

OmniDiT: Extending Diffusion Transformer to Omni-VTON Framework

Weixuan Zeng^{1†}, Pengcheng Wei^{2†}, Huaiqing Wang³, Boheng Zhang³, Jia Sun³, Dewen Fan³, Lin HE³, Long Chen³, Qianqian Gan³, Fan Yang³, and Tingting Gao³

¹ The Chinese University of Hong Kong, ShenZhen

² Beihang University

³ KuaiShou

weixuanzeng@link.cuhk.edu.cn, 22373151@buaa.edu.cn

{weipengcheng03,wanghuaiqing,yangxiao16,sunjia05,fandewen,
helin05,chenlong11,ganqianqian,yangfan,lisize,lihan08}@kuaishou.com



Fig. 1: Generated images by our OmniDiT trained on our dataset Omni-TryOn. Our unified model supports above three tasks and keeps a strong consistency. Please zoom in to see details preservation.

Abstract. Despite the rapid advancement of Virtual Try-On (VTON) and Try-Off (VTOFF) technologies, existing VTON methods face challenges with fine-grained detail preservation, generalization to complex scenes, complicated pipeline, and efficient inference. To tackle these problems, we propose OmniDiT, an omni Virtual Try-On framework based on the Diffusion Transformer, which combines try-on and try-off tasks into

[†]Work done during an internship at KuaiShou Technology.

one unified model. Specifically, we first establish a self-evolving data curation pipeline to continuously produce data, and construct a large VTON dataset Omni-TryOn, which contains over 380k diverse and high-quality garment-model-tryon image pairs and detailed text prompts. Then, we employ the token concatenation and design an adaptive position encoding to effectively incorporate multiple reference conditions. To relieve the bottleneck of long sequence computation, we are the first to introduce Shifted Window Attention into the diffusion model, thus achieving a linear complexity. To remedy the performance degradation caused by local window attention, we utilize multiple timestep prediction and an alignment loss to improve generation fidelity. Experiments reveal that, under various complex scenes, our method achieves the best performance in both the model-free VTON and VTOFF tasks and a performance comparable to current SOTA methods in the model-based VTON task.

Keywords: Virtual Try-On · Virtual Try-Off · Diffusion Model

1 Introduction

With the breakthroughs of Diffusion Models (DMs), Virtual Try-On (VTON) has attracted considerable focus due to its promising market prospect in e-commerce. VTON which aims to superimpose given garments onto specific model images [2, 8, 17, 28, 61, 69], provides an immersive and personalized shopping experience [59]. Early approaches relied on generative adversarial network (GAN) [15], which typically added a warping module to achieve the semantic correspondence between the garments and the human body, and a generator module to fit the warped garments onto the body [9, 13, 14, 20, 21, 41, 69]. However, this two-stage process often results in unnatural fits and cannot generalize to complex human poses due to the limited warping process [10, 71].

Recently, based on latent diffusion models (LDM) [23, 49], many works [7, 10–12, 28, 36, 42, 50, 54, 61, 71, 85] utilized the rich generative prior of pretrained text-to-image (T2I) models and achieved more natural tryon results. U-Net-based LDMs have effectively improved the realism of outfitted images and could generalize to more complicated scenes [35, 72, 80].

LDM methods fall into two paradigms: mask-based and mask-free approaches. For mask-based ones, a binary human-agnostic mask is extracted and diffusion models are applied to inpaint the garment into the masked area [8, 10, 42, 71]. This method greatly depends on the quality of masking, because it will easily generate patchy clothing and visible artifacts if the mask-extractor cannot perfectly segment the target area [1, 10, 27]. In addition, the mask area easily leads to information leakage, in which the mask shape can tell the model how and where to inpaint the given garment, reducing the difficulty of learning and weakening the generalization ability of the model. Taking these drawbacks into account, some works removed the masking step and applied an end-to-end pipeline, in which users input the garment and the human model, then the model outputs the tryon result [6, 45, 80]. Mask-free diffusion models have emerged as the dominant paradigm for high-fidelity virtual try-on [59].

However, the mask-free method also suffers from severe challenges. First, the end-to-end pipeline requires a fully consistent and matching image triple pair, which greatly complicates the construction of training datasets. The triple images demand that the target garments in the clothing image and the tryon image are exactly the same, and that the model image and the tryon image are completely identical in terms of both the model and the background except their target garments. Actually, popular datasets (*e.g.* VITON [20] and DressCode [43]) still miss the images of matching models. Second, the model must infer all aspects of the garments’ presentation, which increases the difficulty of precise and local refinement along the boundaries of clothing or background coherence [59].

Recently, some works have begun exploring the Virtual Try-Off (VTOFF) application: extracting standardized garment images from clothed individuals, which apparently belongs to the inverse task of VTON [18, 57, 58, 66]. From another perspective, VTON systems demand massive high-quality garment lay-flat images, which are laborious to collect from the Internet. So, as a preliminary process, setting up a VTOFF pipeline to produce high-fidelity garment images can benefit VTON systems’ sustainable operation. In addition, an ideal VTON model can not only perform model-based try-on (with two reference images), but command the capability of model-free try-on (with only a single garment image as input) [7, 36, 50, 61].

Currently, most works focus on a single VTON or VTOFF task, which complicates the whole workflow and fails to satisfy users’ diverse demands, only Any2anyTryon [18] attempted to combine the three abilities into one unified model. But their method suffers from simple and small training datasets, and cannot generalize to complex scenes.

To tackle above limitations, we propose OmniDiT, an omni VTON and VTOFF framework built upon a Diffusion Transformer (DiT) [47] and mask-free paradigm. Specifically, we employ token concatenation to integrate reference signals and design an adaptive position encoding to distinguish different token blocks. Aiming to reduce computational costs of the attention modules [56], we are the first to introduce shifted window attention (SWA) [39] into the diffusion model. In addition, to compensate for the performance degradation induced by SWA, we revise the flow matching objective by predicting multiple timesteps during training, promoting the current timestep’s prediction to focus on global timesteps’ prediction and yield more stable trajectories. Lastly, to enhance the clothing fidelity, we add an alignment loss anchored in the local clothing region. To further boost robustness and versatility, we curate a large scale try-on dataset—Omni-TryOn, which contains over 380k high-quality, diverse poses and scenes’ garment, human model and tryon image triple pairs. We develop a self-evolving curation pipeline that combines the mature VTON and VTOFF capabilities to continuously produce data.

In summary, the contributions of our work include the following:

1. We inject multiple condition signals into the DiT by concatenating tokens and designing an adaptive position encoding, aiming to combine VTON and VTOFF tasks into one unified framework.

2. We introduce shifted window attention into the diffusion model to reduce computation complexity.
3. Our multiple timesteps’ prediction training strategy can encourage every timestep’s prediction to focus on following prediction, reducing global error. And additional alignment loss can enhance the clothing fidelity.
4. We propose a self-evolving data curation pipeline, which combines the updated model’s abilities to continuously produce high-quality data. Based on the pipeline, we curate a large dataset with over 380k diverse samples.

2 Related Works

Image-based Virtual Try-On. Image-based virtual try-on has been extensively explored over the recent years, emerging as a promising and formidable task. Early studies based on Generative Adversarial Networks (GANs) [15] proceed with a two-stage pipeline. A warping model deforms the given garment into a shape that approximately fits the person’s pose. Then, a GAN-based generator fuses the wrapped clothing into the result image generation [19, 31, 41, 60, 69, 73]. However, these approaches are frequently hampered by visual artifacts from inaccurate warping [12]. Subsequently, the advent of diffusion models revolutionized the field by reframing the task as an end-to-end conditional image generation, removing the error-prone warping step [4, 10, 11, 17, 42, 50, 71, 75, 84, 85]. The dominant strategy in these methods involves injecting garment features into the denoising process via sophisticated conditioning mechanisms such as parallel encoder branches (i.e., ReferenceNets), ControlNet [77] or IP-adapter [74]. With the advent of Diffusion Transformer architecture [30, 47], some works have explored more generalized conditioning schemes, among which token concatenation has achieved the best performance [18, 35, 44, 52, 65, 83]. Despite achieving unprecedented high-fidelity, the model’s vast size has significantly increased inference latency, especially multiple conditions as input, hindering their applications in real-world scenarios that demand rapid feedback and multi-item outfit composition [12]. Besides model-based try-on, another line of applications is model-free try-on, which needs to generate the human model and corresponding tryon results. This task is more difficult, because it requires the try-on system to imagine the human model based on pure text prompts and maintain the garment conditions’ details. Current works mainly employ the condition-injection paradigm to achieve the goal [18, 36, 50, 61].

Virtual Try-Off. While most existing works focus on virtual try-on, few studies have explored virtual try-off [79, 81, 82], the inverse task of virtual try-on, aiming to reconstruct a clean garment from a tryon image. Early works, such as TileGAN [76], used a two-stage pipeline: a U-Net-like encoder-decoder for coarse synthesis followed by a pix2pix-based refinement. Recent works explored text-guided garment generation based on Diffusion models [18, 32, 57, 58]. But rare works have attempted to combine try-on and try-off into a unified framework to learn [18, 32]. This unified setup not only supports multitask learning but also mitigates architectural, task-specific, and category-specific inductive biases by exposing the model to broader structural variation.

3 Method

3.1 Preliminary

Virtual Try-On and Try-Off. We mainly focus on mask-free Virtual Try-On and Try-Off. Given a human model image and a garment image as inputs, the model-based try-on pipeline generates a tryon image without relying on any mask condition, and the model-free try-on operates similarly, only removing the human model image. As the inverse task of Virtual Try-On, Virtual Try-Off generates the garment image based on the tryon image.

Flow Matching. Unlike Diffusion denoising methods [24, 51], Flow matching [37, 38] predicts a smooth, continuous velocity field from noise to data in a direct manner. By conducting the forward process by linearly interpolating between noise and data, it can potentially achieve faster and more efficient image synthesis. At timestep t , latent x_t is defined as: $x_t = (1 - t)x_0 + tx_1$, where x_0 is the clean image, and $x_1 \in N(0, 1)$ is the Gaussian noise. A model is trained to directly regress the target velocity given the noised latent x_t , timestep t , and conditions c (including the text prompt, reference images.). The flow matching loss can be summarized as follows:

$$\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{t \sim \mathcal{U}[0,1]} [\|v_\theta(x_t, t, c) - (x_1 - x_0)\|^2] \quad (1)$$

where v_θ represents the velocity field predicted by a model parameterized by neural network weights θ .

3.2 Omni-TryOn Dataset Construction

An ideal Virtual Try-On dataset should satisfy three diverse and two consistent requirements: (1) diverse garments: including garment’s categories and garment’s styles; (2) diverse human models: including models’ identities and postures; (3) diverse scenes: including garments and models’ scenes; (4) consistent garments: the target garments in garment and tryon image pairs should keep consistent; (5): consistent models: the human model in the model and tryon image pairs should keep consistent. Current public try-on datasets, such as VITON-HD [9] and DressCode [43] only contain garment-tryon pairs, missing the matching human model, and cannot meet the standard of diverse scenes and poses, limiting the development of the mask-free VTON techniques.

Dataset Construction Pipeline. We first collected two millions of garment display images from the Internet, including 3-5 pure garment images and tryon results for each garment. To construct fully matching and consistent triple pairs, we use powerful VLMs, such as Qwen3-VL [3] and InternVL-3.5 [62] to filter qualified images, and the criteria include garment or model image classification, watermark detection, content proportion, garments or models’ count, garment categories, model type and age, and so on. Among filtered images, we categorize them into two groups based on whether they contain fully matching garments and tryon results: one contains garments and the corresponding tryon results,

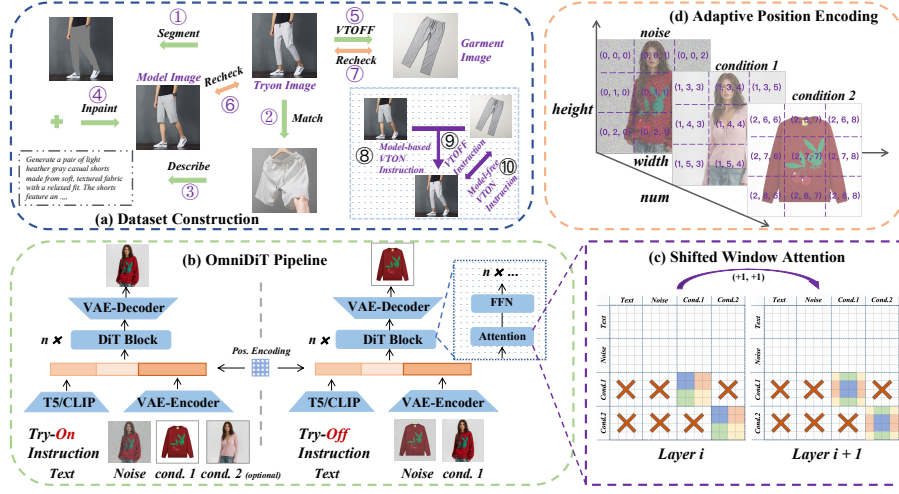


Fig. 2: Overview of Our OmniDiT framework. The four blocks demonstrate our (a) dataset construction, (b) OmniDiT model details, (c) shifted windows attention, and (d) adaptive position encoding.

the other contains only the tryon results without corresponding garments. Then we can employ different workflows to synthesize the expected image pairs:

For garment-tryon pairs: Due to the available garment and tryon results, we only need to produce a human model which has the same model and scene as the tryon image, but different dressed garment to replace the target garment. As shown in Fig. 2a, we use a segmentation model [68] to generate a masked human image. Simultaneously, we sample a matching garment based on the target garment’s category, the age and gender of the model from our pre-built garment database, which includes several types of garments suitable for diverse models with a wide range of ages and genders. Then employ a VLM to generate a detailed text description of the sampled garment, and utilize an inpainting model to fill in the masked image with the text instruction and obtain the corresponding model image. To ensure consistency among the triple image pairs, employ VLMs to recheck the results. Then, we use the Qwen3-VL model to generate detailed text prompts tailed for three tasks: model-based try-on, model-free try-on and try-off tasks. Finally, to guaranty the aesthetics, we utilize an aesthetic model to score the garment and tryon images for each sample.

For only tryon images: The overall pipeline is similar to that of garment-tryon pairs, only need to attain the corresponding target garment images additionally. Current powerful generation model such as Gemini-2.5-image [16] can put off the clothes to perfectly generate the target garment image.

Actually, after collecting the first batch of training data, we can update our original OmniDiT model to acquire the try-on and try-off abilities. Then OmniDiT can generate qualified model images based on tryon results and sampled garments, thus removing the mask and inpainting steps. Also replace the Nano

Banana API service to simplify the try-off pipeline. The regenerated triplets are re-filtered via a VLM, yielding an expanded corpus with both high-quality and wider coverage. Overall, our dataset construction pipeline paired with updated OmniDiT model can achieve a self-evolving consequence, continuously producing high-quality data. After multiple iterations, we collected 380k high-quality, diverse samples with 1895 test samples as shown in Tab. 5. More details can be found in Appendix A.

3.3 Model Architecture

In this part, we present our method for modifying DiT to adapt virtual try-on and try-off tasks. OmniDiT builds on Flux.1-Kontext-dev [30], which injects the reference image and text prompt along with the noisy signal into its MM-DiT blocks by concatenating all the image and text tokens into a unified sequence. The token concatenation facilitates cross-modal interactions to iteratively guide and refine the synthesis process, thus generating images that faithfully reflect the prompts and reference conditions [59]. Several previous works have demonstrated that token concatenation yields the best performance [35, 44, 52, 65]. Although the original Flux model only supports one reference image as input, it can accept more reference images by concatenating extra image tokens into the unified sequence, enabling seamless incorporation of various control signals and facilitating high-fidelity, controllable image generation.

Multi-Condition In-Context Generation. As shown in Fig. 2b, the text prompt, the noise and reference images (garment and model images for the model-based try-on, garment images for the model-free try-on, tryon images for the try-off task) are encoded into the same latent space using text and image encoders. Assuming the text tokens T , noisy tokens X and reference condition tokens $[C_1, \dots, C_n]$, $n = 2$ or $n = 1$ for our above three tasks, all tokens are concatenated along the sequence dimension $S = [T; X; C_1; \dots, C_n]$ and fed into the model.

Shifted Window Attention. Due to the attention’s quadratic complexity with respect to the sequence length, adding more high-resolution reference images into the sequence will cause the global self-attention computation unaffordable. To tackle this problem, we are the first to introduce the shifted window attention (SWA) [39] into the diffusion model. Considering the characteristic of image generation, we apply SWA only to the reference images. As illustrated in Fig. 2c, a full reference image is partitioned into several non-overlapping local windows, and the attention is computed within the local windows. To introduce cross-window connections, a shifted window partitioning approach is applied in consecutive attention blocks. The first layer i uses a regular window partitioning strategy in which the 4×4 feature map is evenly partitioned into 2×2 windows of size 2×2 ($M = 2$). Then, the next layer $i + 1$ is shifted from that of the preceding layer, by rolling the windows by $(\lfloor \frac{M}{2} \rfloor, \lfloor \frac{M}{2} \rfloor)$ pixels towards the bottom right direction. The shifted window attention achieves linear complexity by limiting self-attention computation to non-overlapping local windows, and obtains comparable performance with global attention by allowing cross-window

connection. Inspired by [55, 83], we also introduce the Causal Conditional Attention design, in which the attention from condition tokens to denoising tokens (noise and text) is blocked, thus further reducing redundant computation and improving efficiency. In the case of two $1024 \times$ reference images as input and generate a $1024 \times$ image, the inference time is cut down from original 55s to 47s(-14.5%) after applying SWA on an A800 GPU, and the effect will be more pronounced with higher resolution and more input conditions.

Adaptive Position Encoding. After concatenating conditions into the unified sequence, we need reassign the position index of each image token to reduce condition confusion. FLUX.1-Kontext employs a three-dimensional RoPE [53] scheme that assigns the position indices (i, w, h) to both text and image tokens. In the original setting, text tokens are assigned a consistent position index of (0, 0, 0), while noisy image and reference tokens are allocated shared position indices (w, h) where $w \in [0, W - 1]$ and $h \in [0, H - 1]$. Only one different point lies in the first dimension index that $i = 0$ for noisy token and $i = 1$ for reference tokens. To inherit the original position priors and adapt to multiple conditions, we align the reference image with the noisy image in the diagonal position, as shown in Fig. 2d. We remain the position index of the text and noisy image, and the position index for the token (w, h) in the i_{th} reference image is defined as:

$$(\hat{i}, \hat{w}, \hat{h}) = (i, w_{noisy} + \sum_{j=1}^{i-1} w_{ref_j} + w, h_{noisy} + \sum_{j=1}^{i-1} h_{ref_j} + h) \quad (2)$$

where w_{noisy} and h_{noisy} represent the width and height of the noisy latent, and w_{ref_j} and h_{ref_j} represent the width and height of the j_{th} reference condition latent. This position index assignment can avoid index overlap and encourage the model to distinguish different conditions.

Actually, the size of condition images is not necessarily same as the noisy image. So we need to interpolate position encodings to ensure spatial alignment inspired by UNO [65], and the final position index is represented as:

$$(\hat{i}, \hat{w}, \hat{h}) = (i, w_{noisy} + \sum_{j=1}^{i-1} w_{ref_j} + w * S_w, h_{noisy} + \sum_{j=1}^{i-1} h_{ref_j} + h * S_h) \quad (3)$$

where $S_w = \frac{w_{noisy}}{w_{ref_i}}$, $S_h = \frac{h_{noisy}}{h_{ref_i}}$ are scaling factors in width and height directions.

3.4 Training Strategy.

Multiple Timesteps Prediction. In practice, most Flow Matching implementations employ a single-step prediction objective, where the model is trained to predict the velocity at isolated random time points $t \sim \mathcal{U}[0, 1]$. Although effective, this approach provides no explicit constraint on the temporal consistency of the velocity field across adjacent time steps. Consequently, the learned velocity field may exhibit high-frequency oscillations, leading to unstable numerical integration and degraded generation quality.

In this work, we introduce Multi-Timestep Prediction (MTP), which extends the training objective by unrolling multiple Euler integration steps within a single training iteration and supervising the velocity prediction at each intermediate time point. Single-Timestep Prediction (SSP) trains the model by sampling a single random time t and minimizing the velocity prediction error at that time point, as in Eq. (1). Multi-Timestep Prediction (MTP) unrolls $K - 1$ Euler integration steps from time t to $t - (K - 1)\Delta t$, supervising the velocity prediction at each intermediate step:

$$\mathcal{L}_{\text{MTP}} = \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E} [\|v_{\theta}(x_{t_k}, t_k, c) - (x_1 - x_0)\|^2] \quad (4)$$

where $x_{t_{k+1}} = x_{t_k} + (t_{k+1} - t_k) \cdot v_{\theta}(x_{t_k}, t_k)$ and $t_k = t - k\Delta t$. We prove that MTP implicitly imposes a temporal smoothness constraint on the velocity field, effectively reducing its Lipschitz constant and yielding more stable trajectories. Detailed theoretical analysis is demonstrated in Appendix C.

Alignment Loss. We additionally employ an alignment loss to enhance the fidelity of garment regions [70]. Denoting a feature extraction model as $\mathcal{E}$ (Dinov2 [46] in our work), the generated image as G and the ground-truth (GT) image as GT , we segment the ground-truth’s garment region M in advance, and extract the mask region’s feature to compute the cosine distance between GT-aligned garment features of the generated and ground-truth images as:

$$\mathcal{L}_{\text{align}} = 1 - \cos(\mathcal{E}(M \odot GT), \mathcal{E}(M \odot G)) \quad (5)$$

The overall training objective is a weighted sum of the above two losses:

$$\mathcal{L} = \mathcal{L}_{\text{MTP}} + \lambda \mathcal{L}_{\text{align}} \quad (6)$$

where $\lambda = 0.10$ aims to control the contribution of alignment loss.

4 Experiments

4.1 Experimental Setup

Training Data. To make a fair comparison, we train a unified model on two publicly available datasets: VITON-HD [9], containing 13,679 high-resolution pairs of frontal half-body models and upper-body garments, and DressCode [43] with 53,792 pairs of full-body models and upper-body, lower-body, and dress garments. We use the official train/test splits provided by both datasets. All reference images are resized to 512×768 as input and generate 768×1024 images. In addition, we train a unified model on our Omni-TryOn with a resolution of 768×768 and reference images of 512×512 as supplementary.

Implementation Details. All experiments are conducted using LoRA [25] with a rank of 128 on $8 \times$ NVIDIA H200 GPUs. For SWA, the window size M is 16, and for MTP, we predict 2 timesteps for every optimization step, namely $K = 2$.

Table 1: Quantitative comparison on **VITON-HD** and **DressCode** dataset for the **model-based try-on** task. We multiply KID by 1000 for better comparison. The best and the second best results are denoted as **Bold** and underline, respectively.

Dataset	VITON-HD				DressCode			
Method	FID↓	KID↓	SSIM↑	LPIPS↓	FID↓	KID↓	SSIM↑	LPIPS↓
IDM-VTON [10]	<u>6.1150</u>	1.0458	0.8654	0.0744	5.2637	2.0087	0.8874	0.0723
OOTDiffusion [71]	6.5364	<u>0.9353</u>	0.8403	0.0875	3.9241	0.7381	0.8868	<u>0.0678</u>
StableGarment [61]	9.5081	3.1353	0.8353	0.0925	6.2972	2.5633	0.8762	0.0641
CatVTON [11]	6.1124	0.9448	<u>0.8746</u>	0.0976	3.4869	0.8294	0.8984	0.0869
FastFit [12]	7.3009	1.3189	0.8535	0.1208	3.3824	<u>0.5449</u>	0.8915	0.0696
FitDiT [27]	11.4429	4.2931	0.7827	0.1603	5.6900	2.1629	0.8437	0.1105
Any2anyTryon [18]	14.6365	7.1984	0.7795	0.1888	8.0300	3.5759	0.8305	0.1529
Jco-MVTON [59]	11.7850	4.8869	0.7888	0.1587	8.7186	3.9160	0.8462	0.1183
OmniDiT	6.4564	0.7502	0.8838	<u>0.0784</u>	<u>3.4522</u>	0.5245	<u>0.8980</u>	0.0789

Table 2: Quantitative results on **VITON-HD** dataset for the **model-free try-on** task. The best and the second best results are denoted as **Bold** and underline.

Method	DINO-I ↑	CLIP-I ↑	SSIM ↑	LPIPS ↓	FID ↓	KID ↓	FFA ↑
MagicClothing [7]	0.3091	0.7014	0.5750	0.4689	46.6366	37.0567	0.5619
StableGarment [61]	0.3710	0.7319	0.5331	0.5056	55.9763	21.2100	0.5818
Any2anyTryon [18]	0.2819	0.7880	0.6716	0.3921	73.3430	48.2852	0.5337
DreamFit [36]	0.3626	<u>0.7935</u>	0.6826	<u>0.3039</u>	<u>15.6002</u>	<u>5.4227</u>	0.5683
IMAGDressing [50]	0.3459	0.7845	<u>0.6915</u>	0.3352	17.6286	8.7806	0.5640
OmniDiT	<u>0.3638</u>	0.8022	0.7014	0.2844	12.4448	3.2455	<u>0.5712</u>

We adopt a two-stage training approach: the first stage optimizes only the model-free try-on and try-off tasks, and the second stage optimizes all three tasks for better consistency. In inference, we set the denoising steps to 30 and the guidance scale to 4. Additional details can be found in Appendix B.

4.2 Model-based Try-On

We adopt four popular metrics, including Structural Similarity (SSIM) [63], Learned Perceptual Image Patch Similarity (LPIPS) [78], Fréchet Inception Distance (FID) [22] and Kernel Inception Distance (KID) [5].

For the public benchmarks, the baseline methods contain U-net-based and DiT-based models, including IDM-VTON [10], OOTDiffusion [71], StableGarment [61], CatVTON [11], FastFit [12], FitDiT [27], Any2anyTryon [18], Jco-MVTON [59]. As shown in Tab. 1, our model achieves comparable performance in most metrics and exceeds the SOTA methods in KID and SSIM. Compared with the unified model Any2anyTryon, OmniDiT outperforms it in all metrics, confirming our outstanding ability in producing high-quality and consistent tryon

Table 3: Quantitative comparison on **VITON-HD** dataset for the **try-off** task. The best and the second best results are denoted as **Bold** and underline, respectively.

Method	CLIP-I $\uparrow$	ms-SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	clip-FID $\downarrow$	KID $\downarrow$	DISTS $\downarrow$
TryOffDiff [57]	0.9177	0.6991	<u>0.1933</u>	17.1050	7.0578	5.2198	0.2297
MGT [58]	0.9015	0.6443	0.2508	21.8485	8.0539	8.8200	0.2716
TryOffAnyone [66]	<u>0.9242</u>	0.6419	0.2109	<u>11.3476</u>	<u>5.1583</u>	1.9186	<u>0.2169</u>
Any2anyTryon [18]	0.8749	0.5607	0.3126	20.4215	8.0235	5.4847	0.2972
OmniDiT	0.9391	<u>0.6782</u>	0.1859	10.5479	2.4875	1.5611	0.1958

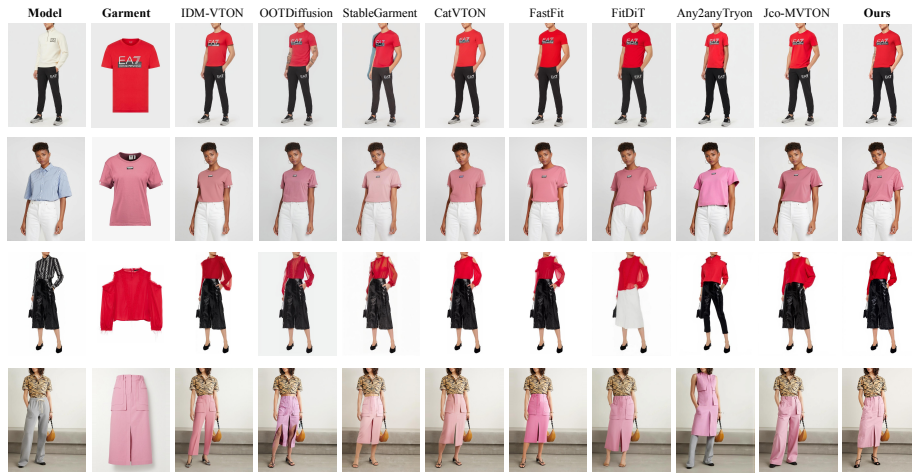


Fig. 3: Qualitative comparison of the model-based try-on generation results on the VITON-HD and DressCode benchmarks. Please zoom in to see details preservation.

images. In Fig. 3, we show some qualitative cases on public benchmarks. Our method consistently excels in preserving key attributes: text and logo on the red shirt, color on the pink shirt, texture, and details on the skirts.

In addition, we compare some methods on our Omni-TryOn benchmark. Besides professional try-on methods, we also test general in-context generation methods, including UNO [65], DreamOmni2 [67], DreamO [44], OmniGen2 [64]. Quantitative results in Tab. 6 demonstrate our method’s eminent performance in handling complex scenes. In Fig. 4, professional try-on methods cannot deal with complex scenes and postures, and other general methods are limited to weak consistency. In contrast, our method fully maintains the garment’s details and renders it on the human perfectly, verifying our outstanding performance. More visualization examples can be found in Figs. 11 and 12.



Fig. 4: Qualitative comparison of the model-based try-on generation results on our Omni-TryOn benchmark. All methods adopt their optimal resolutions.

4.3 Model-free Try-On

For the model-free try-on task, quantitative metrics include: the visual similarity using DINO [46] and CLIP [48], SSIM, LPIPS, FID, KID and FFA [29]. Baselines contain MagicClothing [7], StableGarment [61], Any2anyTryon [18], DreamFit [36], IMAGDressing [50]. The model-free try-on task requires the try-on system to imagine a suitable human model based on the given text prompt, and wear the given garment, thus being more challenging than the model-based try-on task. Our method shows a strong capability to create a real human model and maintain better fidelity in texture, color, and details in Fig. 5.

Quantitative results in Tabs. 2, 7 and 8 verify that in both ‘in shop’ and ‘in the wild’ scenes, our method significantly outperforms all baselines, especially in terms of FID and KID, indicating that our model can capture comprehensive and fine-grained features from reference garments and fit the garment well into the generated human model. More visualization examples can be found in Fig. 13.

4.4 Try-Off

For the try-off task, baselines include TryOffDiff [57], MGT [58], TryOffAnyone [66], Any2anyTryon [18]. As the inverse task of VTON, try-off needs to display the garment’s full view and preserve all details, like color, logo, texture and texts. In Tabs. 3, 9 and 10, our method remarkably outperforms all baselines on public and our benchmarks, confirming that our method takes good command of try-on and try-off abilities. Their detail preservation ability is mutually reinforcing and does not conflict. In Fig. 6, it is evident that the realism and fidelity of our try-off results are far superior to those of Any2anyTryon, verifying the validation of our unified model. More visualization examples can be found in Fig. 14.

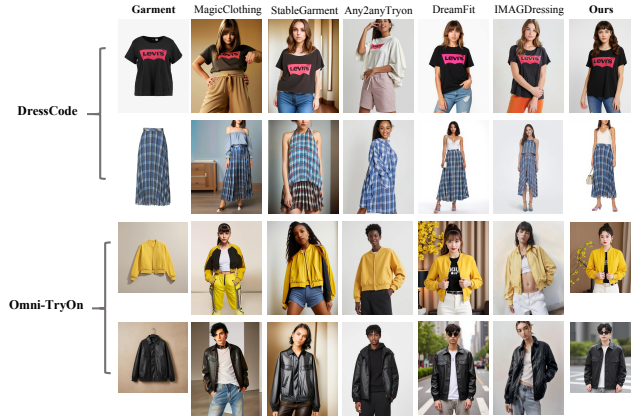


Fig. 5: Qualitative comparison of the model-free try-on generation results. All methods adopt their optimal resolution. Please zoom in to see details preservation.



Fig. 6: Qualitative comparison of the try-off generation results. All methods adopt their optimal resolution. Please zoom in to see details preservation.

4.5 Ablation Study

We conduct ablation experiments to study the effect of Shift Window Attention, Multiple Timesteps Prediction and Alignment Loss. As shown in Tab. 4 and Fig. 7, compared with original full attention ($ws=0$), a small window size will degrade performance, which is reasonable that the small window prevents attending further regions and limits the overall perspective. Because a larger window is approximately equivalent to the global latent size and the performance gap between $ws = 16$ and full attention is not intolerable, we set $ws = 16$ as the final choice. After employing MTP, four metrics have shown notable improvements, in which FID reduces from 8.2141 to 7.7120, and KID reduces from 1.8236 to 1.1282, indicating that MTP has positive effects on improving generation fidelity, due to achieving more stable and smooth trajectories, and lowering integration error, as analyzed in Appendix C.5.



Fig. 7: Qualitative comparisons on different window sizes and training strategies.

Table 4: Ablation study on **VITON-HD** dataset for the **model-based try-on** task. ref-512 + ws-16 + mtp-2 denotes that the reference condition is 512×512 , local window size is 16 and 2 timesteps prediction.

Method	Alignment	FID↓	KID↓	SSIM↑	LPIPS↓
ref-512 + ws-0 + mtp-1	✗	7.6341	1.2143	0.8607	0.1070
ref-512 + ws-4 + mtp-1	✗	11.0544	2.6720	0.8531	0.1392
ref-512 + ws-8 + mtp-1	✗	10.2492	2.6107	0.8680	0.1159
ref-512 + ws-16 + mtp-1	✗	8.5141	1.7236	0.8683	0.1090
ref-512 + ws-32 + mtp-1	✗	8.3264	1.7037	0.8712	0.1095
ref-384 + ws-16 + mtp-2	✗	8.6728	1.3855	0.8738	0.1075
ref-512 + ws-16 + mtp-2	✗	7.6120	0.9082	0.8781	0.0951
ref-768 + ws-16 + mtp-2	✗	7.5431	0.8455	0.8800	0.0873
ref-512 + ws-16 + mtp-2	✓	6.4564	0.7502	0.8838	0.0784

In Tab. 4, we also compare different resolutions for the reference conditions, and the result of $384 < 512 < 768$ demonstrates that larger reference images will reduce the loss of detail information and ensure generation quality at the cost of a longer inference time. After adding the alignment loss, it is evident to see a significant performance improvement, and even the metrics surpass those of ref-768, which verifies the validation of alignment loss.

4.6 Limitations

Despite the outstanding performance OmniDiT has achieved, we still face a major challenge: OmniDiT struggles to maintain the human model attributes, especially the human model’s gesture. This mainly stems from that the inpainted human is slightly different from the tryon results in the data curation stage and not filtered out due to the VLM’s weak visual ability, thus confusing the optimization objective in training afterwards. Therefore, research on how to develop a stronger VLM can benefit our data quality and model improvement, which is our future direction.

5 Conclusion

In this paper, we propose one unified VTON and VTOFF framework OmniDiT based on the Diffusion Transformer, demonstrating outstanding performance in three VTON and VTOFF tasks. To incorporate multiple reference conditions, we design an adaptive positive encoding to reduce token signal confusion induced by concatenating all tokens into one sequence. Meanwhile, we introduce the Shifted Window Attention to relieve the bottleneck of attention computation, thus achieving a linear computation complexity. Multiple timesteps prediction and alignment loss further improve the generation quality and fidelity. We also construct a large VTON dataset produced by our self-evolving data curation pipeline, aiming to address the data scarcity problem. Experiments reveal that our model has a significant advantage in generating high-fidelity and consistent try-on and try-off results under various complex scenes.

Acknowledgements

Special thanks to Kuaishou Technology for supporting this research.

References

1. Atef, M., Ayman, M., Rashed, A., Saeed, A., Saeed, A., Fares, A.: Efficientviton: An efficient virtual try-on model using optimized diffusion process. arXiv preprint arXiv:2501.11776 (2025)
2. Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.): Computer Vision – ECCV 2022. Springer (2022). <https://doi.org/10.1007/978-3-031-19769-7>
3. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., et al.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)
4. Baldrati, A., Morelli, D., Cartella, G., Cornia, M., Bertini, M., Cucchiara, R.: Multimodal garment designer: Human-centric latent diffusion models for fashion image editing. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 23393–23402 (2023)
5. Bińkowski, M., Sutherland, D.J., Arbel, M., Gretton, A.: Demystifying mmd gans. arXiv preprint arXiv:1801.01401 (2018)
6. Chang, T., Chen, X., Wei, Z., Zhang, X., Chen, Q., Luo, W., Song, P., Yang, X.: Pemf-vto: Point-enhanced video virtual try-on via mask-free paradigm. IEEE Transactions on Consumer Electronics (2025)
7. Chen, W., Gu, T., Xu, Y., Chen, A.: Magic clothing: Controllable garment-driven image synthesis. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 6939–6948 (2024)
8. Chen, X., Huang, L., Liu, Y., Shen, Y., Zhao, D., Zhao, H.: Anydoor: Zero-shot object-level image customization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6593–6602 (2024)
9. Choi, S., Park, S., Lee, M., Choo, J.: Viton-hd: High-resolution virtual try-on via misalignment-aware normalization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14131–14140 (2021)

10. Choi, Y., Kwak, S., Lee, K., Choi, H., Shin, J.: Improving diffusion models for authentic virtual try-on in the wild. In: European Conference on Computer Vision. pp. 206–235. Springer (2024)
11. Chong, Z., Dong, X., Li, H., Zhang, S., Zhang, W., Zhang, X., Zhao, H., Jiang, D., Liang, X.: Catvton: Concatenation is all you need for virtual try-on with diffusion models. arXiv preprint arXiv:2407.15886 (2024)
12. Chong, Z., Lei, Y., Zhang, S., He, Z., Wang, Z., Zhang, X., Dong, X., Wu, Y., Jiang, D., Liang, X.: Fastfit: Accelerating multi-reference virtual try-on via cacheable diffusion models. arXiv preprint arXiv:2508.20586 (2025)
13. Ge, C., Song, Y., Ge, Y., Yang, H., Liu, W., Luo, P.: Disentangled cycle consistency for highly-realistic virtual try-on. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16928–16937 (2021)
14. Ge, Y., Song, Y., Zhang, R., Ge, C., Liu, W., Luo, P.: Parser-free virtual try-on via distilling appearance flows. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8485–8493 (2021)
15. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. *Advances in neural information processing systems* **27** (2014)
16. Google: Introducing gemini 2.5 flash image (2025), <https://developers.googleblog.com/en/introducing-gemini-2-5-flash-image/>
17. Gou, J., Sun, S., Zhang, J., Si, J., Qian, C., Zhang, L.: Taming the power of diffusion models for high-quality virtual try-on with appearance flow. In: Proceedings of the 31st ACM International Conference on Multimedia. pp. 7599–7607 (2023)
18. Guo, H., Zeng, B., Song, Y., Zhang, W., Liu, J., Zhang, C.: Any2anytryon: Leveraging adaptive position embeddings for versatile virtual clothing tasks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19085–19096 (2025)
19. Han, X., Hu, X., Huang, W., Scott, M.R.: Clothflow: A flow-based model for clothed person generation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10471–10480 (2019)
20. Han, X., Wu, Z., Wu, Z., Yu, R., Davis, L.S.: Viton: An image-based virtual try-on network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7543–7552 (2018)
21. He, S., Song, Y.Z., Xiang, T.: Style-based global appearance flow for virtual try-on. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3470–3479 (2022)
22. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems* **30** (2017)
23. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models (2020), <https://arxiv.org/abs/2006.11239>
24. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. *Advances in neural information processing systems* **33**, 6840–6851 (2020)
25. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. *Iclr* **1**(2), 3 (2022)
26. Iserles, A.: A first course in the numerical analysis of differential equations. Cambridge University Press (2009)
27. Jiang, B., Hu, X., Luo, D., He, Q., Xu, C., Peng, J., Zhang, J., Wang, C., Wu, Y., Fu, Y.: Fitdit: Advancing the authentic garment details for high-fidelity virtual try-on. arXiv preprint arXiv:2411.10499 (2024)

28. Kim, J., Gu, G., Park, M., Park, S., Choo, J.: Stableviton: Learning semantic correspondence with latent diffusion model for virtual try-on. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8176–8185 (2024)
29. Kotar, K., Tian, S., Yu, H.X., Yamins, D., Wu, J.: Are these the same apple? comparing images based on object intrinsics. *Advances in Neural Information Processing Systems* **36**, 40853–40871 (2023)
30. Labs, B.F., Batifol, S., Blattmann, A., Boesel, F., Consul, S., Diagne, C., Dockhorn, T., English, J., English, Z., Esser, P., et al.: Flux. 1 kontekst: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742* (2025)
31. Lee, S., Gu, G., Park, S., Choi, S., Choo, J.: High-resolution virtual try-on with misalignment and occlusion-handled conditions. In: *European Conference on Computer Vision*. pp. 204–219. Springer (2022)
32. Lee, S., Kwak, J.g.: Voost: A unified and scalable diffusion transformer for bidirectional virtual try-on and try-off. In: *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*. pp. 1–11 (2025)
33. Lewis, K.M., Varadharajan, S., Kemelmacher-Shlizerman, I.: Tryongan: Body-aware try-on via layered interpolation (2021), <https://arxiv.org/abs/2101.02285>
34. Li, K., jin Chong, M., Zhang, J., Liu, J.: Toward accurate and realistic outfits visualization with attention to details (2021), <https://arxiv.org/abs/2106.06593>
35. Li, Q., Qiu, S., Han, J., Xu, X., Seyfioglu, M.S., Koo, K.K., Bouyarmane, K.: Dittvton: Diffusion transformer framework for unified multi-category virtual try-on and virtual try-all with integrated image editing. *arXiv preprint arXiv:2510.04797* (2025)
36. Lin, E., Zhang, X., Zhao, F., Luo, Y., Dong, X., Zeng, L., Liang, X.: Dreamfit: Garment-centric human generation via a lightweight anything-dressing encoder. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. vol. 39, pp. 5218–5226 (2025)
37. Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747* (2022)
38. Liu, X., Gong, C., Liu, Q.: Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003* (2022)
39. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: *Proceedings of the IEEE/CVF international conference on computer vision*. pp. 10012–10022 (2021)
40. Loshchilov, I., Hutter, F., et al.: Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101* **5**(5), 5 (2017)
41. Men, Y., Mao, Y., Jiang, Y., Ma, W.Y., Lian, Z.: Controllable person image synthesis with attribute-decomposed gan. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 5084–5093 (2020)
42. Morelli, D., Baldrati, A., Cartella, G., Cornia, M., Bertini, M., Cucchiara, R.: Ladvton: Latent diffusion textual-inversion enhanced virtual try-on. In: *Proceedings of the 31st ACM international conference on multimedia*. pp. 8580–8589 (2023)
43. Morelli, D., Fincato, M., Cornia, M., Landi, F., Cesari, F., Cucchiara, R.: Dress code: High-resolution multi-category virtual try-on. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 2231–2235 (2022)

44. Mou, C., Wu, Y., Wu, W., Guo, Z., Zhang, P., Cheng, Y., Luo, Y., Ding, F., Zhang, S., Li, X., et al.: Dreamo: A unified framework for image customization. In: Proceedings of the SIGGRAPH Asia 2025 Conference Papers. pp. 1–12 (2025)
45. Niu, Y., Yi, D., Wu, L., Liu, Z., Cai, P., Wang, J.: Pfdm: Parser-free virtual try-on via diffusion model (2024), <https://arxiv.org/abs/2402.03047>
46. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
47. Peebles, W., Xie, S.: Scalable diffusion models with transformers (2023), <https://arxiv.org/abs/2212.09748>
48. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PmLR (2021)
49. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models (2022), <https://arxiv.org/abs/2112.10752>
50. Shen, F., Jiang, X., He, X., Ye, H., Wang, C., Du, X., Li, Z., Tang, J.: Imagdressing-v1: Customizable virtual dressing. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 6795–6804 (2025)
51. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. arXiv preprint arXiv:2011.13456 (2020)
52. Song, Y., Liu, C., Shou, M.Z.: Omniconsistency: Learning style-agnostic consistency from paired stylization data. arXiv preprint arXiv:2505.18445 (2025)
53. Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., Liu, Y.: Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing* **568**, 127063 (2024)
54. Sun, K., Cao, J., Wang, Q., Tian, L., Zhang, X., Zhuo, L., Zhang, B., Bo, L., Zhou, W., Zhang, W., et al.: Outfitanyone: Ultra-high quality virtual try-on for any clothing and any person. arXiv preprint arXiv:2407.16224 (2024)
55. Tan, Z., Liu, S., Yang, X., Xue, Q., Wang, X.: Ominicontrol: Minimal and universal control for diffusion transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14940–14950 (2025)
56. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. *Advances in neural information processing systems* **30** (2017)
57. Velioglu, R., Bevandic, P., Chan, R., Hammer, B.: Tryoffdiff: Virtual-try-off via high-fidelity garment reconstruction using diffusion models. arXiv preprint arXiv:2411.18350 (2024)
58. Velioglu, R., Bevandic, P., Chan, R., Hammer, B.: Mgt: Extending virtual try-off to multi-garment scenarios. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6039–6048 (2025)
59. Wang, A., Li, W., Luo, H., Ao, M., Zhu, C., Li, X., Wang, F.: Jco-mvton: Jointly controllable multi-modal diffusion transformer for mask-free virtual try-on. arXiv preprint arXiv:2508.17614 (2025)
60. Wang, B., Zheng, H., Liang, X., Chen, Y., Lin, L., Yang, M.: Toward characteristic-preserving image-based virtual try-on network. In: Proceedings of the European conference on computer vision (ECCV). pp. 589–604 (2018)
61. Wang, R., Guo, H., Liu, J., Li, H., Zhao, H., Tang, X., Hu, Y., Tang, H., Li, P.: Stablegarment: Garment-centric generation via stable diffusion. arXiv preprint arXiv:2403.10783 (2024)

62. Wang, W., Gao, Z., Gu, L., Pu, H., Cui, L., Wei, X., Liu, Z., Jing, L., Ye, S., Shao, J., et al.: Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. arXiv preprint arXiv:2508.18265 (2025)
63. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* **13**(4), 600–612 (2004)
64. Wu, C., Zheng, P., Yan, R., Xiao, S., Luo, X., Wang, Y., Li, W., Jiang, X., Liu, Y., Zhou, J., et al.: Omnigen2: Exploration to advanced multimodal generation. arXiv preprint arXiv:2506.18871 (2025)
65. Wu, S., Huang, M., Wu, W., Cheng, Y., Ding, F., He, Q.: Less-to-more generalization: Unlocking more controllability by in-context generation. arXiv preprint arXiv:2504.02160 (2025)
66. Xarchakos, I., Koukopoulos, T.: Tryoff anyone: Tiled cloth generation from a dressed person. arXiv preprint arXiv:2412.08573 (2024)
67. Xia, B., Peng, B., Zhang, Y., Huang, J., Liu, J., Li, J., Tan, H., Wu, S., Wang, C., Wang, Y., et al.: Dreamomni2: Multimodal instruction-based editing and generation. arXiv preprint arXiv:2510.06679 (2025)
68. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. *CoRR abs/2105.15203* (2021), <https://arxiv.org/abs/2105.15203>
69. Xie, Z., Huang, Z., Dong, X., Zhao, F., Dong, H., Zhang, X., Zhu, F., Liang, X.: Gp-vton: Towards general purpose virtual try-on via collaborative local-flow global-parsing learning. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 23550–23559 (2023)
70. Xu, H., Cheng, W., Xing, P., Fang, Y., Wu, S., Wang, R., Zeng, X., Jiang, D., Yu, G., Ma, X., et al.: Withanyone: Towards controllable and id consistent image generation. arXiv preprint arXiv:2510.14975 (2025)
71. Xu, Y., Gu, T., Chen, W., Chen, A.: Ootdiffusion: Outfitting fusion based latent diffusion for controllable virtual try-on. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. vol. 39, pp. 8996–9004 (2025)
72. Yang, H., Zhang, R., Guo, X., Liu, W., Zuo, W., Luo, P.: Towards photo-realistic virtual try-on by adaptively generating-preserving image content. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 7850–7859 (2020)
73. Yang, Z., Chen, J., Shi, Y., Li, H., Chen, T., Lin, L.: Occlumix: Towards de-occlusion virtual try-on by semantically-guided mixup. *IEEE Transactions on Multimedia* **25**, 1477–1488 (2023)
74. Ye, H., Zhang, J., Liu, S., Han, X., Yang, W.: Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. arXiv preprint arXiv:2308.06721 (2023)
75. Zeng, J., Song, D., Nie, W., Tian, H., Wang, T., Liu, A.A.: Cat-dm: Controllable accelerated virtual try-on with diffusion model. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 8372–8382 (2024)
76. Zeng, W., Zhao, M., Gao, Y., Zhang, Z.: Tilegan: category-oriented attention-based high-quality tiled clothes generation from dressed person. *Neural Computing and Applications* **32**(23), 17587–17600 (2020)
77. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: *Proceedings of the IEEE/CVF international conference on computer vision*. pp. 3836–3847 (2023)

78. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 586–595 (2018)
79. Zhang, S., Chong, Z., Zhang, X., Li, H., Cheng, Y., Yan, Y., Liang, X.: Garmental-igner: Text-to-garment generation via retrieval-augmented multi-level corrections. In: European Conference on Computer Vision. pp. 148–164. Springer (2024)
80. Zhang, X., Song, D., Zhan, P., Chang, T., Zeng, J., Chen, Q., Luo, W., Liu, A.: Boow-vton: Boosting in-the-wild virtual try-on via mask-free pseudo data training (2024), <https://arxiv.org/abs/2408.06047>
81. Zhang, X., Sha, Y., Kampffmeyer, M.C., Xie, Z., Jie, Z., Huang, C., Peng, J., Liang, X.: Armani: Part-level garment-text alignment for unified cross-modal fashion design. In: Proceedings of the 30th ACM International Conference on Multimedia. pp. 4525–4535 (2022)
82. Zhang, X., Yang, B., Kampffmeyer, M.C., Zhang, W., Zhang, S., Lu, G., Lin, L., Xu, H., Liang, X.: Diffcloth: Diffusion based garment synthesis and manipulation via structural cross-modal semantic alignment. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 23154–23163 (2023)
83. Zhang, Y., Yuan, Y., Song, Y., Wang, H., Liu, J.: Easycontrol: Adding efficient and flexible control for diffusion transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19513–19524 (2025)
84. Zhou, Z., Liu, S., Han, X., Liu, H., Ng, K.W., Xie, T., Cong, Y., Li, H., Xu, M., Pérez-Rúa, J.M., et al.: Learning flow fields in attention for controllable person image generation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 2491–2501 (2025)
85. Zhu, L., Yang, D., Zhu, T., Reda, F., Chan, W., Saharia, C., Norouzi, M., Kemelmacher-Shlizerman, I.: Tryondiffusion: A tale of two unets. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4606–4615 (2023)

A Dataset Details

As illustrated in Sec. 3.2, we carry out several rounds of data curation iterations, and add additional data samples from two public datasets [9, 43] as a supplementary, thus obtaining a total of over 380k samples. Among all samples, each garment is labeled with one category and there are 23 categories in our dataset, as shown in Fig. 8. We keep the original garment labels for the public dataset samples. For model evaluation, we sample 1895 samples to establish the test set with stratified sampling based on the category distribution. Some data samples can be viewed in Fig. 9. In Tab. 5, we compare our Omni-TryOn with other popular Try-On datasets, it is evident that our dataset is largest, most complete, and diverse. Our dataset equipped with detailed task instructions and matching human models, can make up the limitation of current try-on datasets and promote the VTON and VTOFF technology’s development.

Table 5: Comparison between our Omni-TryOn and related try-on datasets. ‘Prompts’ and ‘Models’ mean that dataset contains detailed text instructions for three tasks and matching models for the model-based try-on task.

Dataset	Prompts	Models	#Garments	#Pairs	Resolution
TryOnGAN [33]	✗	✗	52,000	52,000	512×512
Revery AI [34]	✗	✗	321,000	321,000	512×512
VITON-HD [9]	✗	✗	13,679	13,679	1024×768
DressCode [43]	✗	✗	53,792	53,792	1024×768
IGPair [50]	✓	✗	86,873	324,857	$> 2K \times 2K$
Omni-TryOn(Ours)	✓	✓	335,146	382,223	diverse

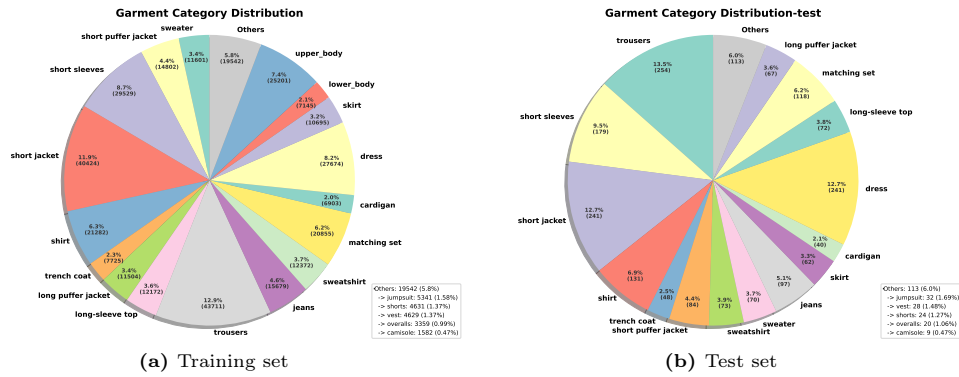


Fig. 8: Garment category distribution of our Omni-TryOn dataset.

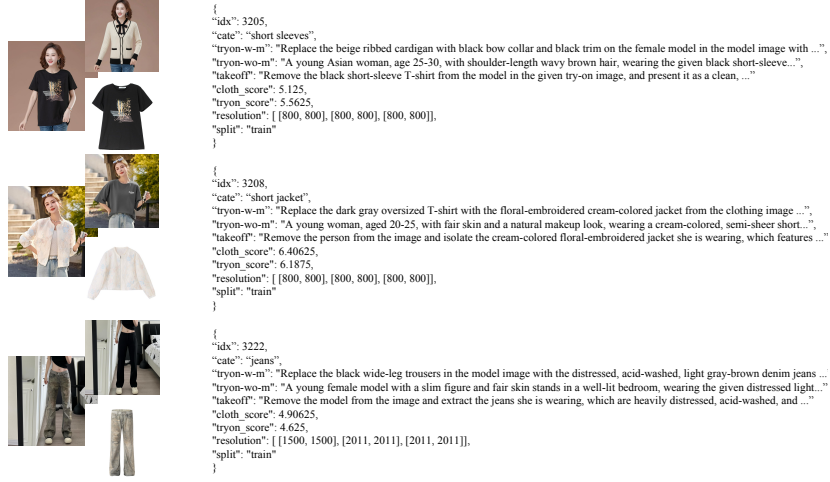


Fig. 9: Some data sample showcases in our dataset Omni-TryOn.

B Experiment Settings

B.1 Model Architecture

We replace all the attention modules in MM-DiT blocks (including double-stream and single-stream) with our introduced shifted window attention, and shift the windows every two layers. The window size is 16 and the shift size is 8. All local window attention computations are applied to the reference images.

B.2 Training Settings

Inspired by UNO [65], with the aim of progressive cross-modal alignment, we adopt a two-stage training approach. For the first stage, we train the model using single-reference pair data (model-free try-on and try-off tasks) for one epoch. Then, we continue training on mixed reference pair data (all three tasks) for 2-3 epochs. For public datasets, both stages utilize the learning rate of $4e - 5$ and a batch size of 2 for each GPU (a total batch size of 16). For our own dataset, both stages utilize the learning rate of $4e - 5$, a batch size of 2 for each GPU (a total batch size of 16) for the first stage and a batch size of 4 for each GPU (a total batch size of 32) for the second stage. In all trainings, we use the AdamW optimizer [40].

For public datasets, we resize the reference images to 512×768 as input, and generate the target images with a resolution of 768×1024 . And for our own dataset Omni-TryOn, the reference images are 512×512 and target images are 768×768 .

For MTP, we predict two timesteps for every iteration, and the time interval Δt is 30 in training.

To ensure the aesthetics of generated garments and tryon results, we sample some proportions of data whose garment and tryon images’ aesthetic scores are larger than 4.5 for model-free try-on and try-off tasks, and the rest of the datasets for model-based try-on task to improve the consistency.

B.3 Batch Sampling Strategy

Considering the mixed number of reference images in the second training stage (one reference image for model-free try-on and try-off tasks, and two reference images for model-based try-on task), we strictly restrict that one batch has the same number of reference images. In other words, in all optimization steps, the model-based try-on samples are isolated from the other two tasks.

C Theoretical Analysis

C.1 Flow Matching Framework

Let p_0 and p_1 be probability distributions over $\mathbb{R}^d$, representing the source and target distributions respectively. Flow Matching seeks to learn a velocity field $v_\theta : \mathbb{R}^d \times [0, 1] \rightarrow \mathbb{R}^d$ such that the solution to the initial value problem:

$$\frac{dx}{dt} = v_\theta(x, t), \quad x(0) \sim p_0 \quad (7)$$

satisfies $x(1) \sim p_1$. In practice, the velocity field is parameterized by a neural network and trained using a conditional flow matching objective.

For a pair of samples (x_0, x_1) where $x_0 \sim p_0$ and $x_1 \sim p_1$, the conditional flow is defined as:

$$\phi_t(x_0, x_1) = (1 - t)x_0 + tx_1 \quad (8)$$

and the corresponding conditional velocity field is:

$$u_t(x|x_0, x_1) = x_1 - x_0 \quad (9)$$

The standard Flow Matching loss minimizes the discrepancy between the learned velocity v_θ and the conditional velocity u_t :

$$\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{\substack{t \sim \mathcal{U}[0,1] \\ (x_0, x_1) \sim p_0 \times p_1}} [\|v_\theta(\phi_t(x_0, x_1), t) - (x_1 - x_0)\|^2] \quad (10)$$

C.2 Single-Step vs. Multi-Timestep Prediction

Single-Step Prediction (SSP) trains the model by sampling a single random time t and minimizing the velocity prediction error at that time point:

$$\mathcal{L}_{\text{SSP}} = \mathbb{E}_{t, x_t} [\|v_\theta(x_t, t) - (x_1 - x_0)\|^2] \quad (11)$$

Multi-Timestep Prediction (MTP) extends this by unrolling $K - 1$ Euler integration steps from time t to $t - (K - 1)\Delta t$, supervising the velocity prediction at each intermediate step:

$$\mathcal{L}_{\text{MTP}} = \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E} [\|v_\theta(x_{t_k}, t_k) - (x_1 - x_0)\|^2] \quad (12)$$

where $x_{t_{k+1}} = x_{t_k} + (t_{k+1} - t_k) \cdot v_\theta(x_{t_k}, t_k)$ and $t_k = t - k\Delta t$.

C.3 Lipschitz Continuity and Velocity Field Smoothness

A key property of well-behaved velocity fields is Lipschitz continuity, which ensures stable numerical integration.

Definition 1 (Lipschitz Continuity). *A velocity field $v : \mathbb{R}^d \times [0, 1] \rightarrow \mathbb{R}^d$ is L -Lipschitz continuous if:*

$$\|v(x, t) - v(x', t')\| \leq L(\|x - x'\| + |t - t'|) \quad (13)$$

for all $x, x' \in \mathbb{R}^d$ and $t, t' \in [0, 1]$.

The Lipschitz constant L quantifies the smoothness of the velocity field. **Smaller L implies smoother trajectories and lower numerical integration error.**

C.4 Theoretical Results

We now present our main theoretical results establishing the advantages of MTP.

Theorem 1 (Implicit Smoothness Regularization). *Multi-Step Prediction implicitly regularizes the temporal smoothness of the velocity field. Specifically, the MTP loss can be decomposed as:*

$$\mathcal{L}_{\text{MTP}} = \mathcal{L}_{\text{SSP}} + \lambda \cdot \mathcal{R}_{\text{smooth}} + \mathcal{O}(\Delta t^2) \quad (14)$$

where $\lambda > 0$ and the smoothness regularizer is:

$$\mathcal{R}_{\text{smooth}} = \mathbb{E} \left[\sum_{k=0}^{K-2} \|v_\theta(x_{t_{k+1}}, t_{k+1}) - v_\theta(x_{t_k}, t_k)\|^2 \right] \quad (15)$$

Proof. Consider two consecutive time steps t_k and $t_{k+1} = t_k - \Delta t$. By the triangle inequality:

$$\begin{aligned} & \|v_\theta(x_{t_{k+1}}, t_{k+1}) - v_\theta(x_{t_k}, t_k)\| \\ & \leq \|v_\theta(x_{t_{k+1}}, t_{k+1}) - u_{t_{k+1}}\| + \|u_{t_{k+1}} - u_{t_k}\| + \|u_{t_k} - v_\theta(x_{t_k}, t_k)\| \\ & \leq \|v_\theta(x_{t_{k+1}}, t_{k+1}) - u_{t_{k+1}}\| + \|u_{t_k} - v_\theta(x_{t_k}, t_k)\| + \mathcal{O}(\Delta t) \end{aligned} \quad (16)$$

where $u_t = x_1 - x_0$ is the ground-truth conditional velocity, and we used the fact that u_t is constant with respect to t .

Squaring both sides and taking expectations:

$$\begin{aligned} & \mathbb{E} [\|v_\theta(x_{t_{k+1}}, t_{k+1}) - v_\theta(x_{t_k}, t_k)\|^2] \\ & \leq 2\mathbb{E} [\|v_\theta(x_{t_{k+1}}, t_{k+1}) - u_{t_{k+1}}\|^2] + 2\mathbb{E} [\|v_\theta(x_{t_k}, t_k) - u_{t_k}\|^2] + \mathcal{O}(\Delta t^2) \\ & = 2\mathcal{L}_{\text{SSP}}(t_{k+1}) + 2\mathcal{L}_{\text{SSP}}(t_k) + \mathcal{O}(\Delta t^2) \end{aligned} \quad (17)$$

Summing over all consecutive pairs and rearranging:

$$\mathcal{L}_{\text{MTP}} = \frac{1}{K} \sum_{k=0}^{K-1} \mathcal{L}_{\text{SSP}}(t_k) \geq \mathcal{L}_{\text{SSP}} + \frac{1}{2K} \mathcal{R}_{\text{smooth}} - \mathcal{O}(\Delta t^2) \quad (18)$$

which completes the proof with $\lambda = \frac{1}{2K}$.

Remark 1. Theorem 1 shows that MTP penalizes large changes in the velocity field between adjacent time steps. This encourages the model to learn a temporally consistent velocity field with smaller Lipschitz constant.

Theorem 2 (Trajectory Integration Error Bound). *Let v_θ be an L_θ -Lipschitz velocity field learned via Flow Matching, and let x_0^{gen} be the result of numerically integrating the ODE $\frac{dx}{dt} = v_\theta(x, t)$ from x_1 to $t = 0$ using Euler’s method with step size Δt . Then the trajectory error satisfies:*

$$\|x_0^{\text{gen}} - x_0\| \leq C \cdot L_\theta \cdot \Delta t + \mathcal{O}(\Delta t^2) \quad (19)$$

where C is a constant independent of L_θ and Δt .

Proof. The global truncation error of Euler’s method for an L -Lipschitz ODE is bounded by [26]:

$$\|x(t) - x_{\text{num}}(t)\| \leq \frac{M}{2L} (e^{Lt} - 1) \Delta t \quad (20)$$

where M bounds the second derivative of the true solution. For our linear conditional flow $\phi_t(x_0, x_1) = (1-t)x_0 + tx_1$, we have $M = 0$, but the error arises from the discrepancy between v_θ and the true velocity u_t .

Let $\epsilon(t) = v_\theta(x_t, t) - u_t$ be the velocity prediction error. The accumulated error after $N = 1/\Delta t$ steps is:

$$\begin{aligned} \|x_0^{\text{gen}} - x_0\| &= \left\| \sum_{k=0}^{N-1} [v_\theta(x_{t_k}, t_k) - u_{t_k}] \Delta t \right\| \\ &\leq \sum_{k=0}^{N-1} \|\epsilon(t_k)\| \Delta t \end{aligned} \quad (21)$$

By the Lipschitz property of v_θ and the fact that u_t is constant:

$$\begin{aligned}
\|\epsilon(t_{k+1}) - \epsilon(t_k)\| &= \|v_\theta(x_{t_{k+1}}, t_{k+1}) - v_\theta(x_{t_k}, t_k)\| \\
&\leq L_\theta(\|x_{t_{k+1}} - x_{t_k}\| + \Delta t) \\
&\leq L_\theta(\|v_\theta\| \Delta t + \Delta t) \\
&\leq L_\theta C' \Delta t
\end{aligned} \tag{22}$$

where $C' = \|v_\theta\| + 1$.

This implies that the error $\epsilon(t)$ cannot change too rapidly, and the accumulated error is bounded by:

$$\|x_0^{\text{gen}} - x_0\| \leq C \cdot L_\theta \cdot \Delta t + \mathcal{O}(\Delta t^2) \tag{23}$$

where C absorbs constants related to $\|v_\theta\|$ and the number of steps.

Corollary 1. *If MTP reduces the Lipschitz constant of the learned velocity field such that $L_\theta^{\text{MTP}} < L_\theta^{\text{SSP}}$, then:*

$$\|x_0^{\text{gen, MTP}} - x_0\| < \|x_0^{\text{gen, SSP}} - x_0\| \tag{24}$$

for sufficiently small Δt .

C.5 Empirical Results

We employ the SSP and MTP model weights to generate 1500 try-on results under the same settings. As illustrated in Fig. 10, the Lipschitz constants of the MTP model are apparently lower than those of the SSP model, which corresponds to the conclusion of Theorem 2.

Above all, we demonstrate that our MTP training strategy can enhance the smoothness of trajectory prediction and achieve lower integration error from both theoretical and empirical results.

D Additional Results

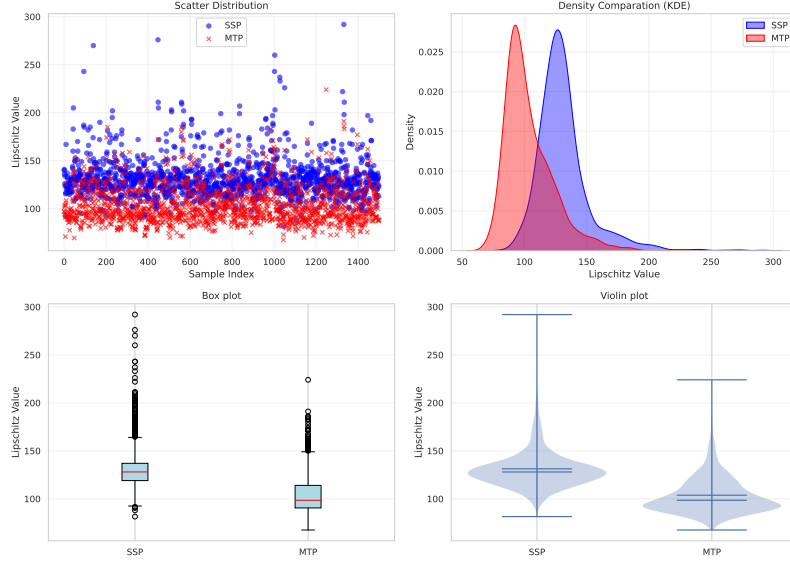


Fig. 10: Lipschitz constant comparison between SSP and MTP.

Table 6: Quantitative comparison on **Omni-TryOn** dataset for the **model-based try-on** task. We multiply KID by 1000 for better comparison. The best and the second best results are denoted as **Bold** and underline, respectively.

Dataset Method	Omni-TryOn			
	FID↓	KID↓	SSIM↑	LPIPS↓
CatVTOn [11]	36.1327	15.3471	0.5904	0.3497
FitDiT [27]	25.1988	7.0999	0.6673	<u>0.2440</u>
Any2anyTryon [18]	25.9283	9.7514	0.6537	0.2954
Jco-MVTOn [59]	27.5184	7.9581	<u>0.6702</u>	0.2449
DreamOmni2 [67]	<u>20.7575</u>	<u>3.0112</u>	0.6444	0.2645
DreamO [44]	32.6664	9.6479	0.4688	0.5072
UNO [65]	31.3706	9.0984	0.5470	0.4775
OmniGen2 [64]	25.1083	4.7097	0.5277	0.4315
OmniDiT	15.9592	1.4308	0.7115	0.1728

Table 7: Quantitative comparison on **DressCode** dataset for the **model-free try-on** task. The best and the second best results are denoted as **Bold** and underline, respectively.

Method	DINO-I $\uparrow$	CLIP-I $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	KID $\downarrow$	FFA $\uparrow$
MagicClothing [7]	0.3652	0.7884	0.6137	0.4831	29.7541	22.0792	0.5872
StableGarment [61]	0.3687	0.7537	0.5636	0.5465	57.6464	37.0746	0.5941
Any2anyTryon [18]	0.3269	0.7998	0.7083	0.4004	56.2823	35.8366	0.5407
DreamFit [36]	<u>0.4087</u>	0.8253	0.7486	<u>0.2574</u>	<u>14.6571</u>	<u>6.5216</u>	<u>0.5962</u>
IMAGDressing [50]	0.4049	<u>0.8302</u>	<u>0.7713</u>	0.2848	16.1254	8.7264	0.5879
OmniDiT	0.4336	0.8533	0.7758	0.2124	9.6294	3.0329	0.6103

Table 8: Quantitative comparison on **Omni-TryOn** dataset for the **model-free try-on** task. The best and the second best results are denoted as **Bold** and underline, respectively.

Method	DINO-I $\uparrow$	CLIP-I $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	KID $\downarrow$	FFA $\uparrow$
MagicClothing [7]	0.3721	0.7667	0.4847	0.5448	49.2383	20.7358	0.5514
StableGarment [61]	0.3949	0.7551	0.4582	0.5604	61.8785	22.3345	0.5695
Any2anyTryon [18]	0.3701	0.8014	0.5409	0.5185	84.2664	53.0184	0.5539
DreamFit [36]	0.3777	0.7388	0.5205	0.4567	29.1108	6.6340	0.5539
IMAGDressing [50]	0.3900	<u>0.7930</u>	0.5361	0.5297	41.0636	15.2807	0.5615
DreamOmni2 [67]	0.3685	0.6963	0.4922	0.4769	38.7223	15.5396	0.5593
OminiControl [52]	0.3606	0.7324	0.5144	0.4844	31.4160	9.7545	0.5476
DreamO [44]	0.4092	0.7074	0.5075	<u>0.4553</u>	30.3898	8.3570	0.6028
UNO [65]	<u>0.4100</u>	0.7639	<u>0.5499</u>	0.4716	29.7925	7.6892	0.5923
OmniGen2 [64]	0.4217	0.7315	0.5090	0.4705	<u>26.1271</u>	<u>4.3034</u>	<u>0.6120</u>
OmniDiT	0.3822	0.7417	0.5658	0.3878	22.7795	3.7496	0.6180

Table 9: Quantitative comparison on **DressCode** dataset for the **try-off** task. The best and the second best results are denoted as **Bold** and underline, respectively.

Method	CLIP-I $\uparrow$	MS-SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	CLIP-FID $\downarrow$	KID $\downarrow$	DISTS $\downarrow$
TryOffDiff [57]	0.9257	0.7432	<u>0.1979</u>	11.7073	8.2309	<u>3.4271</u>	0.2480
MGT [58]	<u>0.9259</u>	<u>0.7377</u>	0.2022	<u>11.1108</u>	<u>8.0411</u>	3.6770	<u>0.2476</u>
TryOffAnyone [66]	-	-	-	-	-	-	-
Any2anyTryon [18]	0.8586	0.5572	0.3635	22.9081	10.6153	7.6255	0.3399
OmniDiT	0.9365	0.7052	0.1746	7.5019	2.3383	1.2174	0.2013

Table 10: Quantitative comparison on **Omni-TryOn** dataset for the **try-off** task. The best and the second best results are denoted as **Bold** and underline, respectively.

Method	CLIP-I $\uparrow$	MS-SSIM $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	CLIP-FID $\downarrow$	KID $\downarrow$	DISTS $\downarrow$
TryOffDiff [57]	0.8606	<u>0.5745</u>	0.3884	34.3007	9.7895	11.1588	0.3334
MGT [58]	0.8722	0.5698	0.3798	28.0170	8.4188	7.9347	0.3127
TryOffAnyone [66]	0.8472	0.4658	0.4169	46.5044	12.4310	20.9535	0.3061
Any2anyTryon [18]	0.8102	0.4728	0.4366	68.1454	16.6143	36.3953	0.3406
DreamOmni2 [67]	<u>0.9107</u>	0.4888	0.3600	<u>23.2848</u>	<u>4.3091</u>	<u>5.2890</u>	<u>0.2701</u>
OminiControl [52]	0.8853	0.4700	0.3796	25.6226	5.5035	5.9275	0.2858
DreamO [44]	0.8755	0.4825	<u>0.3577</u>	30.7476	9.6856	10.0566	0.2809
UNO [65]	0.8709	0.5052	0.3834	33.7029	11.5271	11.3796	0.2880
OmniGen2 [64]	0.8894	0.4651	0.4008	27.5556	5.9572	6.5501	0.2899
OmniDiT	0.9315	0.6225	0.2367	13.3311	2.1556	0.7115	0.2035

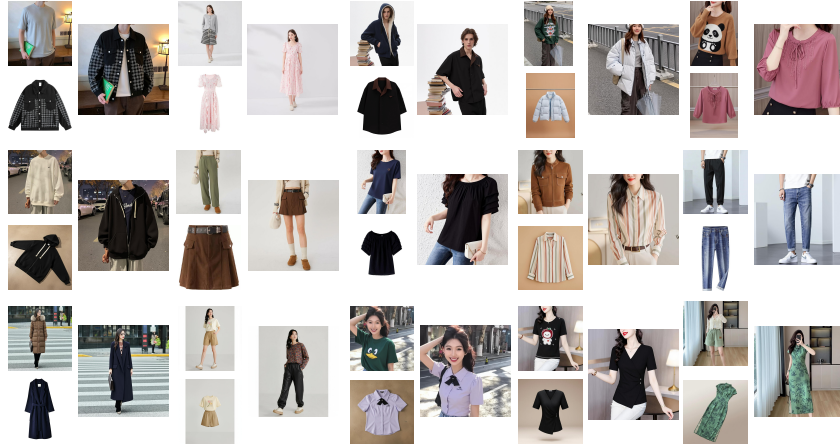


Fig. 11: Additional model-based VTOn showcases(1) in our dataset Omni-TryOn.

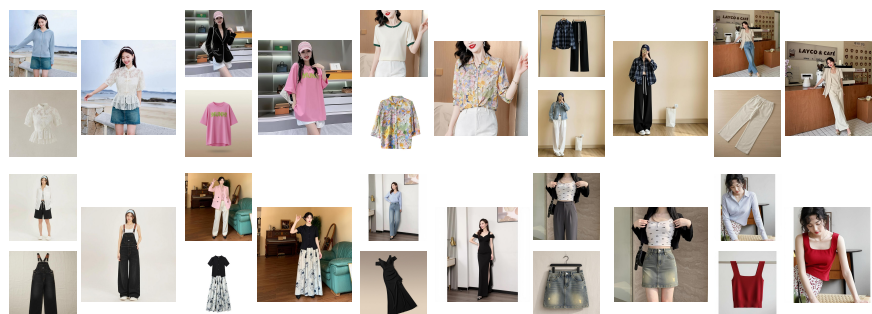


Fig. 12: Additional model-based VTON showcases(2) in our dataset Omni-TryOn.

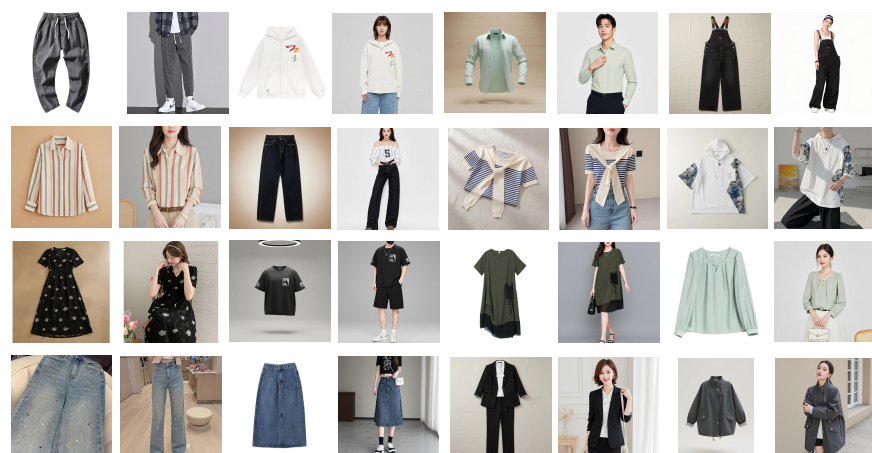


Fig. 13: Additional model-free VTON showcases in our dataset Omni-TryOn.

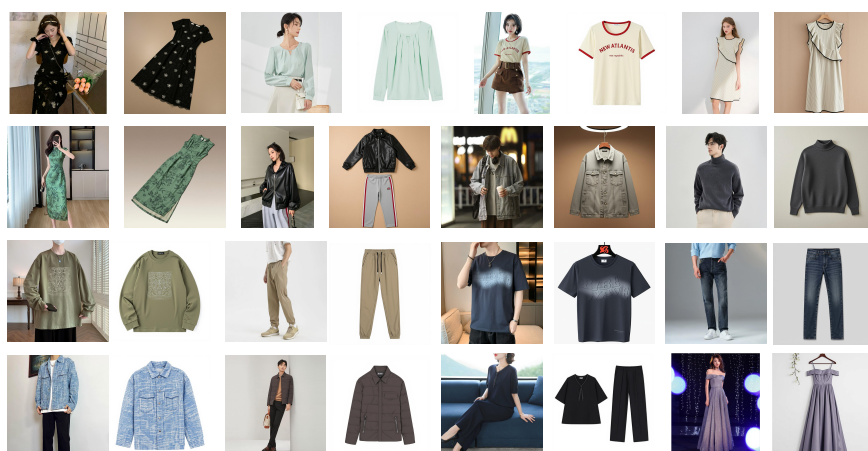


Fig. 14: Additional VTOFF showcases in our dataset Omni-TryOn.