therefore, we propose to control the self attention feature only in the shallow layers of the denoising UNet to decouple the texture details feature from the general rendered domain features. During the implementation, we also find that in the deep feature spaces with higher downsampling rates, it becomes challenging to identify features related to the texture details. Thus, our TAC is defined as:

$$\widehat{Q^t}, \widehat{K^t} = TAC\left(Q^t_{cg}, K^t_{cg}, Q^t_r, K^t_r, t\right) = \begin{cases} Q^t_{cg}, K^t_{cg} & \text{if } t < \gamma T, f > F \\ Q^t_r, K^t_r & \text{otherwise} \end{cases} \tag{9}$$

where γ is the parameter that indicates how many steps before the TAC should be applied and f is the feature size of different layers, only those layers exceeding the specified size F undergo TAC, particularly in the shallow layers. Specifically, the cg-domain self-attention features are derived from the reverse sampling process starting from the noisy latent, which is obtained by performing DDIM inversion on the input image latent. In contrast, the r-domain self-attention features differ due to the incorporation of negative domain guidance and the self-attention injection.

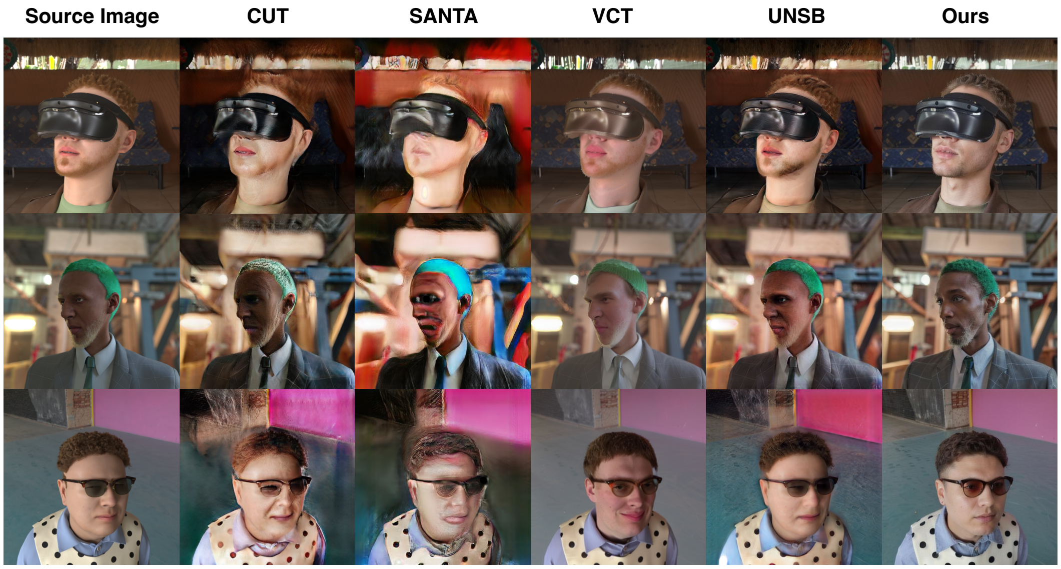

Figure 4: Results on the Face Synthetics dataset. (**Please zoom in for details.**)

4 Experiments

4.1 Datasets

To evaluate our method and conduct comprehensive comparisons, we introduce a high-quality rendered fashion image dataset, named Synthetic Fashion (SynFashion), with the professional garment design software Style3D Studio. SynFashion consists of 10k rendered images in 20 categories, including pants, T-shirt, lingerie and swimwear, half skirt, hoodie, coat, jacket, set, home-wear, hat, Hanfu, jeans, shorts, down jacket, vest and camisole, shirt, suit, dress, sweater and trench coat. For each category, we use Style3D Studio to build 10 to 40 projects in different 3D geometry with corresponding texture and design, and then randomly sample several new textures to change its appearance. There are overall 375 projects in 3D and 500 additional texture collected from Internet. For each textured 3D geometry, we render four views, including front, back, and two randomly sampled views. After rendering, we crop the enlarged garment area of each image and resize it to 768 × 1024. Due to legal issues, some of the images contain a digital human figure but not the complete face. To supplement the evaluation on rendered human faces, we also conduct experiments on the public available Face Synthetics dataset [1] with its first 10k images.

4.2 Implementation Details

Implementation. We implement our method with pretrained Stable Diffusion (SD) model and finetune the base model with 2500 realistic images at a 1024×1024 resolution for source domain knowledge injection. The finetuning uses images from iMaterialist (Fashion) 2019 FGVC dataset [73], based on the publicly available SD v1.5, and is conducted on 2 RTX 4090 with a batch size of 6. Based on the finetuned model, we train our negative domain embedding with 2500 rendered images on a single RTX 4090 with a batch size of 1. The rendered images are resized to the resolution of 512×512. The placeholder embedding size is 75 and the learning rate is 5e-4. During sampling, we perform DDIM sampling with default 50 denoising steps with a denoising strength of 0.3 as default. The γ is set to 0.9 as default, which means that the TAC is performed on the first 90% of sampling steps. Only the attention maps in the first and second shallow layers are used for TAC. Note that the denoising strength and γ may be changed to obtain different level of image translation. We compare our method with three state-of-the-art unpaired image-to-image translation method, CUT, SANTA and UNSB, and one diffusion-based style transfer method VCT. For CUT, SANTA and UNSB, we train the models for about 400 epochs following the official code with same training data.

Table 1: Quantitative comparisons on Face Synthetics and SynFashion datasets.

Dataset	**Face Synthetics**			**SynFashion**		
	KID↓(std)	LPIPS↓(std)	SSIM↑(std)	KID↓(std)	LPIPS↓(std)	SSIM↑(std)
CUT [10]	80.553 (2.447)	0.365 (0.073)	0.664 (0.079)	59.238 (1.599)	0.170 (0.060)	0.847 (0.067)
SANTA [11]	90.390 (2.929)	0.387 (0.079)	0.618 (0.104)	61.636 (1.628)	0.294 (0.067)	0.741 (0.082)
VCT [13]	74.445 (2.273)	**0.096** (0.027)	0.807 (0.072)	59.489 (1.499)	0.178 (0.058)	0.807 (0.085)
UNSB [12]	76.389 (2.465)	0.229 (0.069)	0.818 (0.070)	59.496 (1.453)	0.130 (0.040)	**0.891** (0.054)
Ours	**73.871** (1.973)	0.121 (0.035)	**0.831** (0.068)	**54.720** (1.362)	**0.067** (0.025)	0.881 (0.055)

Table 2: User studies on overall realism, image quality and consistency. The table shows the percentage of votes that existing methods are preferred to ours.

Dataset	**Face Synthetics**			**SynFashion**		
	Overall Realism	Image Quality	Consistency	Overall Realism	Image Quality	Consistency
CUT	0.529%	0.529%	13.175%	8.994%	6.878%	16.931%
SANTA	0.922%	1.383%	12.304%	3.333%	5.238%	11.571%
VCT	5.952%	14.286%	20.714%	2.041%	6.122%	18.367%
UNSB	4.511%	6.767%	21.278%	9.821%	9.821%	26.607%

4.3 Results

Qualitative Results Fig. 3 and Fig. 4 show the visual comparison between our method, CUT [10], SANTA [11], UNSB [12] and VCT [13] on the SynFashion and Face Synthetics datasets. As can be seen from the figures, both the CUT and SANTA methods exhibit some degree of image degradation and fail to effectively learn the concept of image realism from data across rendered and real domains, thus enable to generate realistic images. The diffusion based style transfer method VCT maintains image quality but fails to extract realistic image features from the guidance image, also resulting in the loss of image details. Compared to previous methods, the UNSB method achieves better consistency in terms of content, but like CUT and SANTA, it performs poorly in maintaining color fidelity and the realism effect is not good. The proposed method effectively enhances the overall realism of the image, particularly in capturing the facial and hand features of models, as well as the texture and wrinkle details of the garment.

Quantitative Results The absence of ground truth for rendered-to-real translation and domain gap between the source rendered and target real domains make quantitative evaluation challenging.

Following the previous work [8], we use KID to evaluate the realism of the generated images and the average SSIM and LPIPS to assess content similarity. For each dataset, we use the 7500 testing result images from each method and calculate the KID against the realistic images and the SSIM/LPIPS against the rendered images. As shown in Tab. 1, our method shows significant improvements in terms of realism as well as overall texture and content consistency. The standard deviations here show the variance over test inputs for a fixed model to demonstrate the stability and generalization ability.

User Studies We adopt user studies to provide more quantitative insight into perceived realism, image quality, and consistency to input rendered images. We follow StyleDiffusion [74] in style-transfer and compare our method to previous works in pairs. Specifically, we randomly sample 100 image pairs from each dataset for user evaluation. Each pair contains one image generated by our method and a corresponding image generated by another comparison method, presented side by side in random order. Users are asked to assess the images based on three criteria: 1) which result appears more realistic, 2) which result demonstrates overall better image quality, and 3) which result shows better consistency with the reference image.

We collected approximately 2,000 votes per question from 20 users and present the percentage of votes where existing methods were preferred over ours in the Tab. 2. Lower percentages indicate that our method was favored over the competitors. Our approach garnered a strong preference in terms of overall realism and image quality, while also showing a clear advantage in maintaining consistency with the reference images.

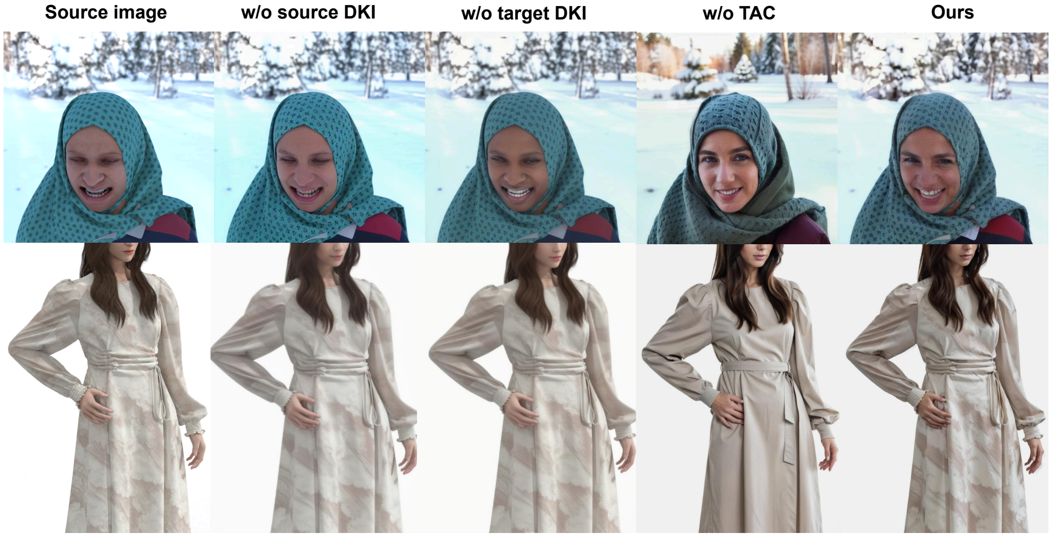

Figure 5: Visual examples of ablation study in a drop-one-out manner. (DKI: Domain Knowledge Injection. TAC: Texture-preserving Attention Control.)

Table 3: Ablation study in a drop-on-out manner.

Dataset	**Face Synthetics**			**SynFashion**		
	KID↓(std)	LPIPS↓(std)	SSIM↑(std)	KID↓(std)	LPIPS↓(std)	SSIM↑(std)
w/o source DKI	77.376 (2.063)	0.107 (0.029)	0.857 (0.059)	58.520 (1.902)	0.059 (0.019)	0.903 (0.065)
w/o target DKI	78.927 (2.134)	0.114 (0.031)	0.845 (0.063)	60.186 (1.623)	0.064 (0.022)	0.897 (0.056)
w/o TAC	69.349 (1.485)	0.253 (0.070)	0.720 (0.085)	51.392 (1.083)	0.183 (0.047)	0.794 (0.074)
Ours	73.831 (1.973)	0.121 (0.035)	0.831 (0.068)	54.720 (1.362)	0.067 (0.025)	0.881 (0.055)

4.4 Ablation Study and Further Analysis

We conduct ablation study on two datasets in a drop-one-out manner and evaluate the performance of each module in the proposed method during inference and analyze the impact on the final results. As shown in Fig. 5, without source DKI (embedding), the fine-tuned base model tends to recover the input rendering image with DDIM inversion. Without target DKI (fine-tuning), the rendering effect slightly decreases but the output is still not real enough due to lack of concentrated knowledge on real human and clothing. Without TAC, the semantic structure such as face identity and clothing design can significantly deviate from the input. The quantitative results are in Tab. 3.

Fig. 6 shows the trade-off between image realism and texture preservation. With a high denoising strength, the generated images resemble realistic images more closely but retain fewer details from the original rendered image. Increasing the TAC ratio helps to better preserve the texture details and facial features. Unlike other content preservation techniques such as inpainting, which can lead to potential visual incoherence, our TAC seems to blend the attention features smoothly into the generation process and cause no obvious coherence issues.

5 Conclusion

In this paper, we introduce a novel diffusion-based framework for rendered-to-real fashion image translation and create a high-quality rendered fashion image dataset (SynFashion), which includes 10k images with multiple classes. With Domain Knowledge Injection (DKI) and Texture-preserving Attention Control (TAC), our method can successfully translate the rendered fashion image into its realistic counterpart with significant realism improvement and texture details preservation. Extensive experimental results demonstrate the superiority and effectiveness of our method.

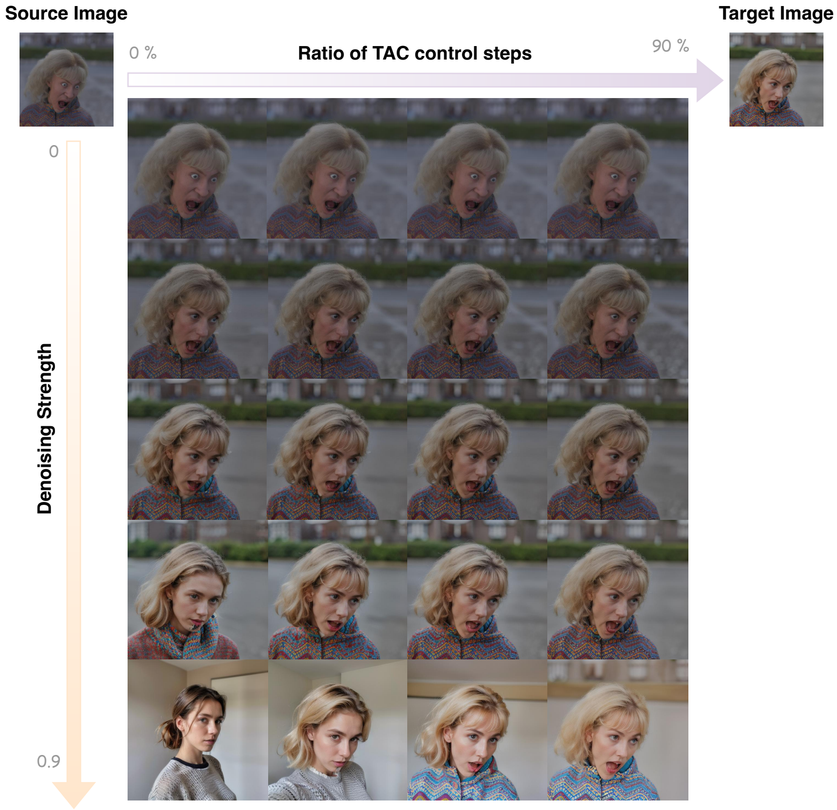

Figure 6: A visual example of tuning TAC ratio and denoising strength.

Table 4: Comparison of memory required and testing time across different methods.

	CUT	SANTA	VCT	UNSB	Ours
Memory Required (GB)	3.3	4.5	22	7.4	7.7
Testing Time (s)	0.38	0.33	62.47	0.53	7.98

Limitations and social impacts While our method achieves superior results on this challenging task, there are still several problems to be further explored. In this work, we simply use DDIM inversion to extract texture-related attention features. However, the inversion process slows down the generation, requiring approximately one minute to translate an image with a resolution of 768×1024. This could potentially be accelerated by recent inversion-free methods. We test the inference time and resource consumption for a 512x512 image on an RTX 3090, as shown in Tab. 4. Note that comparing to VCT, which is also based on diffusion, our method takes much less memory and time during testing as we do not need to perform additional optimization for each testing image. Our method cannot handle real-time applications for now, but has potential for improvement with future integration with SD Turbo or SD Lightning. Additionally, for different images, finding the optimal balance between the TAC ratio and denoising strength may require more empirical refinements to achieve the best result. Due to limitations on computational resources, experiments were not conducted on more advanced models such as SDXL [75]. Given that our method is based on SD1.5 and for human-related content generation, potential negative societal impacts of exploiting this method could be violation of portrait rights, racial bias, or inappropriate content in generation when the denoising strength is high. Relative solutions can include but are not limited to using authorized, diverse and balanced training data and training detection models to prevent inappropriate content generation.

6 Acknowledgments

This work is supported in part by the National Key Research and Development Program of China (No: 2021YFF0501503) and by the Talent Program of Zhejiang Province (No: 2021R51004).

References

[1] Erroll Wood, Tadas Baltrušaitis, Charlie Hewitt, Sebastian Dziadzio, Thomas J Cashman, and Jamie Shotton. Fake it till you make it: face analysis in the wild using synthetic data alone. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3681–3691, 2021.

[2] Astitva Srivastava, Pranav Manu, Amit Raj, Varun Jampani, and Avinash Sharma. Wordrobe: Text-guided generation of textured 3d garments. *arXiv preprint arXiv:2403.17541*, 2024.

[3] Botao Wu, Zhendong Wang, and Huamin Wang. A gpu-based multilevel additive schwarz preconditioner for cloth and deformable body simulation. *ACM Transactions on Graphics (TOG)*, 41(4):1–14, 2022.

[4] Zhendong Wang, Yin Yang, and Huamin Wang. Stable discrete bending by analytic eigensystem and adaptive orthotropic geometric stiffness. *ACM Transactions on Graphics (TOG)*, 42(6):1–16, 2023.

[5] Yifei Li, Hsiao-yu Chen, Egor Larionov, Nikolaos Sarafianos, Wojciech Matusik, and Tuur Stuyck. Diffavatar: Simulation-ready garment optimization with differentiable simulation. *arXiv preprint arXiv:2311.12194*, 2023.

[6] Micah K Johnson, Kevin Dale, Shai Avidan, Hanspeter Pfister, William T Freeman, and Wojciech Matusik. Cg2real: Improving the realism of computer generated images using a large collection of photographs. *IEEE Transactions on Visualization and Computer Graphics*, 17(9):1273–1285, 2010.

[7] Sai Bi, Kalyan Sunkavalli, Federico Perazzi, Eli Shechtman, Vladimir G Kim, and Ravi Ramamoorthi. Deep cg2real: Synthetic-to-real translation via image disentanglement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2730–2739, 2019.

[8] Stephan R Richter, Hassan Abu AlHaija, and Vladlen Koltun. Enhancing photorealism enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):1700–1715, 2022.

[9] Luyuan Wang, Yiqian Wu, Yongliang Yang, Chen Liu, and Xiaogang Jin. Enhancing the authenticity of rendered portraits with identity-consistent transfer learning. *arXiv preprint arXiv:2310.04194*, 2023.

[10] Taesung Park, Alexei A Efros, Richard Zhang, and Jun-Yan Zhu. Contrastive learning for unpaired image-to-image translation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*, pages 319–345. Springer, 2020.

[11] Shaoan Xie, Yanwu Xu, Mingming Gong, and Kun Zhang. Unpaired image-to-image translation with shortest path regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10177–10187, 2023.

[12] Beomsu Kim, Gihyun Kwon, Kwanyoung Kim, and Jong Chul Ye. Unpaired image-to-image translation via neural schrödinger bridge. *arXiv preprint arXiv:2305.15086*, 2023.

[13] Bin Cheng, Zuhao Liu, Yunbo Peng, and Yue Lin. General image-to-image translation with one-shot image guidance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22736–22746, 2023.

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

[15] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022.

[16] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *The Eleventh International Conference on Learning Representations*, 2022.

[17] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[18] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[19] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1921–1930, 2023.

[20] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

[21] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22560–22570, 2023.

[22] Bingyan Liu, Chengyu Wang, Tingfeng Cao, Kui Jia, and Jun Huang. Towards understanding cross and self-attention in stable diffusion for text-guided image editing. *arXiv preprint arXiv:2403.03431*, 2024.

[23] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

[24] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020.

[25] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[26] Tao Chen, Ming-Ming Cheng, Ping Tan, Ariel Shamir, and Shi-Min Hu. Sketch2photo: Internet image montage. *ACM transactions on graphics (TOG)*, 28(5):1–10, 2009.

[27] Aaron Hertzmann, Charles E Jacobs, Nuria Oliver, Brian Curless, and David H Salesin. Image analogies. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 557–570. 2023.

[28] Lara Raad and Bruno Galerne. Efros and freeman image quilting algorithm for texture synthesis. *Image Processing On Line*, 7:1–22, 2017.

[29] Yichang Shih, Sylvain Paris, Frédo Durand, and William T Freeman. Data-driven hallucination of different times of day from a single outdoor photo. *ACM Transactions on Graphics (TOG)*, 32(6):1–11, 2013.

[30] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[31] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2287–2296, 2021.

[32] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[33] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[34] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[35] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. *Advances in neural information processing systems*, 30, 2017.

[36] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. In *International conference on machine learning*, pages 1857–1865. PMLR, 2017.

[37] Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *Proceedings of the IEEE international conference on computer vision*, pages 2849–2857, 2017.

[38] Sihan Xu, Ziqiao Ma, Yidong Huang, Honglak Lee, and Joyce Chai. Cyclenet: Rethinking cycle consistency in text-guided diffusion for image manipulation. *Advances in Neural Information Processing Systems*, 36, 2024.

[39] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, Kun Zhang, and Dacheng Tao. Geometry-consistent generative adversarial networks for one-sided unsupervised domain mapping. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2427–2436, 2019.

[40] Sagie Benaim and Lior Wolf. One-sided unsupervised domain mapping. *Advances in neural information processing systems*, 30, 2017.

[41] Chanyong Jung, Gihyun Kwon, and Jong Chul Ye. Exploring patch-wise semantic relation for contrastive learning in image-to-image translation tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18260–18269, 2022.

[42] Weilun Wang, Wengang Zhou, Jianmin Bao, Dong Chen, and Houqiang Li. Instance-wise hard negative example generation for contrastive learning in unpaired image-to-image translation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 14020–14029, 2021.

[43] Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. The spatially-correlative loss for various image translation tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16407–16417, 2021.

[44] Bin Wang, Wenping Wang, Huaiping Yang, and Jiaguang Sun. Efficient example-based painting and synthesis of 2d directional texture. *IEEE Transactions on Visualization and Computer Graphics*, 10(3):266–277, 2004.

[45] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016.

[46] Leon A Gatys, Alexander S Ecker, Matthias Bethge, Aaron Hertzmann, and Eli Shechtman. Controlling perceptual factors in neural style transfer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3985–3993, 2017.

[47] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Universal style transfer via feature transforms. *Advances in neural information processing systems*, 30, 2017.

[48] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pages 1501–1510, 2017.

[49] Yingying Deng, Fan Tang, Weiming Dong, Wen Sun, Feiyue Huang, and Changsheng Xu. Arbitrary style transfer via multi-adaptation network. In *Proceedings of the 28th ACM international conference on multimedia*, pages 2719–2727, 2020.

[50] Yuxin Zhang, Nisha Huang, Fan Tang, Haibin Huang, Chongyang Ma, Weiming Dong, and Changsheng Xu. Inversion-based style transfer with diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10146–10156, 2023.

[51] Manuel Lagunas, Xin Sun, Jimei Yang, Ruben Villegas, Jianming Zhang, Zhixin Shu, Belen Masia, and Diego Gutierrez. Single-image full-body human relighting. In *Eurographics Symposium on Rendering (EGSR)*. The Eurographics Association, 2021.

[52] Yu-Ying Yeh, Koki Nagano, Sameh Khamis, Jan Kautz, Ming-Yu Liu, and Ting-Chun Wang. Learning to relight portrait images via a virtual light stage and synthetic-to-real adaptation. *ACM Transactions on Graphics (TOG)*, 41(6):1–21, 2022.

[53] Rongyuan Wu, Tao Yang, Lingchen Sun, Zhengqiang Zhang, Shuai Li, and Lei Zhang. Seesr: Towards semantics-aware real-world image super-resolution. *arXiv preprint arXiv:2311.16518*, 2023.

[54] Fanghua Yu, Jinjin Gu, Zheyuan Li, Jinfan Hu, Xiangtao Kong, Xintao Wang, Jingwen He, Yu Qiao, and Chao Dong. Scaling up to excellence: Practicing model scaling for photo-realistic image restoration in the wild. *arXiv preprint arXiv:2401.13627*, 2024.

[55] Jianyi Wang, Zongsheng Yue, Shangchen Zhou, Kelvin CK Chan, and Chen Change Loy. Exploiting diffusion prior for real-world image super-resolution. *arXiv preprint arXiv:2305.07015*, 2023.

[56] Haoyu Chen, Wenbo Li, Jinjin Gu, Jingjing Ren, Haoze Sun, Xueyi Zou, Zhensong Zhang, Youliang Yan, and Lei Zhu. Low-res leads the way: Improving generalization for super-resolution by self-supervised learning. *arXiv preprint arXiv:2403.02601*, 2024.

[57] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4713–4726, 2022.

[58] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1833–1844, 2021.

[59] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1905–1914, 2021.

[60] Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4791–4800, 2021.

[61] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

[62] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

[63] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.

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

[65] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[66] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510, 2023.

[67] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6007–6017, 2023.

[68] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023.

[69] Kim Youwang, Tae-Hyun Oh, and Gerard Pons-Moll. Paint-it: Text-to-texture synthesis via deep convolutional texture map optimization and physically-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4347–4356, 2024.

[70] Dan Casas and Marc Comino-Trinidad. SMPLitex: A Generative Model and Dataset for 3D Human Texture Estimation from Single Image. In *British Machine Vision Conference (BMVC)*, 2023.

[71] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[72] 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 *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[73] David Shi, Maggie, Menglin Jia, Mihail Sirotenko, and Will Cukierski. imaterialist (fashion) 2019 at fgvc6, 2019.

[74] Zhizhong Wang, Lei Zhao, and Wei Xing. Stylediffusion: Controllable disentangled style transfer via diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7677–7689, 2023.

[75] 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.

A Appendix / supplemental material

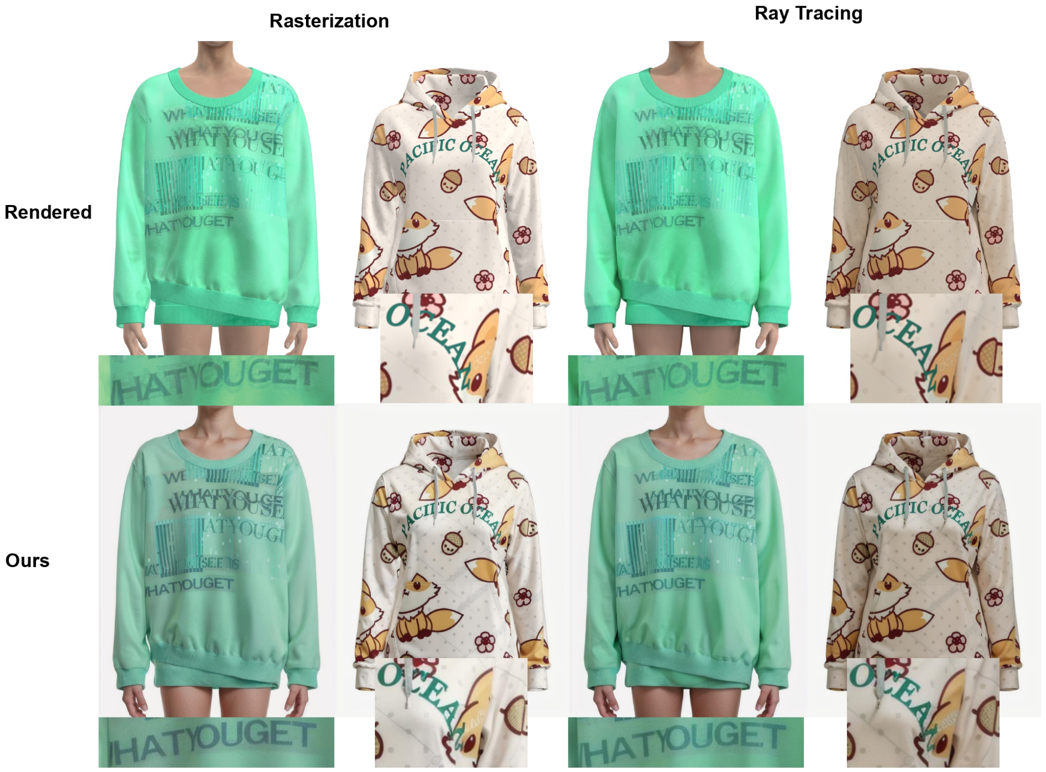

Figure 7: Results on textual textures and different rendering inputs.

A.1 More implementation details

More details of our RIG algorithm is shown in Algorithm. 1.

Algorithm 1 Realistic Image Generation

```
Inputs:
$x_{cg} \leftarrow$ source rendered image
$v_{nd} \leftarrow$ negative domain embedding
$\gamma, F \leftarrow$ step threshold, feature size threshold
Algorithm:
$z_0 = \mathcal{E}(x_{cg})$
$\hat{z}_T \leftarrow$ DDIM-inv$(z_0)$
$z_T \leftarrow \hat{z}_T$    // starting from same seed
for $t = T$ to 1 do
    $z_{t-1}, Q^t_{cg}, K^t_{cg} \leftarrow$ DDIM-samp$(z_t)$
    if $t < \gamma$ & $f > F$ then
        $\hat{z}_{t-1} \leftarrow \tilde{\epsilon}^*_\theta(\hat{z}_t, t, v_{nd})\{Q^t_r \leftarrow Q^t_{cg}; K^t_r \leftarrow K^t_{cg}\}$
    else
        $\hat{z}_{t-1} \leftarrow \tilde{\epsilon}^*_\theta(\hat{z}_t, t, v_{nd})$
    end if
end for
Output: $x_r \leftarrow \mathcal{D}(\hat{z}_0)$
```

Table 5: Number of images in different categories of SynFashion.

Category	Pants	T-shirt	Lingerie & Swimwear	Half Skirt	Hoodie	Coat	Jacket
Numbers	864	416	440	392	448	604	812
Category	Set	Home-wear	Hat	Hanfu	Jeans	Shorts	Down Jacket
Numbers	420	336	308	472	180	420	508
Category	Vest & Camisole	Shirt	Suit	Dress	Sweater	Trench Coat	
Numbers	388	476	672	416	1056	416	

A.2 Results on textual textures and different rendering inputs

As for rendering baselines, we build the 3D projects with Style3D Studio and use its integrated rendering tool based on rasterization. Using UE5 could potentially improve the rendering quality but will not diminish the effectiveness of our method. To verify this, we use more advanced rendering techniques via ray tracing (based on V-ray) to obtain rendered images, and our method consistently demonstrates its advantages in realism. Two visual examples are shown in Fig. 7.

A.3 More results of realistic image translation

To further verify the performance of the proposed method in realistic translation tasks, additional experiments were conducted using the collected SynFashion dataset and the Face Synthetics dataset. The results are illustrated in Figure .8 for Face Synthetics and Figure .9 for SynFashion.

A.4 More details of collected SynFashion dataset

Figure .10, Figure .11, Figure .12, and Figure .13 provide detailed visualizations of the SynFashion dataset. The first column in each figure presents the front view of a designed 3D garment object. Various texture patterns are assigned to each garment object, and the subsequent columns show the images with four different views. The number of images in each category is shown in Table. 5.

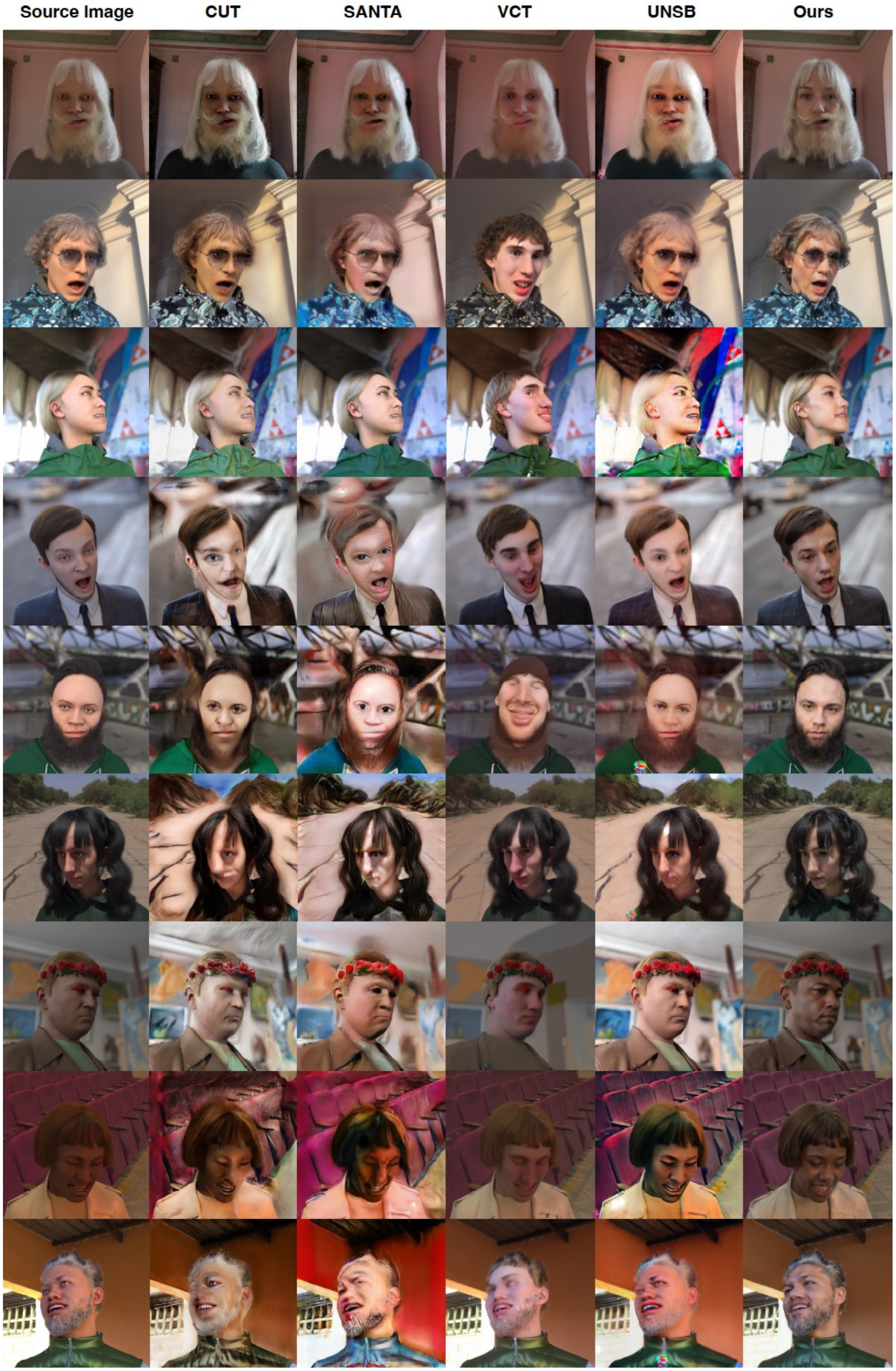

Figure 8: More comparison results on Face Synthetics dataset.

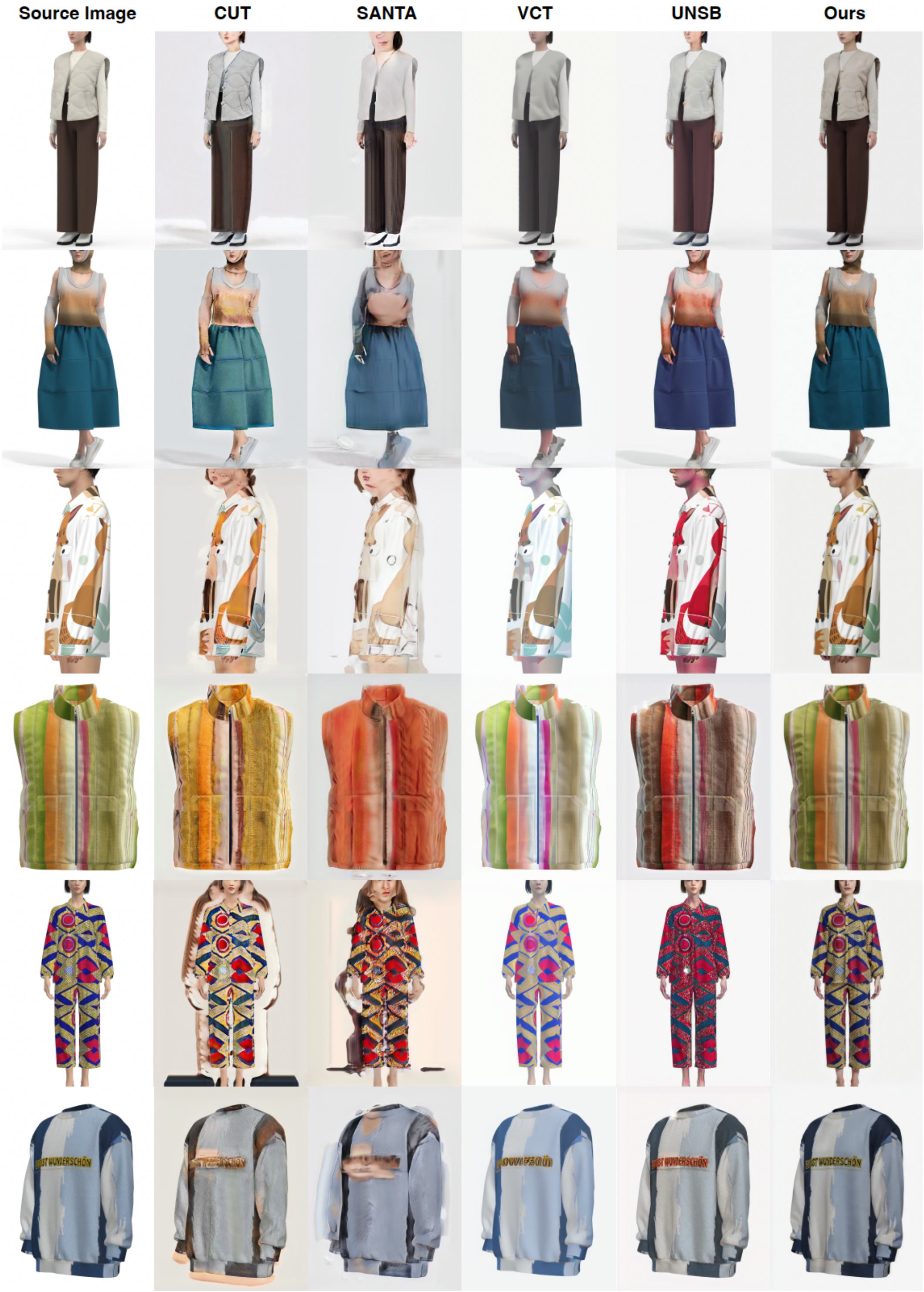

Figure 9: More comparison results on SynFashion dataset.

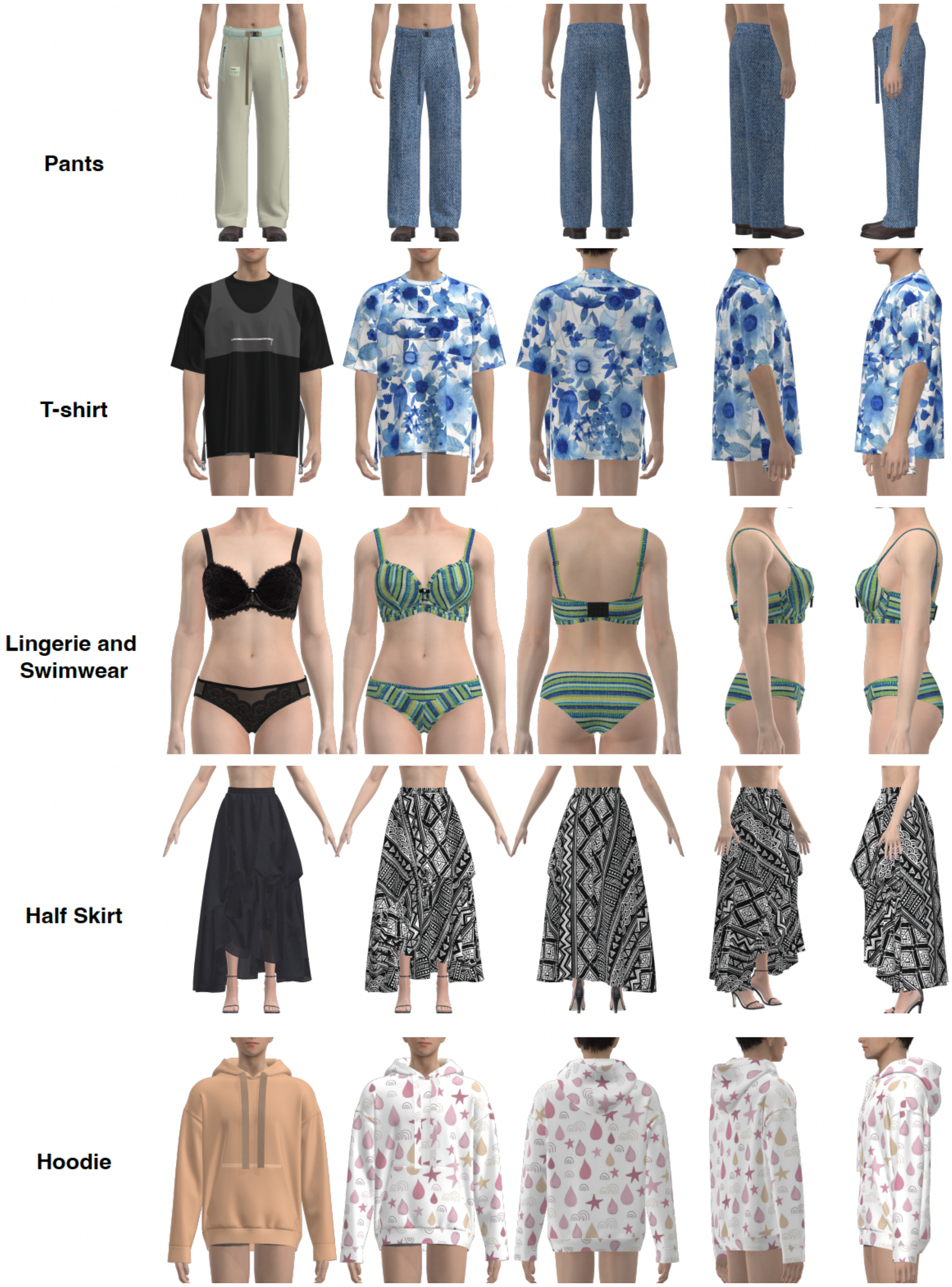

Figure 10: Examples of collected SynFashion dataset (Part 1).

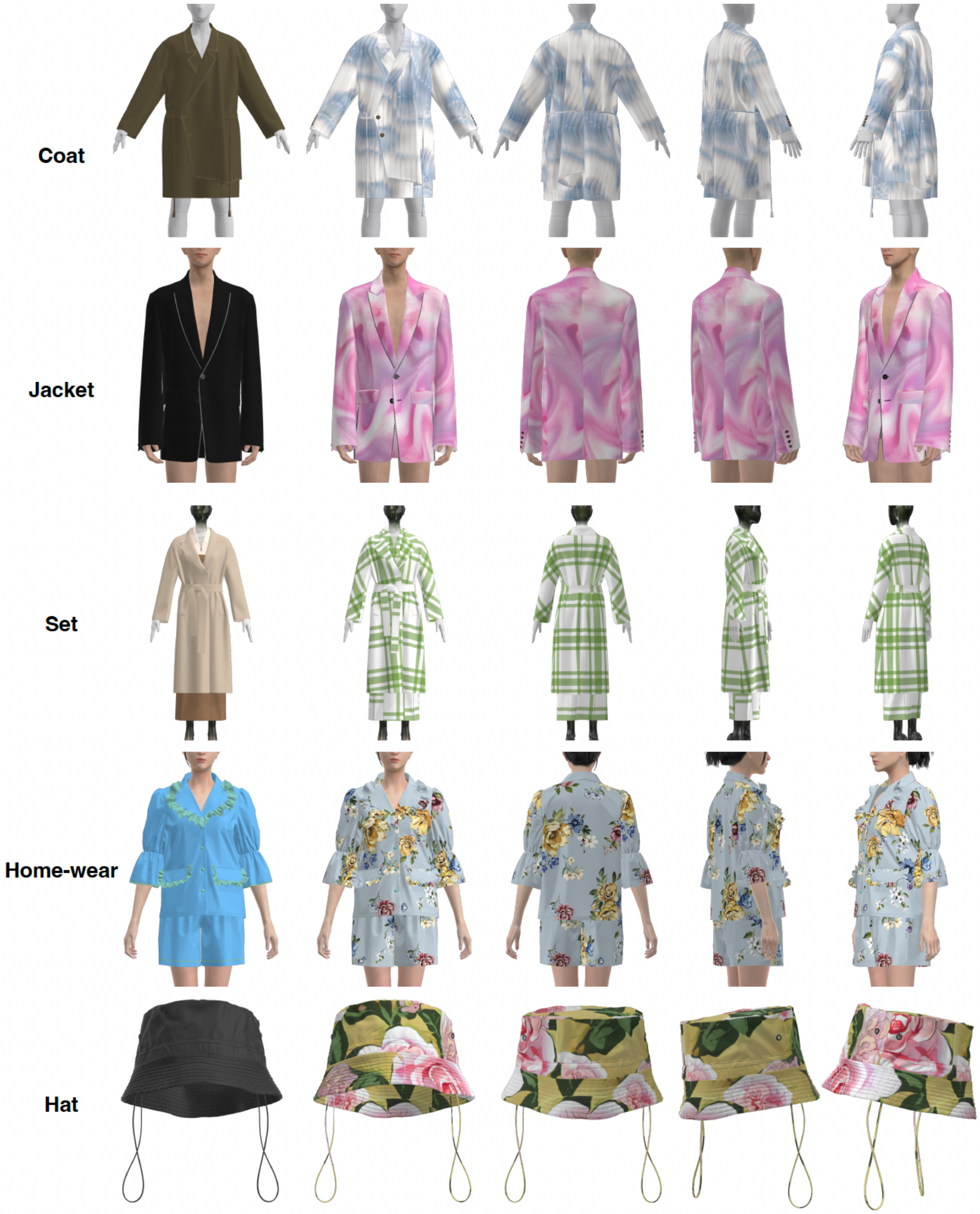

Figure 11: Examples of collected SynFashion dataset (Part 2).

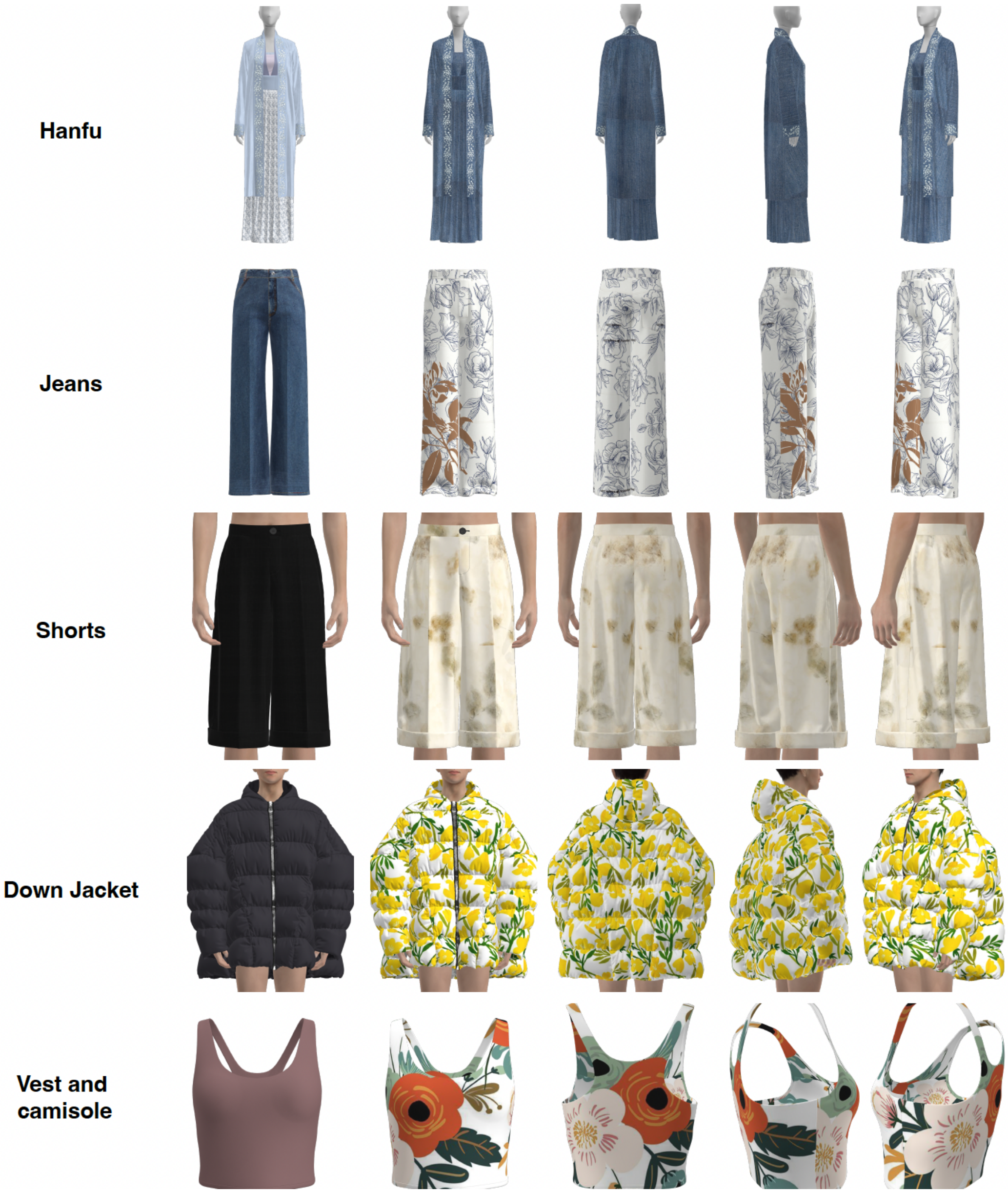

Figure 12: Examples of collected SynFashion dataset (Part 3).

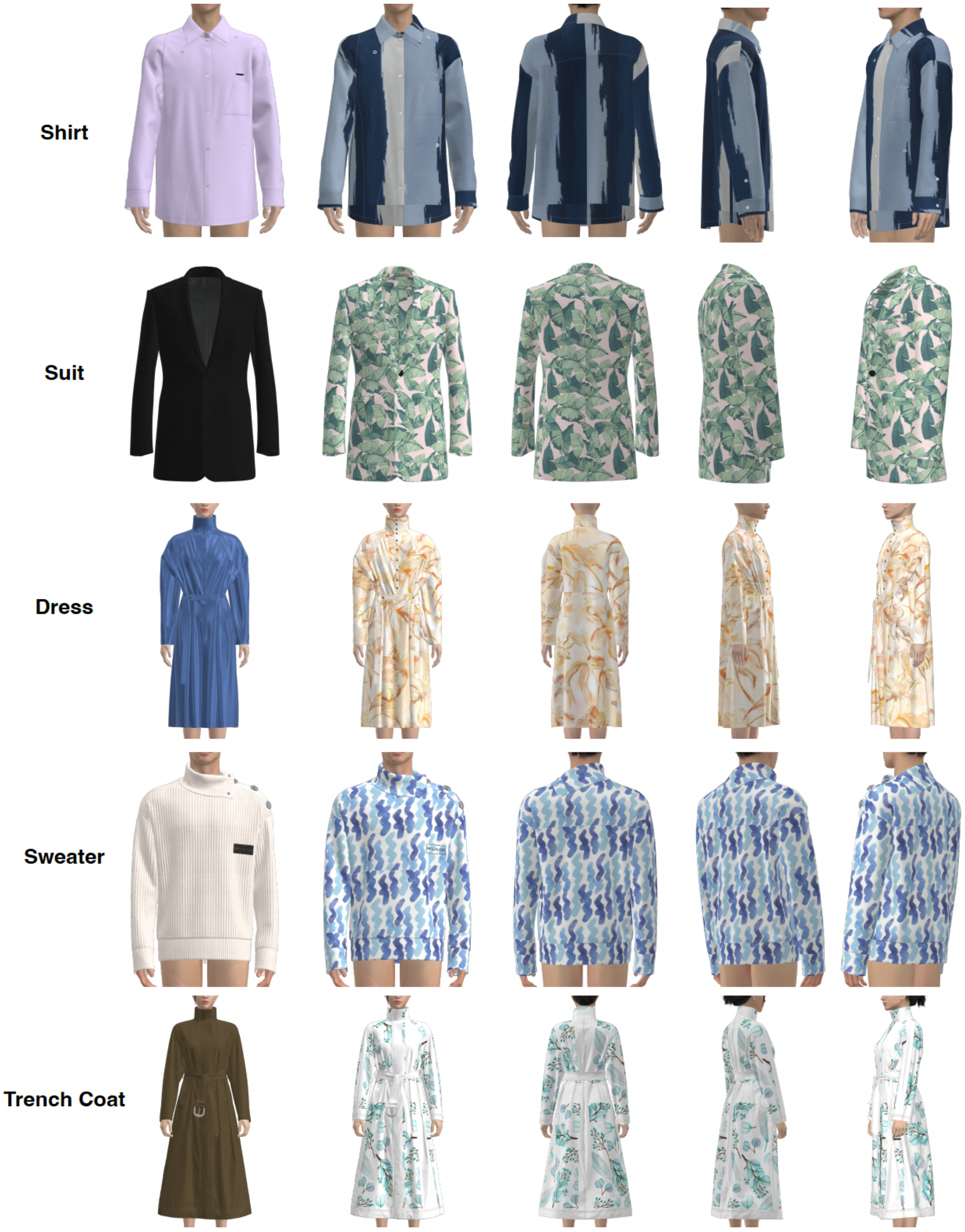

Figure 13: Examples of collected SynFashion dataset (Part 4).

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: The main claims made in the abstract and introduction accurately reflect the papaer's contributions and scope. Details can be found at the end of 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: The limitations of the work are listed in Conclusion.

   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: [NA]

Justification: The paper is more experimental and does not include theoretical results.

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: The paper fully discloses all the information needed to reproduce the main experimental results in Section 4.1 (Datasets) and Section 4.2 (Implementation Details).

   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: Data and code are released in the supplemental materials to preserve anonymity and will be made public available after review process.

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: The training and testing details as shown in Section 4.2 (Implementation Details)

   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]

   Justification: We follow evaluation metrics from previous works and use public evaluation code to compute quantitative results in our paper. Since quantifying realism is still an open question and the metrics are not strictly aligned with the objective, which makes further statistical analysis a bit less informative than in other well-defined problems, we only report the standard deviations to show the variance over test inputs for a fixed model and demonstrate the overall stability and generalization ability.

   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: The detailed information on the computer resources needed are listed in Section 4.2 (Implementation Details).

   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: This research conforms, 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: The broader social impacts are listed in Section 5 (Conclusion)

    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: The only possible misuse of our work would be using a high denoising strength to generate potentially inappropriate images, which is warned in conclusion and will be mentioned when we release the model.

    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: The existing assets are properly credited and respected.

    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: [Yes]

    Justification: The new dataset introduced in the paper is well documented in an anatomized link with the license of CCBY-NC-ND4.0.

    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: The paper does not involve crowdsourcing nor reasearch 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: The paper does 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.