# OmniTry: Virtual Try-On Anything without Masks

**Yutong Feng**[1,†] **Linlin Zhang**[2] **Hengyuan Cao**[2] **Yiming Chen**[1]

**Xiaoduan Feng**[1] **Jian Cao**[1] **Yuxiong Wu**[1] **Bin Wang**[1,‡]

[1]Kunbyte AI [2]Zhejiang University

{fengyutong.fyt, binwang393}@gmail.com {zhanglinlinlin, caohy}@zju.edu.cn
{chenyiming, fengxiaoduan, caojian, wuyuxiong}@k-fashionshop.com

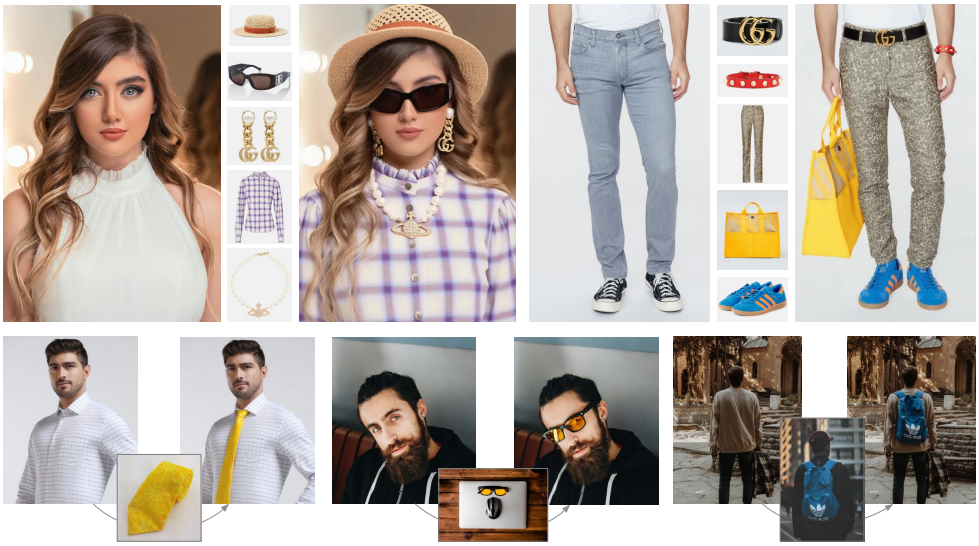


Figure 1: Try-on results of various wearable objects generated by `OmniTry`, which supports object images with white or natural backgrounds, and even try-on results as input.

## Abstract

Virtual Try-ON (VTON) is a practical and widely-applied task, for which most of existing works focus on clothes. This paper presents `OmniTry`, a unified framework that extends VTON beyond garment to encompass any wearable objects, *e.g.*, jewelries and accessories, with mask-free setting for more practical applications. When extending to various types of objects, data curation is challenging for obtaining paired images, *i.e.,* the object image and the corresponding try-on result. To tackle this problem, we propose a two-staged pipeline: For the first stage, we leverage large-scale unpaired images, *i.e.,* portraits with any wearable items, to train the model for mask-free localization. Specifically, we repurpose the inpainting model to automatically draw objects in suitable positions given an empty mask. For the second stage, the model is further fine-tuned with paired images to transfer the consistency of object appearance. We observed

[†]Project Leader. [‡]Corresponding Author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

that the model after the first stage shows quick convergence even with few paired samples. OmniTry is evaluated on a comprehensive benchmark consisting of 12 common classes of wearable objects, with both in-shop and in-the-wild images. Experimental results suggest that OmniTry shows better performance on both object localization and ID-preservation compared with existing methods. The code, model weights, and evaluation benchmark of OmniTry are available at https://omnitry.github.io/.

## 1 Introduction

The image-based virtual try-on (VTON) [19] has received tremendous attention due to its wide application in e-commerce. Given a person image and a garment image, the purpose of VTON is to transfer the garment onto the person as a preview. Thanks to the success of large-scale image generative models [50, 47, 14, 32] with their photorealistic aesthetics, recent efforts [60, 28, 9, 10, 66] have achieved satisfying performance on both generation quality and garment identity preservation.

Despite the advancement of VTON, existing methods mainly concentrate on clothing try-on. Though some works have explored the extension to non-clothing, such as shoes [11] and ornaments [42], there still lacks a unified framework in the literature, supporting any types of wearable objects. Furthermore, most methods require the indication of wearing area on person (*e.g.,* masks or bounding boxes), or use automatic human-body parsers [62] to identify the area. When extending to anything try-on, it would be impractical to expect users to draw the targeting area, as the interaction between the model and various objects can be more considerably more complex. It is also challenging to leverage existing parsers to detect appropriate try-on areas for diverse objects. Thus, we follow the mask-free setting [24, 16, 66] for the model to automatically localize the area with natural composition.

When confronting anything try-on, one key challenge is the data collection. Generally, the training of VTON requires large-scale *paired* images, consisting of a single-shot of the garment, and a corresponding person try-on result. Most datasets are curated from e-commerce websites, with at least thousands of samples, *e.g.,* VITON-HD [8] and DressCode [43]. While for many common types of wearable objects, such as hats and ties, there is no abundant quantity of paired data, but only the product pictures. This limitation makes it necessary to develop an efficient training framework.

In this paper, we present OmniTry, targeting mask-free virtual try-on for any wearable object. OmniTry reduces the heavy reliance on paired training samples, leveraging large-scale *unpaired* images for prior learning. The unpaired images refer to the image containing a person with any wearable objects, which can be easily obtained from existing database. The training of OmniTry can be separated into two stages: (i) The first stage is completely conducted on unpaired data. We use multi-modal large language models (MLLMs) [1] to list all wearable items with descriptions. Each item is detected and erased from the image, forming a training pair. Then an image generative model is trained to re-paint the item, prompted by the corresponding text description. After stage one, the model is expected to know how to transfer various objects onto the person in proper position, size and orientation. (ii) For the second stage, we further leverage high-quality paired data to fine-tune the model. Object image is introduced into the context, modulating the model to preserve the consistency of object appearance. Building upon the model from stage one, we observe that ID-consistency is quickly adapted even fine-tuned with few samples. To summarize, the two stages in OmniTry contributes the ability of mask-free localization and ID-preservation, respectively.

Regarding the model design, we leverage the diffusion transformer as backbone, and compare two variants, *i.e.,* text-to-image and inpainting model. Experimental results show that the inpainting model can be rapidly repurposed as a mask-free generative model, by simply setting the mask input with all-zero values. Image tokens from different images are concatenated in the sequence dimension, and processed with full-attention mechanism for consistency learning [53, 67, 7, 23]. We employ efficient adapter tuning techniques for transferring the model to this task. More specifically, we implement two distinct adapters that handle the tokens from person and object images, individually.

The erasure of wearable object is observed with critical impact. A naive solution is to call object-removal models [52, 71, 27] to fill the area of objects. However, we notice that while the processed area appears visually normal, it contains imperceptible artifacts. Thus, the model learn undesirable shortcuts by identifying these traces, resulting in poor generalization to natural images. To tackle this problem, we propose *traceless erasing* to eliminate the artifacts. We conduct image-to-image [41] to

subtly re-paint the entire image after erasure. Subsequently, the original try-on image is blended with the re-painted image, ensuring the non-object area remains unchanged. Traceless erasing disrupts the erasure boundaries, thereby compelling the model to learn genuine try-on capability.

We construct a comprehensive evaluation benchmark covering 12 common types of wearable objects, divided into clothes, shoes, jewelries and accessories. To fully investigate the model robustness, the objects are set on white and natural backgrounds, or try-on images, referring to Fig. 1. Metrics are designed to evaluate the object consistency, person preservation and wearing position. Experimental results indicate that `OmniTry` outperforms existing methods, and achieves efficient few-shot training.

## 2 Related Works

**Controllable Image Generation.** The breakthrough of diffusion model [20] has driven extensive research on controllable image generation. ControlNet [64] and related pioneering works [45, 48, 68] explore precise control with diverse conditions. IP-Adapter [63] and related studies [15, 22, 31, 34, 39] investigate online concept control to achieve subject customized generation. Recent developments in DiT [46] have further propelled generalized image generation and editing. In-context LoRA [23] enables diverse thematic generation with image concatenation. OminiControl [53] introduces task-agnostic condition control with minimal model modification. OmniGen [58] unifies multi-task processing via large vision-language models. UniReal [7] achieves unified image editing via full-attention and video data prior. VisualCloze [35] enhances visual in-context learning for cross-domain generalization. For localized image customization, Anydoor [6] pioneers to transfer subject into specified region. MimicBrush [5] extends to local components transferring with imitative editing. ACE++ [40] establishes a unified paradigm for generation and editing tasks.

**Image-based Virtual Try-On (VTON)** has emerged as a critical task attracting tremendous efforts. VITON [19] introduces Thin Plate Spline transformations [2] for multi-stage garment processing. CP-VTON [55] formalizes explicit geometric warping and texture synthesis stages. GP-VTON [59] combines local flow estimation with global parsing to improve detail preservation. These warping-based approaches, however, face persistent challenges in cross-sample alignment and generalization. This motivates the adoption of diffusion models [20], including TryOnDiffusion's parallel U-Net [70], LADIVTON's garment tokenization [44], and DCI-VTON's hybrid warping-diffusion framework [17]. OOTDiffusion [60] and FitDiT [25] enhance detail fidelity through specialized attention mechanisms. Though with advanced results, most of them remain constrained by intensive preprocessing requirements (*e.g.,* wearing masks and pose estimation). Boow-VTON [66] creates a mask-free approach through in-the-wild data augmentation. Any2AnyTryon [18] pioneers fully mask-free implementations, eliminating dependency on masks or poses.

## 3 Method

### 3.1 Preliminary

**Diffusion Transformer (DiT).** `OmniTry` is developed on DiT [46], a scalable transformer architecture for diffusion-based generation. The image is encoded into latent space through an autoencoder [29], and patchified into tokens [13]. Diffusion process [20] is conducted on tokens with a transformer consuming the noisy tokens and predicts for denoising. Recent advancement in DiT, *i.e.,* recified flow matching [37] and rotary position embedding (RoPE) [51], are also involved in this paper.

**Virtual Try-On (VTON).** Given a person image $\mathcal{I}_P$ and a wearable object image $\mathcal{I}_O$, the try-on result image is noted as $\mathcal{I}_T$. Suppose the segmentation mask of the object in $\mathcal{I}_T$ is $\mathcal{M}$, then the target of VTON is three-fold: (i) the consistency between objects in original and try-on images, *i.e.* $\min \text{similarity}(\mathcal{I}_T\mathcal{M}, \mathcal{I}_O)$, (ii) the preservation of non-wearing areas, *i.e.,* $\mathcal{I}_T(1-\mathcal{M}) = \mathcal{I}_P(1-\mathcal{M})$, (iii) the object is properly located on person, evaluated through the quality of $\mathcal{I}_T$.

### 3.2 Stage-1: Mask-Free Localization

As illustrated in Fig. 2, the training of `OmniTry` consists of two stages, corresponding to the abilities of localization and ID-preservation, respectively. In the first stage, the objective of training can be regarded as "garment-free VTON", in contrast to the "model-free VTON" in the literature [18]. Given

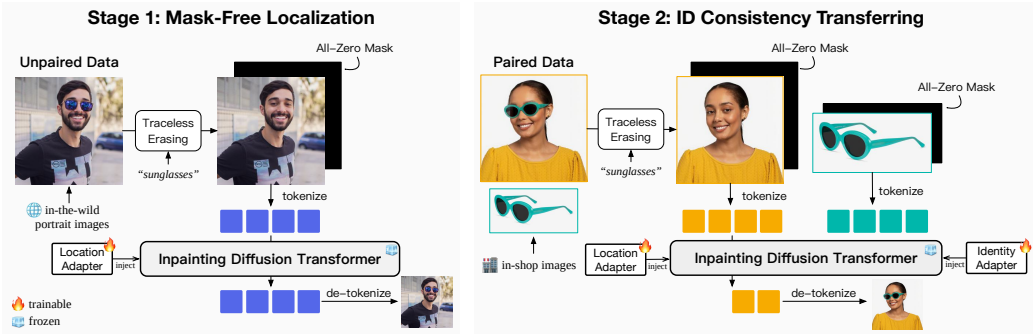



Figure 2: **The two-staged training pipeline of `OmniTry`.** The first stage is built on in-the-wild portrait images to add wearable object onto the person in mask-free manner. The second stage introduces in-shop paired images, and targets to control the consistency of object appearance.

the person image $\mathcal{I}_P$ and the object description, the model aims to edit $\mathcal{I}_P$ by adding the object as described. The type and detailed appearance of object are only prompted by input text. Control signal indicating the wearing area, *e.g.,* bounding boxes, masks or selecting point, is not introduced here. Such an objective enforces the model to concentrate on *where* to paint the object, and *how* to blend it harmoniously with the person image. The training of stage one can be easily supervised by a portrait image database, for which we introduce how to construct the training samples in the next paragraph.

**Unpaired Data Pre-process.** We refer single portrait image as *unpaired* image with only try-on result $\mathcal{I}_T$, in contrast to the paired images $(\mathcal{I}_T, \mathcal{I}_O)$ in next stage. We start by curating a large-scale dataset containing any human-related images. The dataset is filtered by a classifier for images with a person wearing at least one object. Following that, we leverage a MLLM, Qwen-VL 2.5 [1], to list all potential wearable objects in each image. The output includes both the type of object and its appearance description. We also prompt MLLM to add an interaction description, *e.g.,* "wearing sunglasses" and "holding sunglasses in hand", to distinguish various cases. To erase each object for training, we use GroundingDINO [36] and SAM [30] to obtain the object mask, and remove the object with an inpainting-based erasing model. Specifically, we fine-tune an internal erasing model based on Flux.1 Fill [32]. Though without erasing capability, it is observed to quickly adapt to this task with a few training samples. To summarize, the pre-precessing pipeline outputs a set of triples, including the original image as $\mathcal{I}_T$, the object-erased image as $\mathcal{I}_P$, and the object textual description.

**Model Architecture: Text-to-Image *v.s.* Inpainting Model.** There are two candidate variants of model to implement the mask-free try-on task, *i.e.,* the text-to-image (T2I) model, and the mask-based inpainting model. Generally, mask-based VTON models [28, 10] leverage the fill-in capacity of inpainting model, while mask-free methods [66, 18] adapt the T2I model, by injecting subject features into the backbone. Following the recent success in controllable image generation [53, 23, 67], a straightforward solution with T2I model is to concatenate the person image tokens into the sequence of noisy tokens, then processed with the full-attention mechanism in DiT. This strategy effectively transfers the person appearance into the target image, while also doubles the computation cost.

In contrast, `OmniTry` explores to repurpose the inpainting model for mask-free generation. The inpainting model is generally finetuned from the T2I model via extending the input channels. Suppose the noisy latent as $X$, the input image as $I_c$, and the inpainting mask as $M$. Then the extended input is concat$(X; I_c(1-M); M)$, where concat$(\cdot)$ denotes channel-wise concatenation. For repurposing the model, we simply set $M = \mathbf{0}$, thus the input turns to be concat$(X; I_c; \mathbf{0})$. At the initialization, the zero mask leads the output image directly repeating the input. Therefore, compared with T2I-based solution, the model effortlessly learns to copy the person condition, thus attentively focusing on locating the modification area. We inject a location adapter (implemented as LoRA [21]) for finetuning. In practice, the model converges rapidly to adapt the mask-free generative manner.

**Traceless Erasing.** Early experimental results suggest that the model learn unexpected shortcut. We visualize the training monitoring result in Fig. 3 (a), where we evaluate the model on erased training samples. It is shown that the output image almost perfectly recovers the position and shape of the object in ground-truth image, which indicates information leakage. We attribute the problem to the erasing model that leaves invisible traces in the filling area [56, 69]. The model tends to figure out these abnormal area for editing, instead of predicting the reasonable position. When applying to real-world images, the model frequently fails to locate the try-on area, and directly repeat the input.

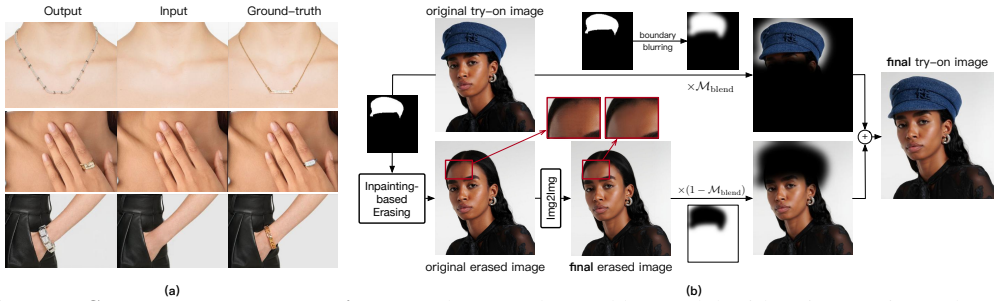



Figure 3: **Study on traceless erasing.** (a) Shortcuts learned by model with naive erasing, where the model recovers the same shape and position as the ground-truth. (b) The pipeline of traceless erasing, where image-to-image model is introduced to disturb the traces (indicated in the red boxes).

To address this problem, we propose a traceless erasing strategy, as shown in Fig. 3 (b). After erasing the object with inpainting model, we apply with an image-to-image (I2I) translation [41] to subtly re-paint the image. We first add noise to the erased image $\hat{\mathcal{I}}_P$, referring a specific timestep $t \in [0, 1]$ in diffusion schedule ($t = 0.2$ in this paper), *i.e.,* $z = enc(\hat{\mathcal{I}}_P) \times (1-t) + \epsilon \times t$, where $enc(\cdot)$ denotes the VAE encoder and $\epsilon$ is a standard Gaussian noise. Then a T2I model denoises $z$ into normal image with partial diffusion process from $t$ to 0. In this manner, the artificial effects in inpainting area are confused with the whole re-painted image, thus avoid information leakage. Since the I2I process modifies the detail of person image, the original try-on image should be correspondingly adjusted. To achieve smooth transition in the object boundary, we modulate the original mask $\mathcal{M}$ into a blending mask $\mathcal{M}_{blend}$ by blurring the boundary area for gradual blending effect. The final try-on image is:

$$\mathcal{I}_T^{\text{blend}} = \mathcal{I}_T \times \mathcal{M}_{\text{blend}} + \text{img2img}(\hat{\mathcal{I}}_P) \times (1 - \mathcal{M}_{\text{blend}}) \tag{1}$$

### 3.3 Stage-2: ID Consistency Preservation

The second stage of `OmniTry` inherits the location adapter from stage one, and steps further to control the consistency of object appearance. Referring to Fig. 2, in-shop image pairs are leveraged containing try-on image $\mathcal{I}_T$ and object image $\mathcal{I}_O$. We pre-process the data with traceless erasing, and gather a list of triple $(\mathcal{I}_T, \mathcal{I}_P, \mathcal{I}_O)$ for training. Considering the lack of enough samples, the objective is to conduct efficient training with minimal adjustment to the model architecture in stage one.

**Masked Full-Attention.** Following the recent full-attention customization researches [53, 23], we directly append the object image tokens into the existing sequence in DiT, and shift their position embedding in the width dimension. Under this settings, OminiControl-2 [54] and EasyControl [67] also explore to block some information flow in attention. In detail, the attention mask is set to zero where the condition tokens serve as query and the generated tokens as key. Such an attention mask improves the inference efficiency, but leads to performance decrease to a certain extent.

The main difference between the above works and `OmniTry` is that the condition image is also concatenated with noisy latents and all-zero mask, for adaption to inpainting model. To cope with such variance, we design two strategies in training: (i) We compute diffusion loss on object image with itself as supervision, *i.e.,* directly copying the input, which is aligned with the zero-mask input. (ii) We block all the data flow from the generating try-on image to object image, thus avoid the detailed object appearance to be interrupted. In practice, we find it helpful to better preserve object identity with the above masked full-attention.

**Two-Stream Adapters.** To fully preserve the ability of mask-free localization, we maintain the forward process of person image tokens exactly consistent with the first stage. Then an identity adapter is initialized for the newly introduced object image tokens. The two adapters, in same architecture, serve for a two-stream computation process, *i.e.,* we switch different adapters by identifying tokens from different image sources. The inference is similar to the multi-modality DiT [14] coping with vision and language information separately.


Figure 4: **Qualitative comparison** among OmniTry and existing methods on multiple objects.

### 3.4 Evaluation Benchmark

As the first work exploring unified virtual try-on task, we establish a comprehensive benchmark, dubbed *OmniTry-Bench*, for better evaluation and comparison with existing works.

**Benchmark Collection.** We gather evaluation samples within 12 common types of wearable objects, which can be summarized into 4 major classes: (i) *clothes* consisting of top, bottom and full-body garments, (ii) *shoes* in common styles, (iii) *jewelries*, including bracelets, earrings, necklaces and rings, (iv) *accessories*, including bags, belts, hats, glasses, sunglasses and ties. We consider detailed sub-types if necessary, such as the backpack, shoulder and tote bags. For each sub-type, we collect 15 paired test images for man and woman, separately. The object images are assigned in white background, natural background, and try-on setting, with 5 pairs for each. The person images are also set in white and natural backgrounds. Such settings ensure to fully evaluate the robustness of model. Overall, the evaluation benchmark contains 360 pairs of images.

**Evaluation Metrics.** As discussed in Sec. 3.1, the objectives of try-on can be divided into three aspects. Since there is no ground-truth result in mask-free setting, we redesign the metrics as follows:

*Object Consistency*: We crop the objects from the try-on and object images via masking, and compute the visual similarity using DINO [3] and CLIP [49], with metrics noted as M-DINO and M-CLIP-I.

*Person Preservation*: In contrast, we crop out the person from try-on and person images, and compute spatial-aligned similarity between them, *i.e.,* LPIPS [65] and SSIM [57].

Table 1: **Evaluation results on OmniTry-Bench**, which is separated into two groups: results on the whole set and the clothes subset, for fair comparison with methods only optimized on clothes data.

| | | **Object Consistency** | | **Person Presevation** | | **Object Localization** | |
|---|---|---|---|---|---|---|---|
| method | mask | M-DINO ↑ | M-CLIP-I ↑ | LPIPS ↓ | SSIM↑ | G-Acc. ↑ | CLIP-T ↑ |
| *on the whole set* | | | | | | | |
| Paint-by-Example [61] | ✓ | 0.4565 | 0.7727 | 0.3903 | 0.8033 | 0.9861 | 0.2804 |
| MimicBrush [5] | ✓ | 0.4693 | 0.7253 | 0.3033 | 0.8575 | 0.9250 | 0.2781 |
| ACE++ [40] | ✓ | 0.4565 | 0.7474 | 0.4561 | 0.7519 | 0.9667 | 0.2791 |
| OneDiffusion [33] | ✗ | 0.4731 | 0.7749 | 0.7001 | 0.5831 | **0.9972** | 0.2309 |
| VisualCloze [35] | ✗ | 0.5292 | 0.7782 | 0.4471 | 0.6190 | 0.9639 | 0.2524 |
| OmniGen [58] | ✗ | 0.5435 | 0.7869 | 0.6703 | 0.5965 | 0.9944 | 0.2535 |
| `OmniTry` (**Ours**) | ✗ | **0.6160** | **0.8327** | **0.0542** | **0.9333** | **0.9972** | **0.2831** |
| *on the clothes subset* | | | | | | | |
| Magic Clothing [4] | ✓ | 0.5665 | 0.7634 | 0.2761 | 0.8786 | 1.0 | 0.2700 |
| CatVTON [10] | ✓ | 0.5744 | 0.7906 | 0.2084 | 0.8828 | 1.0 | 0.2797 |
| OOTDiffusion [60] | ✓ | 0.5961 | 0.8016 | 0.2178 | 0.8865 | 1.0 | 0.2761 |
| FitDiT [25] | ✓ | 0.6733 | 0.8340 | 0.1618 | 0.9027 | 1.0 | 0.2831 |
| Any2AnyTryon [18] | ✗ | 0.6747 | 0.8537 | 0.2089 | 0.8969 | 1.0 | **0.2832** |
| `OmniTry` (**Ours**) | ✗ | **0.6995** | **0.8560** | **0.1021** | **0.9105** | **1.0** | 0.2799 |

*Object Localization*: (i) Counting the success rate whether a visual grounding model [36] detects the object, denoted as G-Accuracy. (ii) Computing the image-text similarity, noted as CLIP-I, between try-on image and a text describing the person trying on the object (generated by MLLM [1]).

# 4 Experiment

## 4.1 Experimental Setup

**Training Data.** For the first stage, we gather a diverse dataset containing both in-the-wild portrait images and in-shop model shots. Considering each image could contain multiple wearable objects, the total amount of training pairs is $188,694$. For the second stage, we collect paired samples following the 12 basic types in our benchmark. The whole dataset contains $51,195$ pairs, which shows class-unbalanced distribution ($14,861$ pairs for clothes and $295$ for ties). During training, each pair is equipped with a brief text description, such as "trying on sunglasses", to help distinguishing different classes. We note that the clothes and shoes are not erased but replaced with another one. Thus, we exchange the prefix as "replacing" for their prompts.

**Implementation Details.** We train the first stage with batch-size of $32$ for $50$K steps, and the second stage with batch-size of $16$ for $25$K steps. All the experiments are conducted on $4$ NVIDIA H800 GPUs. The location and identity adapters are implemented as LoRA [21] with rank $16$. We employ the AdamW [38] optimizer with learning rate of $1^{-4}$ and weight decay of $0.01$. All the images are resized to a maximum of $1$ million pixels while preserving their original aspect ratios to training.

**Compared Methods.** We primarily compare with methods in two basic paradigms:

*Image-based Virtual Try-On*: Most VTON methods focus exclusively on garments. We compare on the clothes subset with representative works, including CatVTON [10], OOTDiffusion [60], Magic Clothing [4], FitDiT [25], and Any2AnyTryon [18] (the only open-sourced mask-free model).

*General Customized Image Generation*: Recent works explore to unify customization-related tasks into a single model, *e.g.,* transferring the whole subject or local components, in mask-based or mask-free manners. We compare with notable implementations, including Paint-by-Example [61], MimicBrush [5], ACE++ [40], OneDiffusion [33], OmniGen [58] and VisualCloze [35].

To cope with the methods requiring masks of editing areas, we manually draw the masks in person image regarding the type of objects. Thus, the results of these methods are listed for reference, instead of direct comparison with the remaining mask-free methods.

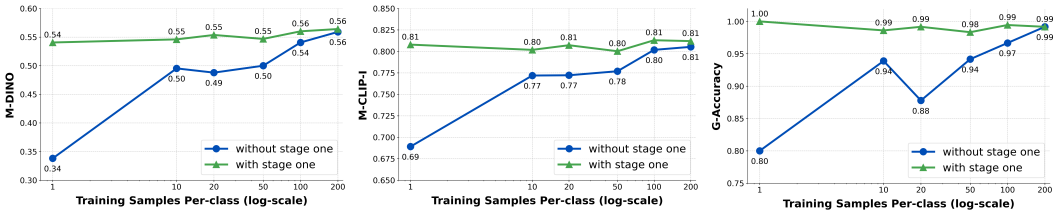



Figure 5: Ablation study on the two-staged training framework in **few-shot settings**. We show the evaluation metrics given varying amounts of paired training samples.

### 4.2 Results on Unified Virtual Try-on

**Qualitative Results.** We visualize the try-on examples generate by representative compared methods in Fig. 4. For the general customization methods, mask-based works only modify the given areas, but show unstable object identity transferring. While for the mask-free works, the results tend to be a free combination of the input person and object. Though with better consistency, they fail to precisely preserve the person image. `OmniTry` achieves accurate object consistency, in the meanwhile only edits the proper try-on areas of person image in mask-free manner. On the clothes subset evaluation, we observe that the existing VTON methods show unnatural output when evaluated on in-the-wild data. `OmniTry` is empowered by the compounded training on both in-the-wild and in-shop data, and shows more generalized ability of adapting various styles of garments.

**Quantitative Results.** Tab. 1 incorporates the evaluation results on the proposed OmniTry-Bench, conducted on the whole benchmark and the clothes subset, respectively. `OmniTry` outperforms existing methods on both sets. For the mask-based customization methods, though the input mask helps to localize the editing area, they sometimes fail to transfer the complete appearance of objects, resulting in lower consistency metrics. For the generalized customization methods in mask-free manner, they achieve better subject-ID preservation, but suffers to maintain the person image, thus show worse LPIPS and SSIM. Such quantitative results are consistent with the visualized comparison results. When evaluated on clothes subset, though `OmniTry` is not specifically optimized on clothes dataset, it still shows advancing performance compared with state-of-the-art works in mask-based and mask-free settings. We note that the mask-free try-on could not be evaluated on previous benchmarks (*e.g.,* VITON-HD [8]) for the missing of person images.

### 4.3 Ablation Study

**On the Training Strategy.** We study one of the key designs in `OmniTry`, *i.e.,* the two-staged training framework. The first stage is intended to leverage large-scale unpaired data, and boost the training efficiency in the second stage. To demonstrate this, we evaluate models initialized by the first stage and from scratch, respectively. For the comparison of efficiency, the models are fine-tuned in few-shot settings, ranging from 1 to 200 training samples per class. The results with representative metrics are illustrated in Fig. 5. For metrics related to person preservation (LPIPS and SSIM), we note that they could be higher when the model fails and directly repeats the input, thus not included.

It is observed that model from scratch shows increasing performance with more training samples per class. While for model initialized from the first stage, it already achieves satisfying performance even with only one example for training. The results demonstrate that the first stage training significantly boosts the efficiency for fine-tuning, and is especially friendly to uncommon types of objects. It is noted that though the few-shot tuning achieves good performance, we still fine-tune it with all available paired data to further increase the stability of model, referring to the results in Tab. 1

Table 2: Ablation study on the model architecture and erasing strategies of `OmniTry`.

| method | M-DINO ↑ | M-CLIP-I ↑ | LPIPS ↓ | CLIP-T ↑ |
|---|---|---|---|---|
| *on the model architecutre (the whole subset)* | | | | |
| Full Method | **0.5991** | **0.8272** | 0.0557 | 0.2830 |
| - txt2img model | 0.5005 | 0.7727 | 0.0676 | 0.2767 |
| - w.o. object loss | 0.5851 | 0.8222 | 0.0420 | 0.2824 |
| - full attention | 0.5752 | 0.8130 | **0.0384** | **0.2832** |
| - one-stream adapter | 0.5840 | 0.8186 | 0.0502 | 0.2802 |
| *on the erasing strategy (the jewelry subset)* | | | | |
| naive erasing | 0.4964 | 0.7554 | 0.0413 | 0.2727 |
| traceless erasing | **0.5389** | **0.7782** | **0.0288** | **0.2732** |

Figure 6: Try-on results of OmniTry fine-tuned on uncommon classes of wearable or holdable objects.

**On the Model Architecture.** We then conduct ablation study on all the explored design of model architecture in OmniTry. The results are shown in Tab. 2, where the "full method" indicates the final solution. (i) We start with the comparison using text-to-image and inpainting model as backbone. Results show that the inpainting backbone performs better on all metrics which is consistent with our assumption that inpainting model takes no efforts to preserve the original image and converges faster. (ii) For the additional loss computation on object image, we observe that removing the loss decreases the model performance to a certain extent. (iii) For the attention mechanism, full attention additionally introduces flow from person to object image, thus the object consistency metrics decrease correspondingly. (iv) We also investigate to use a single adapter for this task, *i.e.* applying the adapter from the first stage to all image tokens. The one-stream framework also decreases the model performance, since it plays different roles in the inference of person and object images.

**On the Traceless Erasing.** To verify the effectiveness of traceless erasing, we conduct ablation study on the jewelry subset with naive and traceless erasing. Results in Tab. 2 suggest that removing the traceless erasing leads to dramatic decrease in all metrics. Therefore, we adopt traceless erasing as a fundamental pre-processing strategy in OmniTry.

### 4.4 Extension to Uncommon Classes

We evaluate OmniTry on 12 common types of objects in the main experiments. To further demonstrate the efficiency of OmniTry, we extend it to some uncommon types, for which the paired training samples are limited to be obtained. The experiment is conducted on types including gloves, earphones, watches, hairbands, books and electronic products, with roughly 20 samples per class. It is noted that some types like books are actually in broader definition of try-on, *i.e.,* holdable items.

The visualization results are shown in Fig. 6. Thanks to the generalized training of the first stage, though with few paired samples, OmniTry succeeds in transferring these relatively uncommon objects onto the correct position. The results encourage broader extension of OmniTry into more application scenarios, without preparing a large amount of paired images.

## 5 Limitations

In this section, we discuss the limitations of OmniTry observed in practice. As the first work exploring unified VTON, OmniTry is still restricted by the object types in training dataset. For the efficient tuning in stage-2, it could be challenging to extend to uncommon objects not involved in the unpaired dataset in stage-1. Larger pre-training dataset is expected to further boost the generalization ability. For the mainly-focused 12 common types, experimental results show that OmniTry could also fail to transfer the object consistency or output poor appearance in some cases, especially for the objects with larger transformation, *e.g.,* bags. The above limitations encourage future works to build upon OmniTry and develop more advanced models towards unified try-on task.

## 6 Conclusion

This paper presents OmniTry, a unified mask-free framework extending the existing garment try-on into any wearable objects. To tackle the problem of lacking abundant paired samples, *i.e.,* object and the try-on image, for many types of objects, we propose a two-staged training pipeline in OmniTry. During the first stage, large-scale unpaired images are leveraged to supervise the model for mask-free object localization. While the second stage tames the model to maintain the object consistency. We elaborate the design of OmniTry, including a traceless erasing for avoiding shortcut learning, an inpainting-based re-purposing strategy for mask-free generation, and a masked full-attention for identity transferring. A new benchmark targeting unified try-on is introduced, and demonstrates the effectiveness of OmniTry compared with existing methods. Extensive experiments also verify that OmniTry achieves efficient learning even with few paired images for training.

## References


[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[2] Fred L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(6), 1989.

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[4] Weifeng Chen, Tao Gu, Yuhao Xu, and Arlene Chen. Magic clothing: Controllable garment-driven image synthesis. In *Proceedings of the 32nd ACM International Conference on Multimedia, MM 2024, Melbourne, VIC, Australia, 28 October 2024 - 1 November 2024*. ACM, 2024.

[5] Xi Chen, Yutong Feng, Mengting Chen, Yiyang Wang, Shilong Zhang, Yu Liu, Yujun Shen, and Hengshuang Zhao. Zero-shot image editing with reference imitation. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024.

[6] Xi Chen, Lianghua Huang, Yu Liu, Yujun Shen, Deli Zhao, and Hengshuang Zhao. Anydoor: Zero-shot object-level image customization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6593–6602, 2024.

[7] Xi Chen, Zhifei Zhang, He Zhang, Yuqian Zhou, Soo Ye Kim, Qing Liu, Yijun Li, Jianming Zhang, Nanxuan Zhao, Yilin Wang, Hui Ding, Zhe Lin, and Hengshuang Zhao. Unireal: Universal image generation and editing via learning real-world dynamics. *CoRR*, 2024.

[8] Seunghwan Choi, Sunghyun Park, Minsoo Lee, and Jaegul Choo. Viton-hd: High-resolution virtual try-on via misalignment-aware normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14131–14140, 2021.

[9] Yisol Choi, Sangkyung Kwak, Kyungmin Lee, Hyungwon Choi, and Jinwoo Shin. Improving diffusion models for virtual try-on. *arXiv e-prints*, pages arXiv–2403, 2024.

[10] Zheng Chong, Xiao Dong, Haoxiang Li, Shiyue Zhang, Wenqing Zhang, Xujie Zhang, Hanqing Zhao, Dongmei Jiang, and Xiaodan Liang. Catvton: Concatenation is all you need for virtual try-on with diffusion models. *arXiv preprint arXiv:2407.15886*, 2024.

[11] Chao-Te Chou, Cheng-Han Lee, Kaipeng Zhang, Hu-Cheng Lee, and Winston H Hsu. Pivtons: Pose invariant virtual try-on shoe with conditional image completion. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part VI 14*, pages 654–668. Springer, 2019.

[12] Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations (ICLR)*, 2024.

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[14] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

[15] 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, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*, 2023.

[16] Yuying Ge, Yibing Song, Ruimao Zhang, Chongjian Ge, Wei Liu, and Ping Luo. Parser-free virtual try-on via distilling appearance flows. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8485–8493, 2021.

[17] Junhong Gou, Siyu Sun, Jianfu Zhang, Jianlou Si, Chen Qian, and Liqing Zhang. 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, MM 2023, Ottawa, ON, Canada, 29 October 2023- 3 November 2023*, 2023.

[18] Hailong Guo, Bohan Zeng, Yiren Song, Wentao Zhang, Chuang Zhang, and Jiaming Liu. Any2anytryon: Leveraging adaptive position embeddings for versatile virtual clothing tasks. *CoRR*, 2025.

[19] Xintong Han, Zuxuan Wu, Zhe Wu, Ruichi Yu, and Larry S Davis. Viton: An image-based virtual try-on network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7543–7552, 2018.

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

[21] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

[22] Miao Hua, Jiawei Liu, Fei Ding, Wei Liu, Jie Wu, and Qian He. Dreamtuner: Single image is enough for subject-driven generation. *CoRR*, 2023.

[23] Lianghua Huang, Wei Wang, Zhi-Fan Wu, Yupeng Shi, Huanzhang Dou, Chen Liang, Yutong Feng, Yu Liu, and Jingren Zhou. In-context lora for diffusion transformers. *arXiv preprint arXiv:2410.23775*, 2024.

[24] Thibaut Issenhuth, Jérémie Mary, and Clément Calauzenes. Do not mask what you do not need to mask: a parser-free virtual try-on. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, pages 619–635. Springer, 2020.

[25] Boyuan Jiang, Xiaobin Hu, Donghao Luo, Qingdong He, Chengming Xu, Jinlong Peng, Jiangning Zhang, Chengjie Wang, Yunsheng Wu, and Yanwei Fu. Fitdit: Advancing the authentic garment details for high-fidelity virtual try-on. *CoRR*, 2024.

[26] Houcheng Jiang, Junfeng Fang, Ningyu Zhang, Guojun Ma, Mingyang Wan, Xiang Wang, Xiangnan He, and Tat seng Chua. Anyedit: Edit any knowledge encoded in language models, 2025. URL `https://arxiv.org/abs/2502.05628`.

[27] Longtao Jiang, Zhendong Wang, Jianmin Bao, Wengang Zhou, Dongdong Chen, Lei Shi, Dong Chen, and Houqiang Li. Smarteraser: Remove anything from images using masked-region guidance. *arXiv preprint arXiv:2501.08279*, 2025.

[28] Jeongho Kim, Guojung Gu, Minho Park, Sunghyun Park, and Jaegul Choo. 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*, pages 8176–8185, 2024.

[29] Diederik P Kingma, Max Welling, et al. Auto-encoding variational bayes, 2013.

[30] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.

[31] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, 2023.

[32] Black Forest Labs. Flux. `https://github.com/black-forest-labs/flux`, 2024.

[33] Duong H. Le, Tuan Pham, Sangho Lee, Christopher Clark, Aniruddha Kembhavi, Stephan Mandt, Ranjay Krishna, and Jiasen Lu. One diffusion to generate them all. *CoRR*, 2024.

[34] Dongxu Li, Junnan Li, and Steven C. H. Hoi. Blip-diffusion: Pre-trained subject representation for controllable text-to-image generation and editing. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[35] Zhong-Yu Li, Ruoyi Du, Juncheng Yan, Le Zhuo, Zhen Li, Peng Gao, Zhanyu Ma, and Ming-Ming Cheng. Visualcloze: A universal image generation framework via visual in-context learning, 2025.

[36] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pages 38–55. Springer, 2024.

[37] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

[38] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[39] Jian Ma, Junhao Liang, Chen Chen, and Haonan Lu. Subject-diffusion: Open domain personalized text-to-image generation without test-time fine-tuning. In *ACM SIGGRAPH 2024 Conference Papers, SIGGRAPH 2024, Denver, CO, USA, 27 July 2024- 1 August 2024*, 2024.

[40] Chaojie Mao, Jingfeng Zhang, Yulin Pan, Zeyinzi Jiang, Zhen Han, Yu Liu, and Jingren Zhou. Ace++: Instruction-based image creation and editing via context-aware content filling. *CoRR*, 2025.

[41] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

[42] Yingmao Miao, Zhanpeng Huang, Rui Han, Zibin Wang, Chenhao Lin, and Chao Shen. Shining yourself: High-fidelity ornaments virtual try-on with diffusion model. *arXiv preprint arXiv:2503.16065*, 2025.

[43] Davide Morelli, Matteo Fincato, Marcella Cornia, Federico Landi, Fabio Cesari, and Rita Cucchiara. Dress code: High-resolution multi-category virtual try-on. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2231–2235, 2022.

[44] Davide Morelli, Alberto Baldrati, Giuseppe Cartella, Marcella Cornia, Marco Bertini, and Rita Cucchiara. Ladi-vton: Latent diffusion textual-inversion enhanced virtual try-on. In *Proceedings of the 31st ACM International Conference on Multimedia, MM 2023, Ottawa, ON, Canada, 29 October 2023- 3 November 2023*, 2023.

[45] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, 2024.

[46] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[47] 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.

[48] Can Qin, Shu Zhang, Ning Yu, Yihao Feng, Xinyi Yang, Yingbo Zhou, Huan Wang, Juan Carlos Niebles, Caiming Xiong, Silvio Savarese, Stefano Ermon, Yun Fu, and Ran Xu. Unicontrol: A unified diffusion model for controllable visual generation in the wild. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[49] 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.

[50] 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.

[51] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[52] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2149–2159, 2022.

[53] Zhenxiong Tan, Songhua Liu, Xingyi Yang, Qiaochu Xue, and Xinchao Wang. Ominicontrol: Minimal and universal control for diffusion transformer. *arXiv preprint arXiv:2411.15098*, 2024.

[54] Zhenxiong Tan, Qiaochu Xue, Xingyi Yang, Songhua Liu, and Xinchao Wang. Ominicontrol2: Efficient conditioning for diffusion transformers. *arXiv preprint arXiv:2503.08280*, 2025.

[55] Bochao Wang, Huabin Zheng, Xiaodan Liang, Yimin Chen, and Meng Yang. Toward characteristic-preserving image-based virtual try-on network. In *15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIII*, 2018.

[56] Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. Dire for diffusion-generated image detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22445–22455, 2023.

[57] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[58] Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Shuting Wang, Tiejun Huang, and Zheng Liu. Omnigen: Unified image generation. *CoRR*, 2024.

[59] Zhenyu Xie, Zaiyu Huang, Xin Dong, Fuwei Zhao, Haoye Dong, Xijin Zhang, Feida Zhu, and Xiaodan Liang. GP-VTON: towards general purpose virtual try-on via collaborative local-flow global-parsing learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, 2023.

[60] Yuhao Xu, Tao Gu, Weifeng Chen, and Arlene Chen. Ootdiffusion: Outfitting fusion based latent diffusion for controllable virtual try-on. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 8996–9004, 2025.

[61] Binxin Yang, Shuyang Gu, Bo Zhang, Ting Zhang, Xuejin Chen, Xiaoyan Sun, Dong Chen, and Fang Wen. Paint by example: Exemplar-based image editing with diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*. IEEE, 2023.

[62] Lu Yang, Wenhe Jia, Shan Li, and Qing Song. Deep learning technique for human parsing: A survey and outlook. *International Journal of Computer Vision*, 132(8):3270–3301, 2024.

[63] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *CoRR*, 2023.

[64] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, 2023.

[65] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[66] Xuanpu Zhang, Dan Song, Pengxin Zhan, Tianyu Chang, Jianhao Zeng, Qingguo Chen, Weihua Luo, and Anan Liu. Boow-vton: Boosting in-the-wild virtual try-on via mask-free pseudo data training. *arXiv preprint arXiv:2408.06047*, 2024.

[67] Yuxuan Zhang, Yirui Yuan, Yiren Song, Haofan Wang, and Jiaming Liu. Easycontrol: Adding efficient and flexible control for diffusion transformer. *arXiv preprint arXiv:2503.07027*, 2025.

[68] Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K. Wong. Uni-controlnet: All-in-one control to text-to-image diffusion models. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[69] Nan Zhong, Yiran Xu, Zhenxing Qian, and Xinpeng Zhang. Rich and poor texture contrast: A simple yet effective approach for ai-generated image detection. *CoRR*, 2023.

[70] Luyang Zhu, Dawei Yang, Tyler Zhu, Fitsum Reda, William Chan, Chitwan Saharia, Mohammad Norouzi, and Ira Kemelmacher-Shlizerman. Tryondiffusion: A tale of two unets. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, 2023.

[71] Junhao Zhuang, Yanhong Zeng, Wenran Liu, Chun Yuan, and Kai Chen. A task is worth one word: Learning with task prompts for high-quality versatile image inpainting. In *European Conference on Computer Vision*, pages 195–211. Springer, 2024.

## 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 abstract and introduction claim to present a unified framework of virtual try-on to any wearable objects, which is the key contribution and fully evaluated in experiments on both common and uncommon objects.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We discuss the limitations in Sec. 5.

   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: This paper does not include theoretical result.

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: We report the training details in Sec. 4.1.

   Guidelines:

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

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

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

   Answer: [No]

   Justification: We will release the code, weights and benchmark after formal publication.

   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 experimental setup is presented in Sec. 4.1.

   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 report the design of evaluation metrics in Sec. 3.4, which will be further presented in detail in the appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: We include the compute resources information in Sec. 4.1.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We have reviewed and followed the Code of Ethics for presenting the paper.

   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: In both abstract, introduction and conclusion, we discuss the positive impacts of `OmniTry` for broader application in e-commerce. The negative impacts mainly rely on whether users distribute the try-on results with inconsistent object appearance, which will be considered with model release.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

Answer: [NA]

Justification: The model to be released is restricted on virtual try-on in e-commerce.

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: This paper is built upon FLUX with Apache License 2.0. The data-source consists of open-sourced stock websites and internal datasets.

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: This paper presents a new benchmark on unified try-on, and the details of benchmark is introduced in Sec. 3.4.

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: This paper does not involve crowdsourcing or research with human subjects.

Guidelines:

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

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

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

Answer: [NA]

Justification: The paper is conducted on automatic data preparation and evaluation, without crowdsourcing or human interactions.

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.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method design does not involve any LLM usage.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

# A Details of Benchmark and Metrics

As the pioneering work investigating the unified virtual try-on task, we construct a comprehensive evaluation benchmark named *OmniTry-Bench*, accompanied by six dedicated metrics to systematically assess the quality of synthesized try-on images.

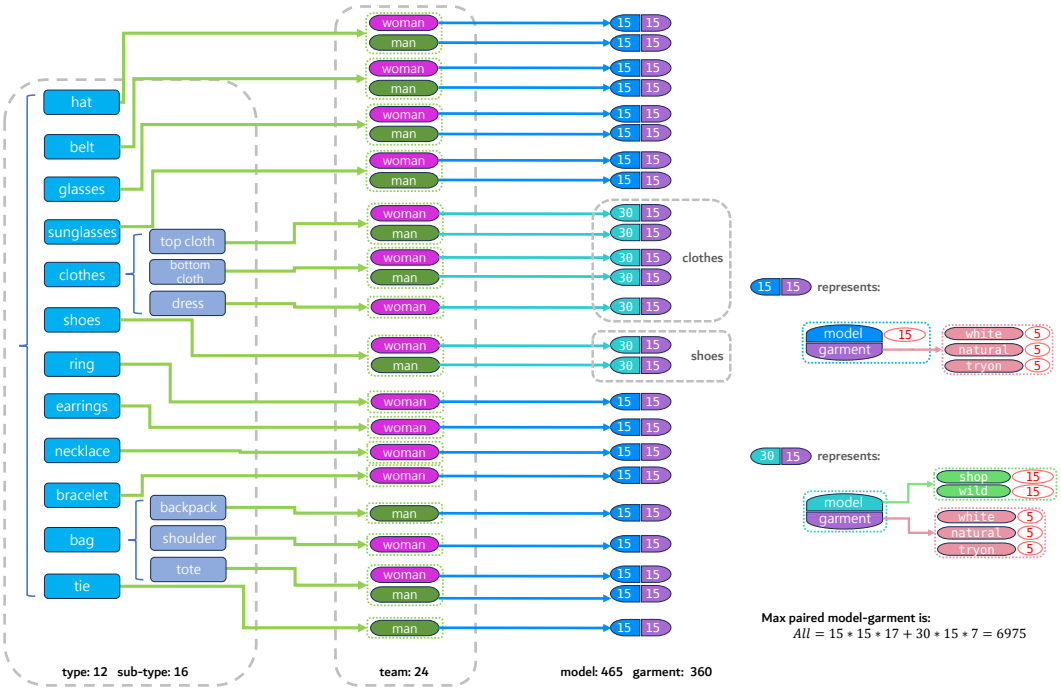



Figure 7: The visualization of the *OmniTry-Bench* constitution.

## A.1 Constitution of Benchmark

As the Figure 7, we gather evaluation samples within 12 common types of wearable objects, which can be summarized into 4 major classes: (i) *clothes* consisting of top, bottom and full-body garments, (ii) *shoes* in common styles, (iii) *jewelries*, including bracelets, earrings, necklaces and rings, (iv) *accessories*, including bags, belts, hats, glasses, sunglasses and ties.

We consider detailed sub-types if necessary, such as the class *bag* consisted of the backpack, shoulder and tote bags. *Clothes* are divided into top cloth, bottom cloth, and dress. Each sub-type contains two gender groups (woman and man), with the exceptions that *jewelries* and *dress* exclusively contain woman samples, while *tie* contains only man samples.

Each gender group includes 15 model images, where the garments are categorized into three settings: white background, natural background, and try-on setting. Every garment setting include 5 images. Following previous work's categorization of virtual try-on scenarios into *in-shop* and *in-the-wild*, we further divide the model images for *clothes* and *shoes* into 15 shop-style and 15 wild-style samples per gender group, resulting in 30 model images per sub-type.

The benchmark predominantly sources images from public repositories (Pexels[2]), supplemented with brand website materials and social media content under compliant data usage protocols.

**Pairing Strategy.** For each gender group, we establish combinatorial pairs between model and garment images through:

- *Maximum Pair Calculation*: $max\_pairs = 15 \times 15 \times 17 + 30 \times 15 \times 7 = 6,975$ pairs, where 17 and 7 denote model settings counts for regular and style -specific categories respectively.

[2] `https://www.pexels.com`

- *Sampled Pair Selection*: $selected\_pairs = 15 \times 15 \times 24 = 360$ paired samples, constrained by single-use garment policy and balanced sampling (15 models per clothes/shoes type, include 7 shop-style and 8 wild-style).

Overall, our experiments are all evaluated on the selected benchmark contains 360 pairs of images

### A.2 Evaluation Metrics

As discussed before, the objectives of try-on can be divided into three aspects. Since there is no ground-truth result in mask-free setting, we redesign the metrics as follows:

*Object Consistency*: We crop the objects from the try-on and object images via masking, then perform white-background normalization on the extracted objects. We compute the visual similarity using DINO [3] and CLIP [49] visual encoders, with metrics denoted as M-DINO and M-CLIP-I. As these metrics measure cosine similarity in the embedding space, their values range in $[-1, 1]$ where higher values indicate better object preservation. The M-DINO scores generally exhibit lower values than M-CLIP-I, as DINO-extracted features are more sensitive to geometric variations compared to CLIP's semantic-aligned embeddings. Our experiments quantitatively validate this behavior across different object categories. This discrepancy stems from their distinct learning objectives:

- **M-DINO** [3]: Learns dense local features through self-supervised distillation, emphasizing spatial consistency of object parts. Then compute the cosine similarity of two features.
- **M-CLIP-I** [49]: Optimizes global semantic alignment between object images, prioritizing category-level coherence. Then compute the cosine similarity of two features. Then compute the cosine similarity of two object features.

*Person Preservation*: We extract the person regions by cropping try-on and original person images, masking the target object areas with black pixels. We then compute spatial-aligned similarity between these aligned image pairs using two complementary metrics:

- **SSIM** (Structural Similarity Index) [57]: Measures structural, luminance, and contrast similarity between images. The metric ranges in $[-1, 1]$ with values approaching 1 indicating higher structural consistency.
- **LPIPS** (Learned Perceptual Image Patch Similarity) [65]: Computes deep feature differences using pretrained VGG networks, better aligning with human perception than traditional metrics. Its values lie in $[0, 1]$ where lower scores denote better preservation quality.

*Object Localization*: We propose a dual-strategy evaluation framework to assess spatial rationality through complementary approaches:

- **G-Accuracy**: Quantifies detection reliability using GroundingDINO [36] with the following implementation protocol: Invoke *predict_with_classes* API with target object categories as *classes* parameter. Configure detection thresholds: *box_threshold* $= 0.25$ (bounding box confidence) and *text_threshold* $= 0.25$ (text-image alignment). Last, calculate success rate as total test cases correct detections.
- **CLIP-I**: Evaluates semantic alignment through multi-modal similarity measurement: Generate descriptive prompts via Qwen2 [1] MLLM. Compute CLIP [49] embedding similarity between try-on images and generated text. Normalize scores to [0,1] range using min-max scaling.

  The final prompt template is formally defined as follows:

```
"""Generate a detailed description of a composite image by combining
elements from the two provided images:
   1. Image 1: The model's appearance (pose, clothing, facial featur-
   es), background and style
   2. Image 2: Only the <{garment_class}>, without any other infos
   (e.g., background, model)
Describe the synthesized image with the model wearing the {garment_cl-
ass}, in 65 words. Only describe the final imagined scene, without the
detail or information of composite. The main description is from
```

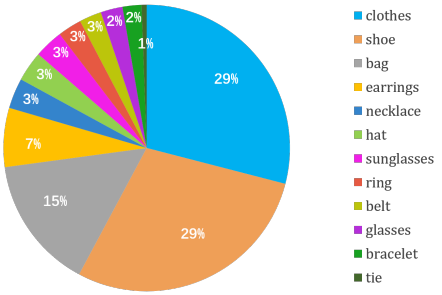
```
Image 1. Briefly and shortly describe the {garment_class} in 6 wor-
ds, no details needed. No words like (e.g., from the Image 2). If
{garment_class} is cloth or dress , the model from the Image 1, re-
place with the {garment_class} from Image 2, no words like (replace
the hair/shirt), using "wear" the {garment_class}.

Examples outputs:
    - "A young woman standing in a studio with a white background.
    She is wearing a denim dress with a button-down collar and long
    sleeves. The dress is knee-length and falls above her knees. T-
    he woman is also wearing black ankle boots with a pointed toe
    and a low heel. She has a brown crossbody bag with a strap acr-
    oss her shoulder. The bag appears to be made of leather and has
    a small flap closure. The overall style of the outfit is casual
    and minimalistic."
    - "A close-up portrait of a young woman's face and upper body.
    She is wearing a black strapless top with a thin silver chain n-
    ecklace around her neck. Her hair is styled in loose waves and
    she is wearing large hoop earrings. The woman is looking off to
    the side with a serious expression on her face. The background
    is plain white."
    - "A close-up portrait of a woman's upper body. She is wearing
    a black collared shirt with a button-down collar and long slee-
    ves. Her hair is styled in loose curls and she is wearing large,
    dangling earrings. Her hand is resting on her chest, with a lar-
    ge ring on her ring finger. The background is plain white. The
    woman appears to be looking off to the side with a serious expr-
    ession on her face."
"""
```

# B Details of Training Dataset

## B.1 Dataset for Stage-1

The model in the first stage is jointly trained on two datasets, *i.e.,* the unpaired in-the-wild images, and the dataset of stage-2 without the object image. We train on the datasets with sampling ratios of 2 : 1. To further investigate the class distribution in the unpaired dataset, we count the highly-frequent words in the object text descriptions. After filtering out the prepositions and verbs, the top-5 words are necklace, hat, glasses, sunglasses and watch. We also observe some classes excluded in our final 12 common classes, *e.g.,* smartphone, cup, scarf, crown and mask. The rich distribution of wearable or holdable objects enhances the generalization of `OmniTry` to uncommon classes.

Figure 8: The class distribution of training dataset.




We also report the scale of dataset during the data preparation. The initial dataset contains 152K in-the-wild images, which are filtered to be 111K images with person and wearable objects. After listing, grounding and removing objects, the total amount of images containing at least one object is 94K, and the corresponding number of objects is 189K (roughly 2 objects per image).

### B.2 Dataset for Stage-2

For the training dataset of the second stage, we visualize the amount of samples for each class in Appendix B . It is shown that the most common classes, *i.e.,* clothes and shoes, constitute more half of the total dataset, while most classes lay in the long-tail of distribution with less than $3\%$. Such a distribution is aligned with our basic assumption that it is hard to obtained paired samples for many wearable objects. For class-balanced training, we manfully assign the sampling weights for clothes, shoes and bags as $4, 4, 3$, and set weights as $1$ for remaining classes.

## C Details of Training and Model Architecture

### C.1 Training Configuration

During training, we resize the image with fixed aspect ratio to be no larger than $1$ million, which means that the model could receive images with varying aspect ratios in one batch. To handle this, we pad the image tokens into the same length of sequence, and modify the attention block to forward only on the valid tokens.

For both training of stage-1 and stage-2, we set the learning rate as $1^{-4}$, gradient accumulation steps as $1$, weight decay as $0.01$ and gradient norm clipping as $1.0$. We use the AdamW [38] optimizer with hyper-parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The model is trained with mixed precision of bfloat16. We note that since we fine-tune based on the distilled version of FLUX [32], the guidance scale is fixed as $1$ during training, and set as $30$ during inference.

### C.2 Details of Re-purposing Inpainting Model

We elaborate the details of adapting the inpainting model, FLUX.1-Fill in this paper, towards mask-free try-on task. During training, the input of model can be split into two sets in sequence dimension:

- The try-on image. Along the channel dimension, it contains the noisy ground-truth try-on image, the input person image and a zero mask in the same shape.
- The object image. Along the channel dimension, it contains the noisy object image, the clean noisy image and a zero mask.

Then during the inference stage, we initialize the above input while replacing the noisy latents with standard Gaussian noise. Through the above formulation, it is shown that the inputs of person and object images are different. The person branch aims to modify the input person image in proper area, while the object branch simply targets to maintain the input, and transfers the object appearance via full attention mechanism.

### C.3 Details of Masked Full-Attention

We discuss the details of applying masked full-attention in the second stage. We set text prompts for both try-on and object images, like "trying on sunglasses". Suppose the length of tokens to be: $L_{I1}$ for try-on image, $L_{T1}$ for try-on text, $L_{I2}$ for object image, and $L_{T2}$ for object text. We concatenate all tokens in the above order. Then the attention mask is:

$$\begin{bmatrix} 1_{L_{I1}\times L_{I1}} & 1_{L_{I1}\times L_{T1}} & 1_{L_{I1}\times L_{I2}} & 0_{L_{I1}\times L_{I1}} \\ 1_{L_{I1}\times L_{I1}} & 1_{L_{I1}\times L_{T1}} & 0_{L_{I1}\times L_{I2}} & 0_{L_{I1}\times L_{I1}} \\ 0_{L_{I1}\times L_{I1}} & 0_{L_{I1}\times L_{T1}} & 1_{L_{I1}\times L_{I2}} & 1_{L_{I1}\times L_{I1}} \\ 0_{L_{I1}\times L_{I1}} & 0_{L_{I1}\times L_{T1}} & 1_{L_{I1}\times L_{I2}} & 1_{L_{I1}\times L_{I1}} \end{bmatrix}, \tag{2}$$

where $1_{m\times n}$ denotes all-one matrix and $0_{m\times n}$ denotes all-zero matrix. More specifically, we apply such a full-attention in both the multi-modality blocks and single blocks of FLUX [32], and figure out the text tokens to achieve the masking. We leverage the attention function with varying length in FlashAttention [12] to implement the block-wise masked attention.

### C.4 LoRA Implementation

We implement the location and identity adapters with LoRA [21]. In detail, we set the rank and $\alpha$ to be 16. We insert the LoRA module into the following layers: the projection into query/key/value, output projection of attention, the linear layers in feedforward block, the layer normalization layer, the input patch projection, and the final output projection.

## D Details of Compared Methods

In this section, we present the details of compared methods and our implementation of them on try-on task. We also report more results of the variants of each method, among which we only report the best result in main experiment.

### D.1 General Customized Image Generation

**OneDiffusion** [33]: A large-scale diffusion framework supporting bidirectional image synthesis across tasks. We evaluated its performance on mask-free/mask-based try-on through instruction-based cases. We also modify its original instructing prompt to achieve better performance.

**OmniGen** [58]: A vision-language unified framework consolidating multiple tasks, supporting both mask-free/mask-based generation. We also test it with both standard and our optimized prompts.

**VisualCloze** [35] implements visual in-context learning for domain generalization. We conduct experiments with single example and multiple examples in the context.

**Paint-by-Example** [61] enables to re-paint a given subject into image via CLIP-based object representation with mask dependency.

**MimicBrush** [5] achieves imitative inpainting for region-specific edits, requiring the input image with mask, together with the reference image without mask.

**ACE++** [40] extends long-context conditioning for instruction-driven generation that tackles various.

### D.2 Image-based Virtual Try-On

**OOTDiffusion** [60] designs a two-branch U-Net architecture to consume the person and garment images, which requires masked input in the person branch.

**Magic Clothing** [4] introduces a garment extractor to progressively insert garment features into the main backbone of try-on generation. Magic Clothing supports the input of either masked person image, or the targeting pose and person ID image. We adapt the former setting to better preserve the person image.FI

**CatVTON** [10] proposes to transfer the identity of garment by simply concatenating it with the person image, and achieve mask-based try-on with inpainting model.

**FitDiT** [25] introduces diffusion transformer (DiT) model into VTON, and designs a GarmentDiT and a DenoisingDiT to implement this task.

**Any2AnyTryon** [18] is the only open-source mask-free VTON model, eliminates the dependence on masks, poses, or any other such conditions.

### D.3 More Comparison Results

We report more comparison results in Tab. 3, including variants of methods with mask/mask-free setting, varying image size and different prompt design. We report only the best result of all variants in the main experiment.

## E More Visualization Results

We visualize more try-on results in Fig. 9, where we include all classes in OmniTry-Bench and different sub-types for full visualization.

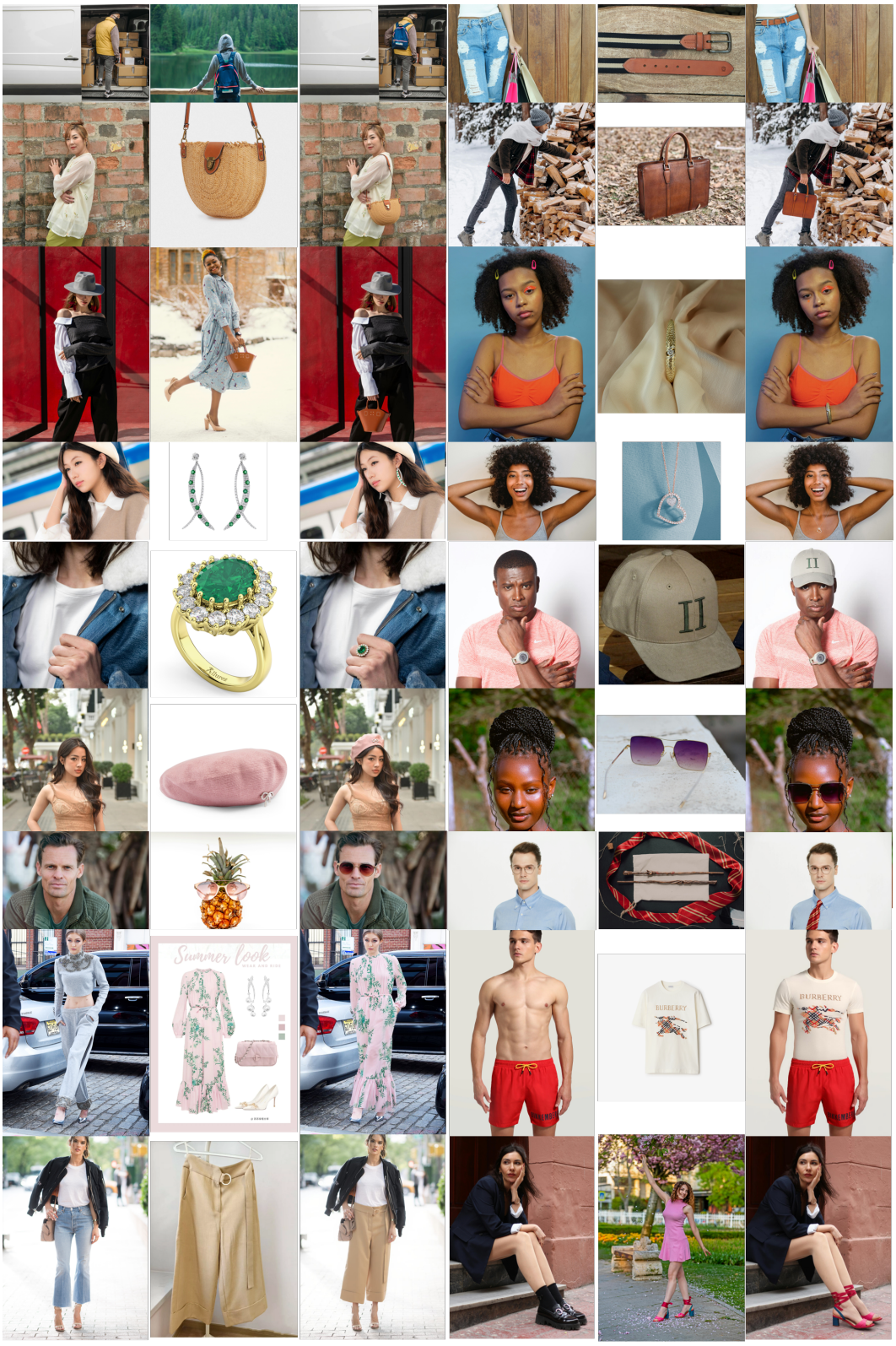



Figure 9: The samples of the model, the object, and the try-on person.

Table 3: More evaluation results of the compared methods with different settings.

| method | mask | Object Consistency | | Person Presevation | | Object Localization | |
|---|---|---|---|---|---|---|---|
| | | M-DINO ↑ | M-CLIP-I ↑ | LPIPS ↓ | SSIM↑ | G-Acc. ↑ | CLIP-T ↑ |
| *on the whole set* | | | | | | | |
| Paint-by-Example ($512^2$) [61] | ✓ | 0.4171 | 0.7328 | 0.4577 | 0.7968 | 0.9833 | 0.2831 |
| Paint-by-Example ($1024^2$) [61] | ✓ | 0.4565 | 0.7727 | 0.3903 | 0.8033 | 0.9861 | 0.2804 |
| MimicBrush [5] | ✓ | 0.4693 | 0.7253 | 0.3033 | 0.8575 | 0.9250 | 0.2781 |
| ACE++ (prompt v1) [40] | ✓ | 0.4565 | 0.7474 | 0.4561 | 0.7519 | 0.9667 | 0.2791 |
| ACE++ (prompt v2) [40] | ✓ | 0.4449 | 0.7427 | 0.4554 | 0.7517 | 0.9722 | 0.2793 |
| VisualCloze (1-example) [35] | ✓ | 0.4705 | 0.7533 | 0.6685 | 0.5320 | 0.9972 | 0.2283 |
| VisualCloze (2-example) [35] | ✓ | 0.4236 | 0.7307 | 0.6767 | 0.4908 | 0.9917 | 0.2260 |
| OmniGen (prompt v2) [58] | ✓ | 0.5151 | 0.7761 | 0.6888 | 0.5870 | 0.9917 | 0.2557 |
| OneDiffusion (prompt v1) [33] | ✓ | 0.5515 | 0.8137 | 0.6607 | 0.6166 | **1.0** | 0.2290 |
| OneDiffusion (prompt v2) [33] | ✓ | 0.5580 | 0.7950 | 0.5795 | 0.6628 | 0.9972 | 0.2401 |
| OneDiffusion (prompt v1) [33] | ✗ | 0.4178 | 0.7358 | 0.7606 | 0.4951 | **1.0** | 0.2309 |
| OneDiffusion (prompt v2) [33] | ✗ | 0.4731 | 0.7749 | 0.7001 | 0.5831 | 0.9972 | 0.2309 |
| VisualCloze (1-example) [35] | ✗ | 0.5292 | 0.7782 | 0.4471 | 0.6190 | 0.9639 | 0.2524 |
| VisualCloze (2-example) [35] | ✗ | 0.4915 | 0.7619 | 0.4730 | 0.5868 | 0.9806 | 0.2540 |
| OmniGen (prompt v1) [58] | ✗ | 0.5299 | 0.7689 | 0.7009 | 0.5727 | 0.9778 | 0.2533 |
| OmniGen (prompt v2) [58] | ✗ | 0.5435 | 0.7869 | 0.6703 | 0.5965 | 0.9944 | 0.2535 |
| `OmniTry` (**Ours**) | ✗ | **0.6160** | **0.8327** | **0.0542** | **0.9333** | 0.9972 | **0.2831** |
| *on the clothes subset* | | | | | | | |
| Magic Clothing [4] | ✓ | 0.5665 | 0.7634 | 0.2761 | 0.8786 | 1.0 | 0.2700 |
| CatVTON [10] | ✓ | 0.5744 | 0.7906 | 0.1664 | **0.9283** | 1.0 | 0.2818 |
| CatVTON (w. garment mask) [10] | ✓ | 0.5534 | 0.7843 | 0.2084 | 0.8828 | 1.0 | 0.2797 |
| OOTDiffusion [60] | ✓ | 0.5961 | 0.8016 | 0.2178 | 0.8865 | 1.0 | 0.2761 |
| FitDiT ($768 \times 1024$) [25] | ✓ | 0.6718 | 0.8324 | 0.1972 | 0.8952 | 1.0 | 0.2822 |
| FitDiT ($1152 \times 1536$) [25] | ✓ | 0.6733 | 0.8340 | 0.1618 | 0.9027 | 1.0 | 0.2831 |
| FitDiT ($1536 \times 2048$) [25] | ✓ | 0.5961 | 0.8016 | 0.2178 | 0.8865 | 1.0 | 0.2761 |
| Any2AnyTryon [18] | ✗ | 0.6747 | 0.8537 | 0.2089 | 0.8969 | 1.0 | **0.2832** |
| `OmniTry` (**Ours**) | ✗ | **0.6995** | **0.8560** | **0.1021** | 0.9105 | **1.0** | 0.2799 |

Table 4: Human evaluation results of `OmniTry` and garment-only methods.

| **Method** | **Magic Clothing** | **CatVTON** | **OOTDiffusion** | **FitDiT** | **Any2AnyTryon** | `OmniTry` **(Ours)** |
|---|---|---|---|---|---|---|
| **Avg. Rank** ↓ | 4.27 | 3.36 | 3.70 | 2.28 | 0.77 | **0.62** |

# F Human Evaluation of the Generated Try-ons

We conduct a human evaluation to assess the realism and usefulness of the generated try-on results, especially in comparison with garment-only methods. Specifically, we invite five annotators to rank the outputs of different methods based on three aspects: try-on success rate, garment consistency, and overall realism. The average ranking results are summarized in Tab. 4, where a lower value indicates a better ranking. As shown, OmniTry achieves the best overall performance among all compared methods.

# G Differences between Stage-1 of `OmniTry` and Editing Methods

The key differences between the stage-1 of `OmniTry` and the editing methods that support the "Add" operation can be summarized as follows. (1) Task and performance: The general editing methods typically involve a wide range of editing tasks, thus may show restricted performance on specific operation, especially on try-on cases requiring fine-grained combination of the added object and the original image. The added object could be more likely to be an independent item, while `OmniTry` focuses on natural combination with parts of input person. (2) Method: The stage-1 of `OmniTry` is designed by re-purposing an inpainting-based model to mask-free editing, leveraging its ability of

Table 5: Compraison between `OmniTry` (stage-1) and editing methods supporting "Add" operation.

| Method | LPIPS | SSIM | G-Acc. | CLIP-T |
|---|---|---|---|---|
| **AnyEdit** | 0.1112 | 0.8455 | 0.8167 | 0.2415 |
| **OmniGen** | 0.3381 | 0.6394 | 0.9889 | **0.2654** |
| `OmniTry` **(stage-1)** | **0.0711** | **0.8959** | **0.9944** | 0.2613 |

Table 6: Additional ablation study on one-stream vs. two-stream adapter.

| Method | trainable params. | M-DINO | M-CLIP | LPIPS | SSIM | G-Acc. | CLIP-T |
|---|---|---|---|---|---|---|---|
| two-stream | 172M (2 LoRA with r=16) | **0.5845** | **0.8159** | **0.0425** | 0.9403 | 0.9806 | **0.2620** |
| one-stream | 172M (1 LoRA with r=32) | 0.5619 | 0.8149 | 0.0439 | **0.9478** | **0.9861** | 0.2604 |

detailed local editing. Specifically, the original image and generated image are concatenated in the channel dimension. However, the general editing methods require larger divergence between the input and output, and are concatenated in the sequential dimension (e.g., UniReal [7] and OmniGen [58]), showing higher computation cost (2x sequence length).

We compare AnyEdit [26] and OmniGen [58] with the stage-one model of OmniTry in Tab. 5, with the metrics of object localization and person preservation. We observe that OmniGen could not guarantee to preserve the original image (similar to its performance in stage-2). For AnyEdit, though it preserve the input image, it could sometimes fail to add any object (worse G-Acc.) or properly combine the object onto the person. We will also include visualization result in revised version.

## H Additional Ablation Study on One-Stream vs. Two-Stream Adapter

To further ensure the alignment of trainable parameters, we train a new location adapter with double LoRA rank (r=32) from stage-1, and initialize it into the second stage for one-stream training. We note that the additional computation cost is doubled than two-stream adapters with r=16. We initialize both settings from an earlier checkpoints of stage-1 with the same training steps to ensure fair comparsion. The results in Tab. 6 show that though with less computation cost, the two-stream setting still shows better performance to seperately cope with different capabilities of `OmniTry`.

## I More Discussion on Unexpected Shortcut in Stage-1

In stage-1, we observe that using naively erased training samples leads the model to produce output images that almost perfectly recover the position and shape of the object in the ground-truth image. We hypothesize that this phenomenon is likely caused by information leakage, for the following reasons. (1) The reconstruction shown in Fig. 3 primarily reflects shape and position reconstruction, rather than appearance reconstruction. Since no object image is provided to the first stage, the model generates objects with diverse appearances but consistently reproduces the same shape and position as the ground-truth object. This observation suggests that the model might exploit the boundary of the erased region, enabling it to perfectly reconstruct the object's location and outline. (2) A stronger piece of evidence is observed when we train the model with traceless erasing under the same number of training steps. In this case, the model produces objects with random shapes, positions, and appearances, even when evaluated on the training samples, indicating that the shape recovery in the naive erasing setup indeed stems from boundary leakage.