# DEFT-VTON: Efficient Virtual Try-On with Consistent Generalised H-Transform

Xingzi Xu[1,2 *†] Qi Li[1*] Shuwen Qiu[3†] Julien Han[1] Karim Bouyarmane[1]
[1]Amazon [2]Duke University [3] University of California, Los Angeles (UCLA)
[1]{qlimz,hameng,bouykari}@amazon.com
[2]xingzi.xu@duke.edu [3]janetqiu@cs.ucla.edu
https://deft-vton.github.io/

## Abstract

*Diffusion models enables high-quality virtual try-on (VTO) with their established image synthesis abilities. Despite the extensive end-to-end training of large pre-trained models involved in current VTO methods, real-world applications often prioritize limited training and inferencing/serving/deployment budgets for VTO. To solve this obstacle, we apply Doob's h-transform efficient fine-tuning (DEFT) for adapting large pre-trained unconditional models for downstream image-conditioned VTO abilities. DEFT freezes the pre-trained model's parameters and trains a small h-transform network to learn a conditional h-transform. The h-transform network allows to train only* $1.42\%$ *of the frozen parameters, compared to baseline* $5.52\%$ *in traditional parameter-efficient fine-tuning (PEFT). To further improve DEFT's performance, and decrease existing models' inference time, we additionally propose an adaptive consistency loss. Consistency training distills slow but performing diffusion models into a fast one while retaining performances by enforcing consistencies along the inference path. Inspired by constrained optimization, instead of distillation, we combine the consistency loss and the denoising score matching loss in a data-adaptive manner for fine-tuning existing VTO models at a low cost. Empirical results show proposed DEFT-VTON method achieves SOTA performances on VTO tasks, as well as a number of function evaluations (denoising steps) as low as 15, while maintaining competitive performances.*

## 1. Introduction

Virtual try-on (VTO) technology has emerged as a transformative solution in the e-commerce fashion and general image editing industries, addressing the critical gap between online shopping convenience and the traditional in-store fitting experience [1–3]. VTO allows customers to visualize how clothing items would look on themselves without physical interaction with the garments, revolutionizing the online shopping experience. Virtual try-on technology has evolved significantly with the emergence of denoising diffusion models. Denoising diffusion models are a class of expressive generative models that progressively convert noise to realistic data [4–6].

Large-scale denoising diffusion models have achieved unparalleled success on unconditional image synthesis [7]. Unconditional diffusion models approximate the score of the underlying distribution $s^\theta(t, \mathbf{x}) \approx \nabla_{\mathbf{x}} \log p_t(\mathbf{x})$. For conditional generations regarding a constraint $\mathbf{y}$, we can estimate the conditional score based on the unconditional one through the Bayes' theorem:

$$\nabla_{\mathbf{x}} \log p_t(\mathbf{x}|\mathbf{Y} = \mathbf{y}) \approx s^\theta(t, \mathbf{x}) + \nabla_{\mathbf{x}} \log p_t(\mathbf{Y} = \mathbf{y}|\mathbf{x}), \tag{1}$$

where $\nabla_{\mathbf{x}} \log p_t(\mathbf{Y} = \mathbf{y}|\mathbf{x})$ represents the guidance guiding the denoising process towards the conditioned results. Existing methods explore training-free sampling of conditioned processes with an unconditioned model as well as fine-tuning methods. Although training-free methods do not suffer from extensive retraining and fine-tuning of the unconditional model, they suffer from a slow sampling process rooting in the expensive estimation of the guidance and slower convergence rates than their unconditional counterpart [8, 9]. On the other hand, fine-tuning methods on large unconditional methods require impractical training costs, as well as large amounts of paired data points [10–13]. Doob's h-transform efficient fine-tuning (DEFT) employs a large-scale pre-trained unconditional diffusion model, which can achieve high-quality unconditional generation, but is computationally intractable to fine-tune [14]. DEFT instead suggests learning a small-scale network $h$ to directly approximate the guidance $\nabla_{\mathbf{x}} \log p_t(\mathbf{Y} = \mathbf{y}|\mathbf{x})$.

*Equal contribution.
†Work done during internship at Amazon.

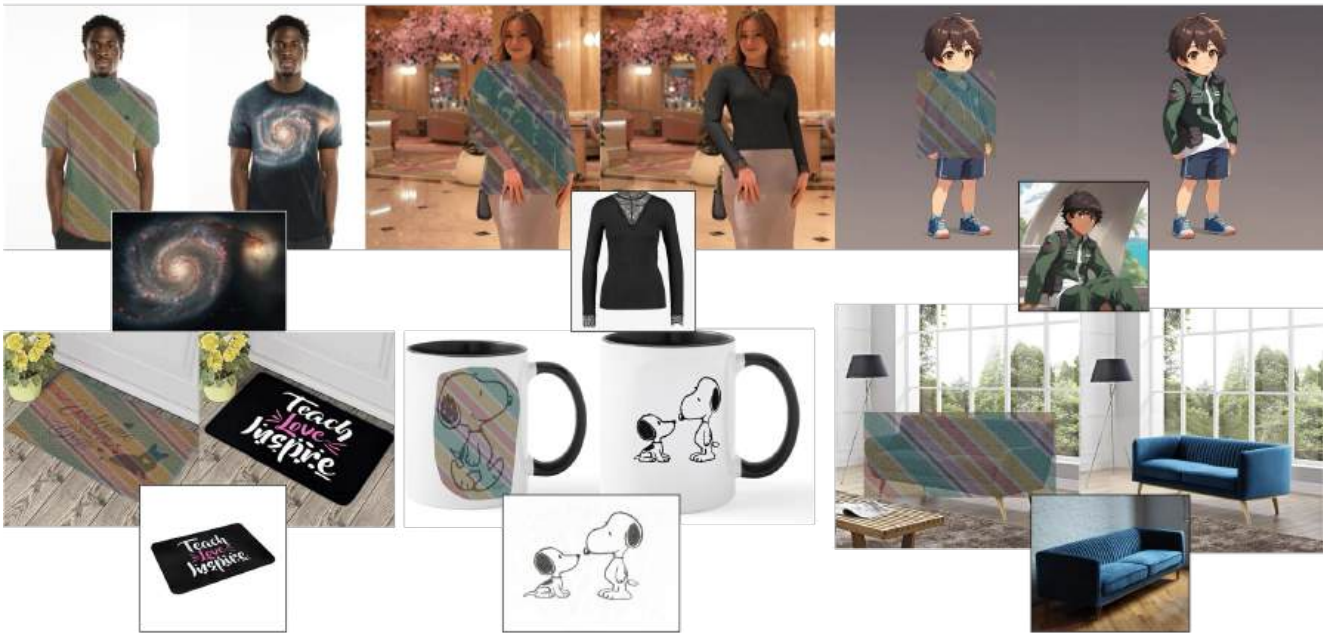


Figure 1. DEFT-VTON achieves realistic results across diverse application domains.

DEFT freezes the weights of the unconditional model, and achieves state-of-the-art (SOTA) perceptual qualities on tasks such as image reconstruction while achieving speedups [14, 15].

Despite their superior generation capabilities, diffusion models are slower than VAEs and GANs because of their large number of function evaluations (NFEs), even with highly optimized samplers [16–18]. Meanwhile, distilling, pre-training, and regularizing with consistency losses along inference paths have provided improved generation qualities and speeds [19–23].

We propose DEFT-VTON to train an adaptor that directs unconditional generations of a pre-trained diffusion model to that of a VTO conditioned one. In particular, we propose an adaptive loss that balances the consistency loss and the DEFT loss on the fly, achieving SOTA performances on VTO tasks, while using only a limited computational budget. Empirical results suggest adding the consistency loss speeds up the VTO inference up to 40% while maintaining the same performances.

## 2. Related works

**Virtual try-on** Given an image of a person and a target garment, virtual try-on (VTO) methods aim to generate an image of the person wearing the target garment, while preserving the garment's fine-grained details and blending naturally into the surrounding context [1]. Traditional methods combine pose, body shape, and other garment-agnostic person representations with images of the target garment for generation [24–34, 34–38].

**Consistency models** Consistency models accelerate and even avoid the iterative sampling process of diffusion models, supporting fast one-step generation by design, while still allowing multi-step sampling to trade compute for sample quality [19]. Given a probability flow ODE that smoothly converts data to noise, consistency models learn to map any point on the ODE trajectory to the data for generative modeling. As all points along the ODE trajectory are trained to map to the same data point, these mappings are called consistency models. [20, 22, 39, 40] expands on the idea, achieving better generation qualities while maintaining low NFEs. Remaining consistent along generation paths have also proved beneficial to avoiding sampling errors, and improves generation performances [21].

**Conditional diffusion models** Conditional generation with diffusion models can be generally divided into training-free and training methods. Training-free methods evaluate score guidance at inference time, guiding unconditional generations towards that of a conditional one [8–10, 15, 41]. Despite these methods do not require any train-

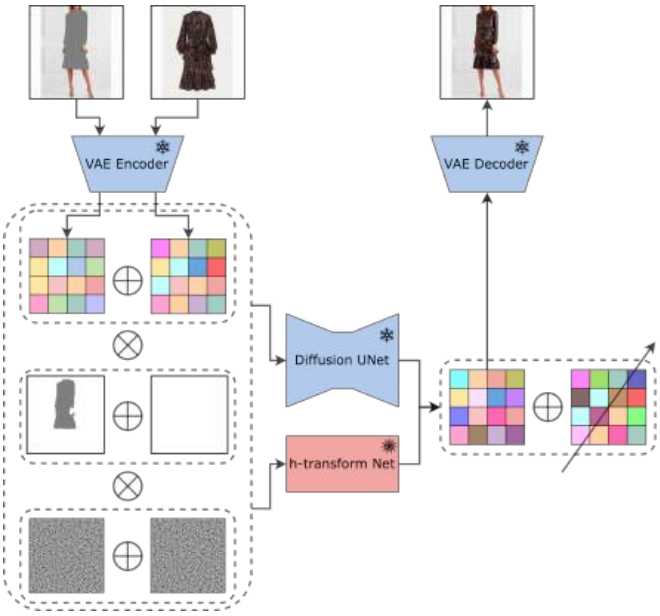


Figure 2. We train an efficient h-transform network to perform virtual try-on tasks. $\oplus$ indicates combination along spatial dimension, $\otimes$ indicates combination along channel dimension, $\nearrow$ indicates discarding the specified part of the tensor, blue indicates frozen parameters, red indicates trainable parameters.

ing computations, they require expensive inference time evaluations, hindering their use in large scale and high resolution VTO tasks. [10, 11] fine-tunes the pre-trained model for better conditional performances. However, such fine-tuning can be unpractical for large scale pre-trained models. [7] trains a small-scale classifier and use a guidance based on the gradient with respect to the classifier at inference time. However, this classifier must be trained on noisy data so it is generally not possible to plug in a pre-trained classifier, and the training and sampling resembles that of a GAN, leading to an inflated performance on the evaluation criteria [11]. DEFT trains an adaptor network extra to the pre-trained model to approximate a Doob's h-transform function, efficiently achieving SOTA perceptual quality and inference speeds on image reconstruction tasks.

## 3. Method

We now introduce our proposed DEFT-VTON method. We first give brief reviews to a baseline VTO model, DEFT, and consistency model, and then combine with constrained optimization to propose an adaptive loss used for training DEFT-VTON.

### 3.1. Baseline model

We use a Latent Diffusion baseline VTO architecture as base model, for similar architectures see [1, 6, 34]. The training process takes a target image $\mathbf{I}_p \in \mathbb{R}^{3\times H\times W}$, a binary garment reference image $\mathbf{I}_g \in \mathbb{R}^{3\times H\times W}$, and a mask image $\mathbf{I}_M \in \mathbb{R}^{H\times W}$. We define the garment agnostic image as:

$$\mathbf{I}_a = \mathbf{I}_p \circ \mathbf{I}_M, \tag{2}$$

where $\circ$ represents the element-wise product. Then we apply pre-trained VAE encoder $\varepsilon$ to the garment agnostic image $\mathbf{I}_a$ and the garment reference image (in the form of in-shop garment or used in-scene) $\mathbf{I}_g$:

$$\mathbf{Z}_a = \varepsilon(\mathbf{I}_a), \tag{3}$$
$$\mathbf{Z}_g = \varepsilon(\mathbf{I}_g), \tag{4}$$

where $\mathbf{Z}_a, \mathbf{Z}_g \in \mathbb{R}^{C_\mathbf{Z}, H_\mathbf{Z}, W_\mathbf{Z}}$, with $H_\mathbf{Z}, W_\mathbf{Z} \ll H, W$. We interpolate the mask $\mathbf{I}_M$ to match the latent space size and get $\mathbf{I}_m \in \mathbb{R}^{C_\mathbf{Z}\times H_\mathbf{Z}\times W_\mathbf{Z}}$. Next, we join $\mathbf{Z}_a, \mathbf{Z}_g$ along the spatial dimension to get $\mathbf{Z}_c \in \mathbb{R}^{C_\mathbf{Z}\times H_\mathbf{Z}\times 2W_\mathbf{Z}}$, and join the mask $I_m$ with an all-zero matrix of the same size to create $\mathbf{I}_{mc} \in \mathbb{R}^{C_\mathbf{Z}\times H_\mathbf{Z}\times 2W_\mathbf{Z}}$:

$$\mathbf{Z}_c = \mathbf{Z}_a \oplus \mathbf{Z}_g,\ \mathbf{I}_{mc} = \mathbf{I}_m \oplus \mathbb{O}, \tag{5}$$

where $\oplus$ represents joining along the spatial dimension, and $\mathbb{O}$ represents the all-zero matrix. At the beginning of the denoising process, we sample a Gaussian noise matrix $\mathbf{W} \sim \mathcal{N}(0, \mathbb{I})$ of the same shape as $\mathbf{Z}_c$ and $\mathbf{I}_{mc}$ to join with them along the channel dimension and get

$$\mathbf{X}_T = \mathbf{Z}_c \otimes \mathbf{I}_{mc} \otimes \mathbf{W}, \tag{6}$$

where $\otimes$ represents joining along the channel dimension. $\mathbf{X}_T$ goes through the UNet s to predict $\mathbf{X}_{T-1}$, and iterate

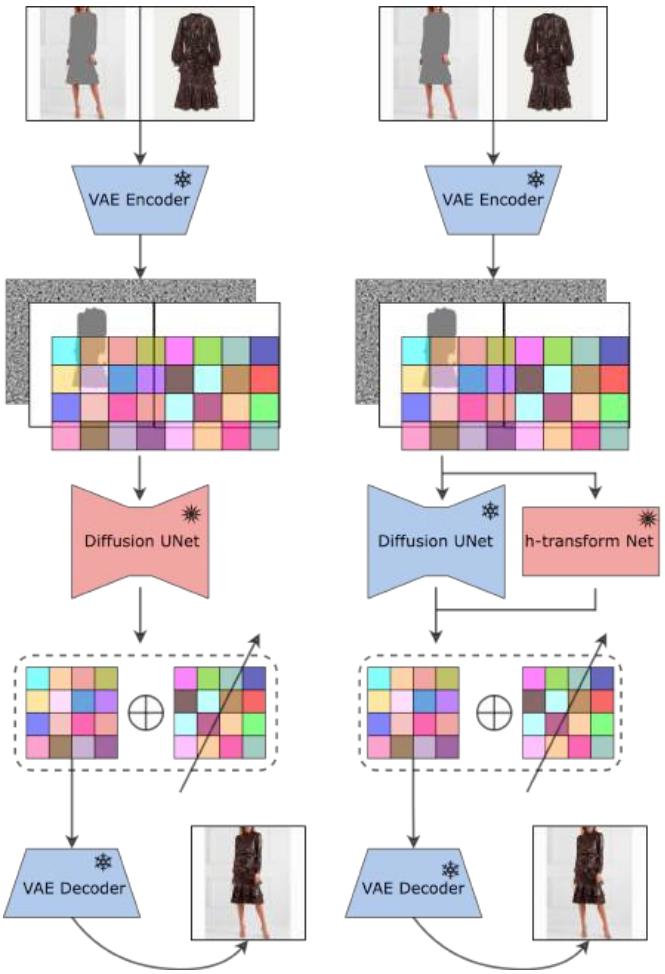


Figure 3. Structure of DEFT-VTON (right) compared to baseline PEFT architecture (left). While the baseline PEFT trains 5.51% percent of the backbone network, we freeze the backbone completely and train an auxiliary network with 1.42% percent of the backbone network parameters.

for a large number $T$ times to predict a clean $\mathbf{X}_0$. In particular, we have

$$\mathbf{X}_{t-1} = \mathbf{s}(\mathbf{Z}_c \otimes \mathbf{I}_{mc} \otimes \mathbf{X}_t, t). \tag{7}$$

Finally, we split $\mathbf{X}_0 \in \mathbb{R}^{C_\mathbf{Z} \times H_\mathbf{Z} \times 2W_\mathbf{Z}}$ along the spatial dimension to get the latent VTO result $\mathbf{X}_0^{\text{VTO}} \in \mathbb{R}^{C_\mathbf{Z} \times H_\mathbf{Z} \times W_\mathbf{Z}}$, and use the pre-trained VAE decoder $\mathbb{D}$ to transform back to the image space, getting our VTO image prediction $\hat{I}^{\text{VTO}} \in \mathbb{R}^{3 \times H \times W}$.

The baseline model freezes the weights of the VAE, and performs a parameter-efficient training of the diffusion U-Net. In particular, the model training consists of first adding various levels of noises to the encoded target image $\varepsilon(\mathbf{I}_p)$ and then predicting the noises added [42].

### 3.2. DEFT

We now give a brief review to conditioning diffusions with the h-transform. Starting with a forward stochastic differential equation (SDE) transforming a data distribution $\mathcal{P}_0 = p_{\text{data}}$ to a Gaussian distribution $\mathcal{P}_T = \mathcal{N}(0, \mathbb{I})$:

$$d\mathbf{X}_t = f_t(\mathbf{X}_t)dt + \sigma_t d\mathbf{W}_t, \mathbf{X}_0 \sim \mathcal{P}_0, \tag{8}$$

where the drift $f_t$ and the diffusion $\sigma_t$ are given explicitly. Specifically, in DDPM discretizations, we have $f_t(\mathbf{X}_t) = -\frac{\beta(t)}{2}\mathbf{X}_t$, and $\sigma_t = \sqrt{\beta(t)}$, with $\beta(t)$ explicitly given. Under mild assumptions, there exists a reverse SDE with drift $b_t$ that transforms $\mathcal{P}_T$ back to $\mathcal{P}_0$:

$$d\mathbf{X}_t = b_t(\mathbf{X}_t)dt + \sigma_t d\mathbf{W}_t, \mathbf{X}_T \sim \mathcal{P}_T, \tag{9}$$

$$b_t(\mathbf{X}_t) = f_t(\mathbf{X}_t) - \sigma_t^2 \nabla_{\mathbf{X}_t} \log p_t(\mathbf{X}_t). \tag{10}$$

Unconditional diffusion models trains by approximating the score function $\mathbf{s}(\mathbf{X}_t, t) = \nabla_{\mathbf{X}_t} \log p_t(\mathbf{X}_t)$, which can be cheaply evaluated for certain drift and diffusion functions [16, 18].

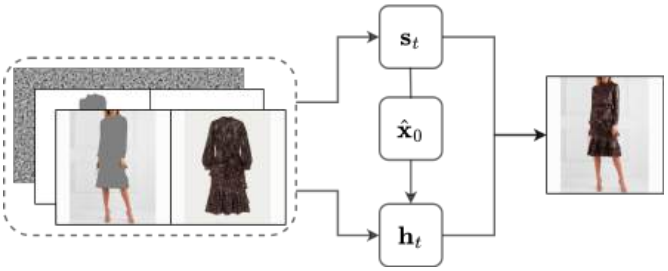


Figure 4. H-transform network takes the same inputs as the diffusion U-Net, and guides the unconditional generator towards VTON results.

Conditional diffusion models want to condition the reverse SDE on a particular observation. Doob's h-transform provides a mathematical tool for guiding an SDE to hit a event at a given time [43, 44]:

**Theorem 1 (Doob's h-transform [14, 44])** *Consider the equation 9, the conditioned process $\mathbf{X}_t|\mathbf{X}_0 \in B$ is a solution to:*

$$\begin{cases} d\mathbf{H}_t = (b_t(\mathbf{H}_t) - \sigma_t^2 \nabla_{\mathbf{H}_t} \log p_{0|t}(\mathbf{X}_0 \in B|\mathbf{H}_t))dt + \sigma_t d\mathbf{W}_t, \\ \mathbf{H}_t \sim \mathcal{P}_T, \end{cases}$$

*where we represent unconditional processes with $\mathbf{X}_t$ and conditional processes with $\mathbf{H}_t$, $b_t(\mathbf{H}_t) = f_t(\mathbf{H}_t) - \sigma_t^2 \nabla_{\mathbf{H}_t} \log p_t(\mathbf{H}_t)$ and $\mathbb{P}(\mathbf{X}_0 \in B) = 1$.*

In the case of VTO, we are interested in inpainting tasks, and conditioning on $\mathbf{Y} = \mathbf{Z}_c \otimes \mathbf{I}_{mc}$, where $\mathbf{Z}_c$ and $\mathbf{I}_{mc}$ are the latent reference and the interpolated mask, as defined in equation 5. We aim to sample from the posterior $p(\mathbf{X} = x_0|\mathbf{Y} = \mathbf{y})$. The h-transform admits a denoising score matching objective given as follows:

**Theorem 2 (DSM for generalised h-transform [14])** *For a given condition* $\mathbf{y}$*, let* $\mathbb{Q}$ *be the path measure of the conditional SDE*

$$d\mathbf{H}_t = (f_t(\mathbf{H}_t) - \sigma_t^2(\nabla_{\mathbf{H}_t} \log p_t(\mathbf{H}_t)))dt + \sigma_t d\mathbf{W}_t, \tag{11}$$

*where* $\mathbf{H}_T \sim Q_T^{f_t}[p(\mathbf{x}_0|\mathbf{y})] = \int p_{T|0}(\mathbf{x}|\mathbf{x}_0)p(\mathbf{x}_0|\mathbf{y})d\mathbf{x}_0$ *converges to the Gaussian distribution* $\mathcal{N}(0, I)$ *exponentially w.r.t.* $T$ *for VP-SDE. The h-transform then admits a denoising score matching (DSM) objective:*

$$\mathbf{h}^* = \arg\min_{\mathbf{h}\in\mathcal{H}} \mathcal{L}_{DSM}^{\mathbf{y}}(\mathbf{h}), \tag{12}$$

$$\mathcal{L}_{DSM}^{\mathbf{y}}(\mathbf{h}) \equiv \mathbb{E}_{\mathbf{X}_0\sim p(\mathbf{x}_0|\mathbf{y}), t\sim Unif(0,T), \mathbf{H}_t\sim p_{t|0}(\mathbf{x}_t|\mathbf{x}_0)} \tag{13}$$

$$[\|(h_t(\mathbf{H}_t) + \nabla_{\mathbf{H}_t} \log p_t(\mathbf{H}_t)) - \nabla_{\mathbf{H}_t} \log p_{t|0}(\mathbf{H}_t|\mathbf{X}_0)\|^2] \tag{14}$$

Theorem 2 induces that the generalized h-transform $\mathbf{h}_t(\mathbf{x}, \mathbf{y}) = \nabla_{\mathbf{x}} \log p_t(\mathbf{y}|\mathbf{x})$ can be approximated by solving the minimization problem

$$\min_{\mathbf{h}\in\mathcal{H}} \mathbb{E}_{\mathbf{y}\sim\mathbf{Y}}[\mathcal{L}_{DSM}^{\mathbf{y}}(\mathbf{h})]. \tag{15}$$

With a DDPM discretization of the SDE and a epsilon-matching formulated pre-trained model $\mathbf{s}_t^{\theta^*}$, this minimization problem 15 reduces to

$$\begin{cases} \min_{h\in\mathcal{H}} \mathbb{E}_{(\mathbf{x}_0,\mathbf{y}),\epsilon,t}[\|(h(\mathbf{H}_t, \mathbf{y}) + \mathbf{s}_t^{\theta^*}(\mathbf{H}_t, t)) - \epsilon\|^2], \\ \mathbf{H}_t = \sqrt{\bar{\alpha}_t}\mathbf{X}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \\ (\mathbf{x}_0, \mathbf{y}) \sim p(\mathbf{X}_0, \mathbf{Y}), \\ \bar{\alpha}_t = \exp(-\int_0^T \beta(s)ds), \\ \epsilon \sim \mathcal{N}(0, \mathbf{I}). \end{cases}$$

We use this minimization formulation for learning the VTO h-transform network to guide an unconditional diffusion model.

### 3.3. Consistency model

Consistency training accelerates diffusion models while maintaining similar performances [23]. At the same time, enforcing consistencies along inference trajectories reduces shift between the training and the sampling distribution of diffusion models and improves the performances [21]. Different from the iterative generation of diffusion models, consistency models are defined as $\mathbf{s} : (\mathbf{x}_t, t) \to \mathbf{x}_\eta$, where $\eta$ is a fixed small number, and are trained to predict the data point in one step [19]. Specifically, consistency models satisfy the self-consistency property:

$$\mathbf{s}(\mathbf{x}_t, t) = \mathbf{s}(\mathbf{x}_{t'}, t'), \forall t, t' \in [\eta, T]. \tag{16}$$

Existing works distill a pre-trained diffusion model into a consistent model, greatly accelerating the generation process [45, 46]. Initializing the distilled parameters with $\theta$,

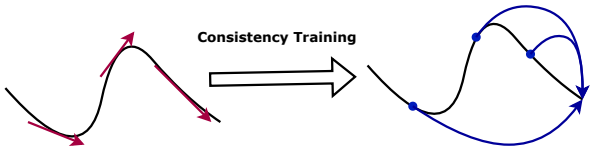


Figure 5. Consistency finetuning reduces number of function evaluations of diffusion models in the sampling process.

we maintain a target model with parameters $\theta^-$, which is updated with the exponential moving average (EMA) of $\theta$, defined as $\theta^- = \mu\theta^- + (1-\mu)\theta$. The distillation model is trained by minimizing the consistency loss:

$$\mathcal{L}(\theta, \theta^-) = \mathbb{E}_{\mathbf{x},n}[d(\mathbf{s}_\theta(\mathbf{x}_{t_{n+1}}, t_{n+1}), \mathbf{s}_{\theta^-}(\mathbf{x}_{t_n}, t_n))], \tag{17}$$

where $d(\cdot,\cdot)$ is a chosen metric. We follow [20] and use the pseudo-Huber loss. In our case, as we do not perform distillations, and have an h-transform network with drastically different structures to the pre-trained diffusion model, we instead define the following training streamline:

$$\begin{cases} \min_{\mathbf{h}\in\mathcal{H}} \mathbb{E}_{\mathbf{x},t}[d(\mathbf{s}_\theta(\mathbf{x}_t, t) + \mathbf{h}(\mathbf{x}_t, t), \mathbf{s}_\theta(\mathbf{x}_{t'}, t') + \mathbf{h}(\mathbf{x}_{t'}, t'))], \\ \mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \\ \epsilon \sim \mathcal{N}(0, \mathbf{I}). \end{cases}$$

We follow the noising schedule and weighting function similar to [20]. Sampling from a consistency model is similar to sampling from a DDIM, where we first predict the clean data at each step, and add slightly less noises. We refer to algorithm 1 of [19] for similar sampling process.

### 3.4. Adaptive loss

We now combine the DEFT training in 3.2 and consistency training in 3.3 adaptively, drawing ideas from constrained optimization. We first perform DEFT training in isolation for a performant h-transform network, using architectures as shown in Figure 2. Afterwards, we finetune the h-transform network with the consistency loss for better sampling while retaining the VTO abilities. This problem can then be formulated as:

$$\min_{h\in\mathcal{H}} \mathbb{E}_{\mathbf{y}\sim\mathbf{Y}}[\mathcal{L}_{CM}^{\mathbf{y}}(\mathbf{h}) + \mathcal{L}_{DSM}^{\mathbf{y}}(\mathbf{h})], \tag{18}$$

$$\text{such that:} \tag{19}$$

$$\mathcal{L}_{CM}^{\mathbf{y}}(\mathbf{h}) = \tag{20}$$

$$\mathbb{E}_{(\mathbf{X}_0),\epsilon,t}[\|\mathbf{h}(\mathbf{H}_t, \mathbf{y}) + \mathbf{s}_t^{\theta^*}(\mathbf{H}_t) - \epsilon\|^2] \leq \tag{21}$$

$$\mathbb{E}_{(\mathbf{x}_0),\epsilon,t}[\|\mathbf{h}^{DEFT}(\mathbf{H}_t, \mathbf{y}) + \mathbf{s}_t^{\theta^*}(\mathbf{H}_t) - \epsilon\|^2], \tag{22}$$

where $\mathbf{h}^{DEFT}$ is the minimizer of the DEFT objective $\mathbb{E}_{\mathbf{y}\sim\mathbf{Y}}\mathcal{L}_{DSM}^{\mathbf{y}}(\mathbf{h})$. Vanilla method to solve this problem at scale is minimizing $\mathcal{L}_{CM} + \lambda\mathcal{L}_{DSM}$. However, this vanilla

method risks losing VTO abilities, as the network is able to find easy local minima with small consistency loss $\mathcal{L}_{CM}$. We generalize on this loss and propose to minimize

$$\mathcal{L}_{adaptive} = \lambda_{DSM1} \max(\mathcal{L}_{DSM}, b_1) \quad (23)$$
$$+ \lambda_{DSM2} \min(\mathcal{L}_{DSM}, b_2) + \lambda_{CM}\mathcal{L}_{CM}, \quad (24)$$

where $\lambda_{CM}$, $\lambda_{DSM1}$, $\lambda_{DSM2}$, $b_1$, and $b_2$ are constants. The h-transform network trains for better VTO abilities when the VTO performance on a data point is lackluster, while focusing on the consistency objective when the VTO performance is good. Empirical results show that this adaptive loss helps preserve VTO performances while accelerating the sampling process.

**Algorithm 1** Adaptive training algorithm for $\mathbf{x}_0$ prediction

**Input:** pre-trained encoder $\varepsilon$, pre-trained score network $\mathbf{s}$, initialized h-transform network $\mathbf{h}$, garment agnoistic images, mask images, reference images, and target images $\{\mathbf{I}_a^{(i)}, \mathbf{I}_M^{(i)}, \mathbf{I}_g^{(i)}, \mathbf{I}_p^{(i)}\}_{i=1}^N$, coefficients $\lambda_{DSM1}, \lambda_{DSM2}, \lambda_{CM}$, threshold $b_1$, $b_2$.
Freeze weights of $\varepsilon$ and $\mathbf{s}$
**while** Not converged **do**
  **for** $i \in 1, \ldots, N$ **do**
    Interpolate image space mask $\mathbf{I}_M$ to get latent space mask $\mathbf{I}_m$.
    Encode input $\mathbf{x}_0^{(i)} = (\varepsilon(\mathbf{I}_a) \oplus \varepsilon(\mathbf{I}_g)) \otimes (\mathbf{I}_m \oplus \mathbb{O}) \otimes (\varepsilon(\mathbf{I}_p))$
    Sample noise level $\{t_j\}_{j=1}^{N_t} \sim \text{Unif}(0, 1)$.
    Sample noise $\mathbf{W} \sim \mathcal{N}(0, \mathbf{I})$.
    Inject noise $\{\mathbf{x}_{t_j}^i = \sqrt{\bar{\alpha}_{t_j}}\mathbf{x}_0^{(i)} + \sqrt{1-\bar{\alpha}_{t_j}}\mathbf{W}\}_{j=1}^{N_t}$.
    Infer $\{\hat{\mathbf{x}}_0^{(i,j)} = \mathbf{s}(t, \mathbf{x}_{t_j}^{(i)}) + h(t, \mathbf{x}_{t_j}^{(i)})\}_{j=1}^{N_t}$.
    Compute $\mathcal{L}_{DSM} = \frac{1}{N_t}\sum_{j=1}^{N_t}$ P-Huber$(\hat{\mathbf{x}}_0^{(i,j)} - \mathbf{x}_0^{(i)})$, $\mathcal{L}_{CM} = \frac{1}{N_t - 1}\sum_{j=1}^{N_t - 1}$ P-Huber$(\hat{\mathbf{x}}_0^{(i,j)} - \hat{\mathbf{x}}_0^{(i,j+1)})$.
    Return loss $\lambda_{DSM1} \max(\mathcal{L}_{DSM}, b_1) + \lambda_{DSM2} \min(\mathcal{L}_{DSM}, b_2) + \lambda_{CM}\mathcal{L}_{CM}$ for optimization.
  **end for**
**end while**
**Return:** trained h-transform network $\mathbf{h}$.

# 4. Experiment

## 4.1. Datasets

**Virtual try-on test dataset** For testing, we conduct experiments using the test-split of the widely recognized public dataset VITON-HD [26] to compare against existing SOTA methods on blending image conditions into reference images in the VTO task. The VITON-HD dataset test-split is composed of pairs of reference garment images, source images, and preprocessed mask and pose images. The source image serves as the ground-truth label, which is composed of the model wearing the item from the reference garment image. The VITON-HD test split includes 2,032 pairs of upper-body garment in front-facing poses. For our testing experiments, we reuse the preprocessed mask images provided in the dataset.

**Virtual try-all dataset** In order to perform the try-on task beyond the front-facing garment try-on and commonly used garment categories like shirts and pants, as is in VITON-HD, we use a Virtual Try-All (VTA) dataset with diverse clothing and product categories, as well as novel object scenarios. The dataset includes 1) Expanded clothing categories, including jackets, pajamas, and more; 2) Clothing and non-clothing. Clothing include bags, hats, and shoes, non-clothing include jewelry, toys, and more. 3) More complicated source and reference images. Unlike existing datasets, reference images in the VTA dataset exhibits variety of object scenarios; for instance, the garment can be shown flat (lay-flat) instead of on a model in 3D form (also known as ghost-mannequin), it can also be shown being worn (on-figure) or not (off-figure).

**Data cleaning and preparation** To pick the right image pairs, we use a classifier to check if the source image is displaying the product in the reference image. To generate accurate masks, we first apply an object detection model to generate the bounding box area of the product in the source image, and then use the bounding box to prompt a segmentation model to create the inpainting mask for the source image.

## 4.2. Implementation details

We use a Latent Diffusion model as our pre-trained model backbone. The model consists of a pre-trained autoencoder and score diffusion U-Net. We freeze these networks and train an h-transform network . During inference, we use a DDIM sampling method adapted to consistency models (for similar methods, see e.g. [19]). We perform training on 8 NVIDIA H100 GPUs.

## 4.3. Evaluation metrics

The evaluation is conducted in a paired and unpaired setting. In the paired setting, the original garment in the source image is the same as the garment provided in the reference image. The source image is served as the ground truth image. We use SSIM [49] and LPIPS [50] to compare the ground truth with the generated results. In an unpaired image, the garment in the source image is a different garment from the referent image. We compute the distribution distance between the source image and the generated image with FID [50] and KID [51] score. We follow the implementation in [52].

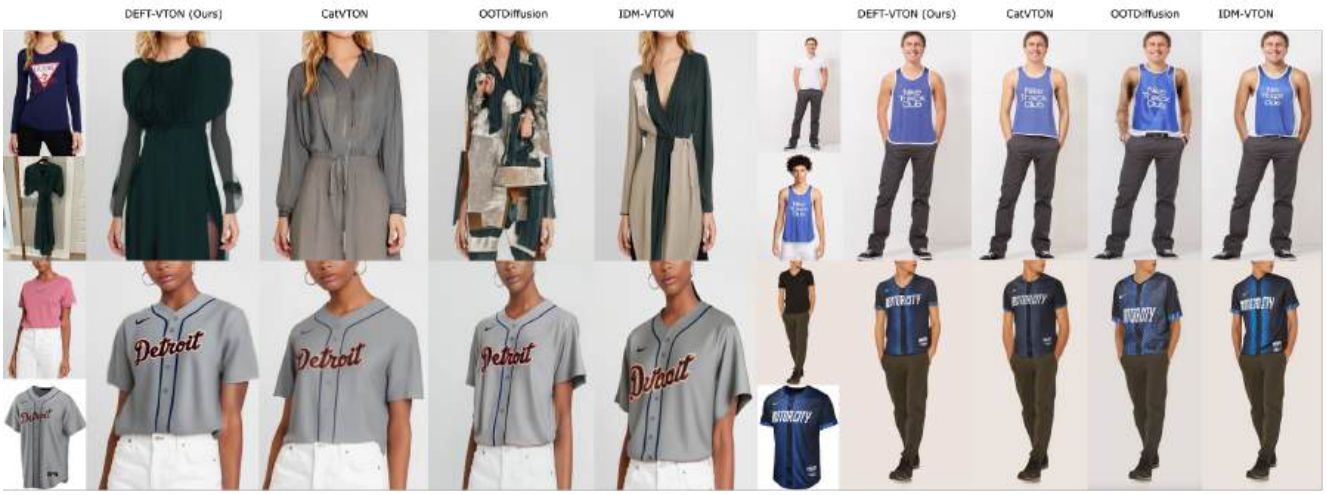


Figure 6. Comparison of our DEFT-VTON (15 sampling steps) with CatVTON (50 sampling steps) [34], OOTDiffusion (20 sampling steps) [47], and IDM-VTON (30 sampling steps) [48].

### 4.4. Qualitative results

We provide four semantic examples in Figure 6 on DEFT-VTON's performances with only 15 sampling steps, and compare to that of SOTA models, CatVTON with 50 sampling steps, OOTDiffusion with 20 sampling steps, and IDM-VTON with 30 sampling steps [34, 47, 48]. We further compare DEFT-VTON's performances with existing SOTA models' performances on complex multi-garment try-on tasks in Figure 7. DEFT-VTON achieves the best performance across different models and different tasks, further confirming its performance boosts.

### 4.5. Quantitative Results

#### 4.5.1. Comparison with different model configurations

We first study which model configurations work best for the VTO task.

We compare our approach with state-of-the-art methods. As shown in Table 1, DEFT-VTON significantly outperforms the established VTO models across all metrics on the VITON-HD test dataset. DEFT-VTON achieves SOTA performances compared to existing works, while using less function evaluations. The improvement is significant, especially on the SSIM, FID, and the KID metric, showing that our model better maintains the structural information in the image. We show more comprehensive evaluations on VITON-HD test dataset in the Supplementary Material.

#### 4.5.2. Number of function evaluations

We perform an ablation study of DEFT-VTON across different sampling steps on the VITONHD dataset. The evaluated scores stabalizes after 12 steps, highlighting consistency fine tuned DEFT-VTON's efficiency in the sampling process. We attach details to the ablation study in the Supplementary Material.

| Models | VITON-HD | | | |
|---|---|---|---|---|
| | SSIM ↑ | LPIPS ↓ | FID ↓ | KID ↓ |
| StableVTON | 0.8543 | 0.0905 | 11.054 | 3.914 |
| LaDI-VTON | 0.8603 | 0.0733 | 14.648 | 8.754 |
| IDM-VTON | 0.8499 | 0.0603 | 9.842 | 1.123 |
| OOTDiffusion | 0.8187 | 0.0876 | 12.408 | 4.680 |
| CatVTON | 0.8704 | 0.0565 | 9.015 | 1.091 |
| DEFT, $\mathcal{L}_{DSM}$, 25 steps | <u>0.9118</u> | <u>0.0533</u> | **8.3351** | **0.5212** |
| DEFT, $\mathcal{L}_{DSM}$, 15 steps | 0.9098 | **0.0521** | 8.6339 | 0.7916 |
| DEFT, $\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 25 st. | 0.9105 | 0.0564 | <u>8.6310</u> | <u>0.7243</u> |
| DEFT, $\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 15 st. | **0.9130** | 0.0542 | 8.8567 | 1.016 |

Table 1. Results of baseline comparisons with DEFT-VTON [34, 47, 48, 53, 54]. Bold texts indicate best models, underlined texts indicate second best models. We provide results on the dresscode dataset in the appendix.

### 4.6. Ablation study on adaptive coefficients

We perform an ablation study on the adaptive balancing coefficients between the h-transform loss and the consistency loss. We use the same threshold for triggering a higher focus on $\mathcal{L}_{DSM}$ and the same coefficients for $\mathcal{L}_{CM}$, and ablate to study the impact of the coefficients of the coefficients for $\mathcal{L}_{DSM}$. As shown in Figure 8, consistency loss increases and DSM loss decreases as we increase the coefficient. We find setting the coefficient at 0.6 to achieve the best balance between $\mathcal{L}_{CM}$ and $\mathcal{L}_{DSM}$.


Figure 7. Comparison of DEFT-VTON with SOTA models on complex multi-garment try-on task.

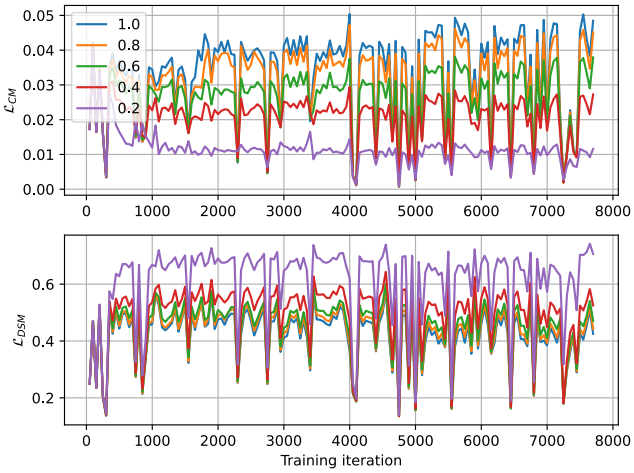


Figure 8. $\mathcal{L}_{CM}$ and $\mathcal{L}_{DSM}$ corresponding to different adaptive training coefficients. We use the same threshold for triggering a higher focus on $\mathcal{L}_{DSM}$ and ablate its coefficients.

## 5. Conclusion

In this paper, using a pre-trained diffusion transformer as backbone, we explore Doob's h-transform efficient finetuning (DEFT) and consistency training for virtual try-on tasks. Compared to existing VTO frameworks, the proposed DEFT-VTON completely freezes the backbone network, and trains an auxilary network with number of parameters as low as $1.42\%$ of the backbone network. Being able to train auxiliary networks allows for efficient adaptations to ever-improving SOTA unconditional models regardless of their model sizes. To further accelerate the sampling process while preserving the VTO ability, we explore an adaptive balancing between the DEFT loss and the consistency loss. Ablation study shows DEFT finetuning leads to SOTA VTO performances, while the consistency finetuning accelerates the sampling process for up to $40\%$ while retaining the same performances. Our model outperforms existing SOTA models both qualitatively and quantitatively, while requiring much less training cost.

## 6. Limitation

While our proposed DEFT-VTON framework exhibits SOTA performances, as the h-transform network is only $1.42\%$ of the pre-trained network, it relies heavily on the pre-trained unconditional model. The model might suffer from the same limitations as the base model, with any artifact or failure pattern observed in the unconditional model generally propagating into the DEFT-VTON model.

## References

[1] Mehmet Saygin Seyfioglu, Karim Bouyarmane, Suren Kumar, Amir Tavanaei, and Ismail B. Tutar. Diffuse to choose: Enriching image conditioned inpainting in latent diffusion models for virtual try-all, 2024. 1, 2, 3

[2] Dan Song, Xuanpu Zhang, Juan Zhou, Weizhi Nie, Ruofeng Tong, Mohan Kankanhalli, and An-An Liu. Image-based virtual try-on: A survey, 2024.

[3] Mehmet Saygin Seyfioglu, Karim Bouyarmane, Suren Kumar, Amir Tavanaei, and Ismail B Tutar. Dreampaint: Few-shot inpainting of e-commerce items for virtual try-on without 3d modeling. *arXiv preprint arXiv:2305.01257*, 2023. 1

[4] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. 1

[5] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2021.

[6] 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. 1, 3

[7] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021. 1, 3

[8] Hyungjin Chung, Jeongsol Kim, Michael T. Mccann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems, 2024. 1, 2

[9] Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*, 2023. 1

[10] Morteza Mardani, Jiaming Song, Jan Kautz, and Arash Vahdat. A variational perspective on solving inverse problems with diffusion models, 2023. 1, 2, 3

[11] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. 3

[12] Georgios Batzolis, Jan Stanczuk, Carola-Bibiane Schönlieb, and Christian Etmann. Conditional image generation with score-based diffusion models, 2021.

[13] Guan-Horng Liu, Arash Vahdat, De-An Huang, Evangelos A. Theodorou, Weili Nie, and Anima Anandkumar. $I^2$sb: Image-to-image schrödinger bridge, 2023. 1

[14] Alexander Denker, Francisco Vargas, Shreyas Padhy, Kieran Didi, Simon Mathis, Vincent Dutordoir, Riccardo Barbano, Emile Mathieu, Urszula Julia Komorowska, and Pietro Lio. Deft: Efficient fine-tuning of diffusion models by learning the generalised $h$-transform, 2024. 1, 2, 4, 5

[15] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models, 2022. 2

[16] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. 2, 4

[17] Qinsheng Zhang and Yongxin Chen. Fast sampling of diffusion models with exponential integrator, 2023.

[18] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models, 2022. 2, 4

[19] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models, 2023. 2, 5, 6

[20] Yang Song and Prafulla Dhariwal. Improved techniques for training consistency models, 2023. 2, 5

[21] Giannis Daras, Yuval Dagan, Alexandros G. Dimakis, and Constantinos Daskalakis. Consistent diffusion models: Mitigating sampling drift by learning to be consistent, 2023. 2, 5

[22] Dongjun Kim, Chieh-Hsin Lai, Wei-Hsiang Liao, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yutong He, Yuki Mitsufuji, and Stefano Ermon. Consistency trajectory models: Learning probability flow ode trajectory of diffusion, 2024. 2

[23] Zhengyang Geng, Ashwini Pokle, William Luo, Justin Lin, and J. Zico Kolter. Consistency models made easy, 2024. 2, 5

[24] Xintong Han, Zuxuan Wu, Zhe Wu, Ruichi Yu, and Larry S. Davis. Viton: An image-based virtual try-on network, 2018. 2

[25] Shuai Bai, Huiling Zhou, Zhikang Li, Chang Zhou, and Hongxia Yang. Single stage virtual try-on via deformable attention flows, 2022.

[26] 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. 6

[27] 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 '23, page 7599–7607. ACM, October 2023.

[28] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan, 2020.

[29] Sangyun Lee, Gyojung Gu, Sunghyun Park, Seunghwan Choi, and Jaegul Choo. High-resolution virtual try-on with misalignment and occlusion-handled conditions, 2022.

[30] Kathleen M Lewis, Srivatsan Varadharajan, and Ira Kemelmacher-Shlizerman. Tryongan: Body-aware try-on via layered interpolation, 2021.

[31] 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, 2023.

[32] Ben Xue, Shenghui Ran, Quan Chen, Rongfei Jia, Binqiang Zhao, and Xing Tang. Dccf: Deep comprehensible color filter learning framework for high-resolution image harmonization, 2022.

[33] Keyu Yan, Tingwei Gao, Hui Zhang, and Chengjun Xie. Linking garment with person via semantically associated landmarks for virtual try-on. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17194–17204, 2023.

[34] Zheng Chong, Xiao Dong, Haoxiang Li, Shiyue Zhang, Wenqing Zhang, Xujie Zhang, Hanqing Zhao, and Xiaodan Liang. Catvton: Concatenation is all you need for virtual try-on with diffusion models, 2024. 2, 3, 7

[35] Jeongho Kim, Gyojung Gu, Minho Park, Sunghyun Park, and Jaegul Choo. Stableviton: Learning semantic correspondence with latent diffusion model for virtual try-on, 2023.

[36] Qi Li, Shuwen Qiu, Julien Han, Kee Kiat Koo, and Karim Bouyarmane. Dit-vton: Diffusion transformer framework for unified multi-category virtual try-on and virtual try-all with integrated image editing. *https://dit-vton.github.io/DiT-VTON/*, 2024.

[37] Shuwen Qiu, Qi Li, Julien Han, Kee Kiat Koo, and Karim Bouyarmane. Is concatenation really all you need: Efficient concatenation-based pose conditioning and pose con-

trol for virtual try on. *https://pose-vton.github.io/vto-pose-conditioning/*, 2024.

[38] Julien Han, Shuwen Qiu, Qi Li, Xingzi Xu, Kavosh Asadi, and Karim Bouyarmane. Instructvton: Optimal auto-masking and natural-language-guided interactive style control for inpainting-based virtual try-on. *https://instructvton.github.io/instruct-vton.github.io/*, 2024. 2

[39] Liangchen Li and Jiajun He. Bidirectional consistency models, 2024. 2

[40] Ling Yang, Zixiang Zhang, Zhilong Zhang, Xingchao Liu, Minkai Xu, Wentao Zhang, Chenlin Meng, Stefano Ermon, and Bin Cui. Consistency flow matching: Defining straight flows with velocity consistency, 2024. 2

[41] Hyungjin Chung, Byeongsu Sim, Dohoon Ryu, and Jong Chul Ye. Improving diffusion models for inverse problems using manifold constraints, 2024. 2

[42] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models, 2022. 4

[43] Jeremy Heng, Valentin De Bortoli, Arnaud Doucet, and James Thornton. Simulating diffusion bridges with score matching, 2022. 4

[44] L. C. G. Rogers and David Williams. *Diffusions, Markov Processes and Martingales*. Cambridge Mathematical Library. Cambridge University Press, 2 edition, 2000. 4

[45] Simian Luo, Yiqin Tan, Suraj Patil, Daniel Gu, Patrick von Platen, Apolinário Passos, Longbo Huang, Jian Li, and Hang Zhao. Lcm-lora: A universal stable-diffusion acceleration module, 2023. 5

[46] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference, 2023. 5

[47] Yuhao Xu, Tao Gu, Weifeng Chen, and Chengcai Chen. Ootdiffusion: Outfitting fusion based latent diffusion for controllable virtual try-on. *arXiv preprint arXiv:2403.01779*, 2024. 7

[48] Yisol Choi, Sangkyung Kwak, Kyungmin Lee, Hyungwon Choi, and Jinwoo Shin. Improving diffusion models for authentic virtual try-on in the wild. *arXiv preprint arXiv:2403.05139*, 2024. 7

[49] 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. 6

[50] 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. 6

[51] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018. 6

[52] 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*, pages 8580–8589, 2023. 6

[53] Davide Morelli, Alberto Baldrati, Giuseppe Cartella, Marcella Cornia, Marco Bertini, and Rita Cucchiara. Ladi-vton: Latent diffusion textual-inversion enhanced virtual try-on, 2023. 7

[54] 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. 7

# DEFT-VTON: Efficient Virtual Try-On with Consistent Generalised H-Transform

Supplementary Material

## 1. Ablation study

We provide ablation studies on the coefficients of $\mathcal{L}_{DSM}$ as well as the number of steps.

**$\mathcal{L}_{DSM}$ coefficient** While the table shows better SSIM and LPIPS scores for fewer steps (15) with changing $\mathcal{L}_{DSM}$ coefficients, the FID and KID scores, which assess perceptual quality, are worse. Empirically, we observe that the consistency finetuned models better preserve garment colors and complex text/graphics, as shown in Figure 1, on some challenging tasks involving complex text/graphics preservation, unclear reference images, and tasks that are rare in the training dataset, 15 steps sampling of the consistency finetuned model qualitatively outperforms the one only finetuned with DEFT loss, reaffirming the quantitative observation that consistency finetuning improves sampling with fewer steps.

We also observe that, when the coefficient on $\mathcal{L}_{DSM}$ is overly small, the DEFT-VTON model loses its VTO abilities.

**Number of sampling steps** Table 1 shows that the SSIM and LPIPS scores exhibit an initial rise and subsequent decline as sampling steps increase in the consistency finetuned DEFT-VTON model, implying an optimal performance at approximately 15 steps. We also observe that the FID and KID score consistently improve as we increase the number of sampling steps. Although this shows improvements in human perception, it does not always translate to better VTO results, with rising cases of hallucinations.

| Models | VITON-HD | | | |
|---|---|---|---|---|
| | SSIM ↑ | LPIPS ↓ | FID ↓ | KID ↓ |
| StableVTON | 0.8543 | 0.0905 | 11.054 | 3.914 |
| LaDI-VTON | 0.8603 | 0.0733 | 14.648 | 8.754 |
| IDM-VTON | 0.8499 | 0.0603 | 9.842 | 1.123 |
| OOTDiffusion | 0.8187 | 0.0876 | 12.408 | 4.680 |
| CatVTON | 0.8704 | 0.0565 | 9.015 | 1.091 |
| DEFT, $\mathcal{L}_{DSM}$, 25 steps | 0.9118 | <u>0.0533</u> | **8.3351** | **0.5212** |
| DEFT, $\mathcal{L}_{DSM}$, 15 steps | 0.9098 | **0.0521** | 8.6339 | 0.7916 |
| DEFT, $0.2\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 12 steps | 0.9063 | 0.0782 | 25.7948 | 15.52 |
| DEFT, $0.2\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 15 steps | 0.9064 | 0.0798 | 27.8843 | 18.92 |
| DEFT, $0.2\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 25 steps | 0.9034 | 0.0853 | 33.2223 | 24.53 |
| DEFT, $0.4\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 12 steps | <u>0.9135</u> | 0.0553 | 9.1671 | 1.2612 |
| DEFT, $0.4\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 15 steps | **0.9136** | 0.0552 | 8.7922 | 0.9970 |
| DEFT, $0.4\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 25 steps | 0.9098 | 0.0581 | <u>8.4704</u> | 0.6649 |
| DEFT, $0.6\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 12 steps | 0.9122 | 0.0560 | 9.0136 | 1.1998 |
| DEFT, $0.6\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 15 steps | 0.9132 | 0.0553 | 8.6611 | 0.9104 |
| DEFT, $0.6\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 25 steps | 0.9100 | 0.0579 | 8.5988 | 0.6713 |
| DEFT, $0.8\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 12 steps | 0.9112 | 0.0571 | 9.0765 | 1.2336 |
| DEFT, $0.8\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 15 steps | 0.9126 | 0.0561 | 8.7394 | 0.9378 |
| DEFT, $0.8\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 25 steps | 0.9101 | 0.0592 | 8.4160 | <u>0.5474</u> |
| DEFT, $1.0\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 10 steps | 0.8935 | 0.0862 | 12.7324 | 3.5322 |
| DEFT, $1.0\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 11 steps | 0.9111 | 0.0573 | 9.2023 | 1.3684 |
| DEFT, $1.0\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 12 steps | 0.9114 | 0.0571 | 8.9859 | 1.1134 |
| DEFT, $1.0\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 15 steps | 0.9125 | 0.0561 | 8.7113 | 0.8937 |
| DEFT, $1.0\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 20 steps | 0.9081 | 0.0617 | 8.8301 | 0.8030 |
| DEFT, $1.0\mathcal{L}_{DSM} + \mathcal{L}_{CM}$, 25 steps | 0.9091 | 0.0599 | 8.5058 | 0.5952 |

Table 1. Results of baseline comparisons with DEFT-VTON on VITON-HD dataset. Bold texts indicate best models, underlined texts indicate second best models.

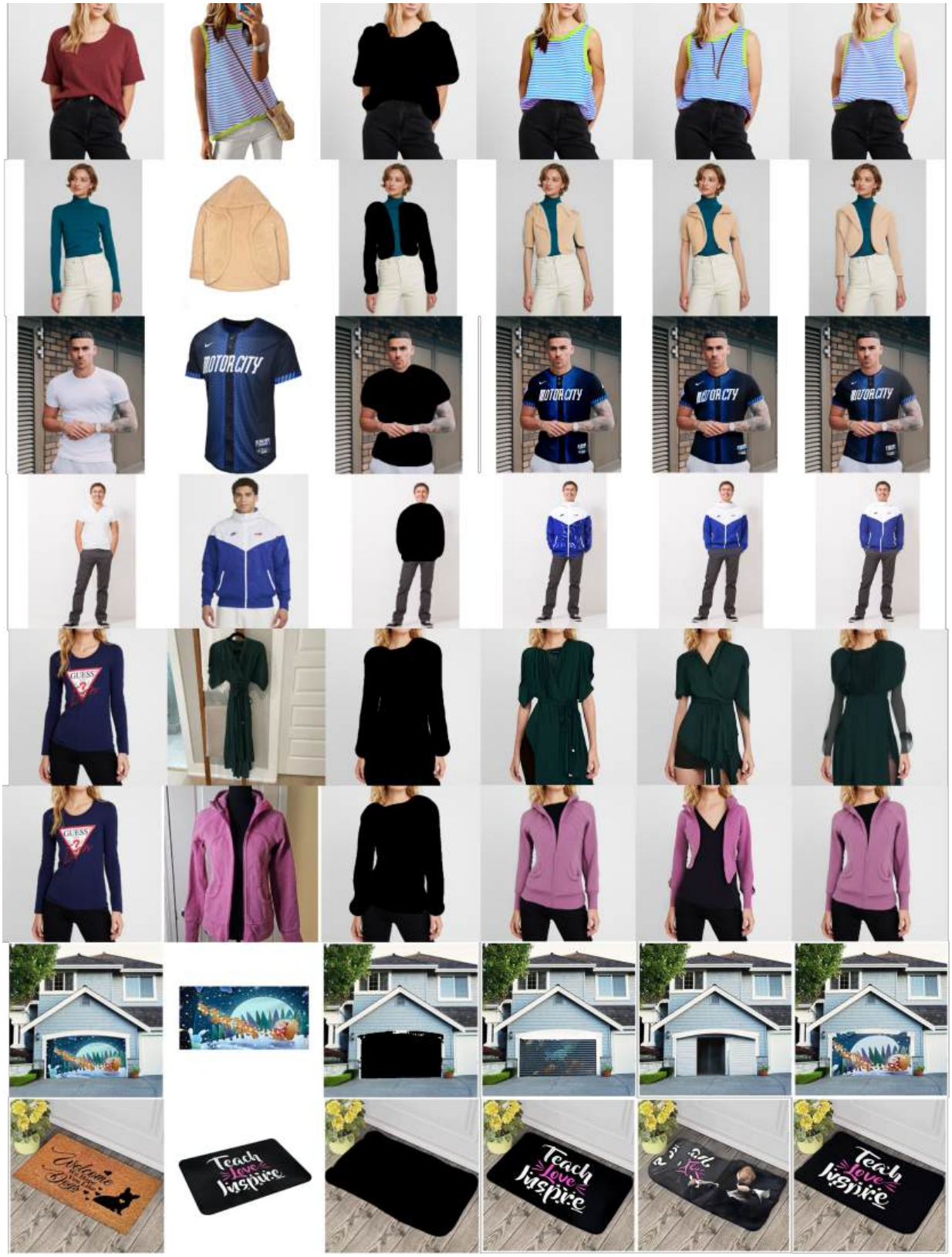


Figure 1. Results from test cases involving complex text and graphics preservation, unclear reference images, and unusual tasks. Each row shows a single task, starting with the original image and progressing through garment image, masked original image, 25-step sampling results (before consistency fine tuning), 15-step sampling results (before consistency fine tuning), and finally, 15-step sampling results (after consistency fine tuning).