Generation with Nuanced Changes: Continuous Image-to-Image Translation with Adversarial Preferences

Yinghua Yao, Yuangang Pan, Ivor W. Tsang, *Fellow, IEEE,* and Xin Yao, *Fellow, IEEE*

***Abstract*—Most previous methods for continuous image-to-image translation resorted to binary attributes with restrictive description ability and thus cannot achieve satisfactory performance. Some works proposed to use fine-grained semantic information, *Relative Attributes (RAs), preferences over pairs of images on the strength of a specified attribute*. However, they still failed to reconcile both goals for smooth translation and for high-quality generation simultaneously. In this work, we propose a new model CTAP to coordinate these two goals for high-quality continuous translation based on RAs. In CTAP, we simultaneously train two modules: a generator that translates an input image to the desired image with smooth nuanced changes w.r.t. the interested attributes; and a ranker that executes adversarial preferences consisting of the input image and the desired image. Particularly, adversarial preferences involve an adversarial ranking process: (1) the ranker thinks no difference between the desired image and the input image in terms of the interested attributes; (2) the generator fools the ranker to believe the attributes of its output image changes as expect compared to the input image. RAs over pairs of real images are introduced to guide the ranker to rank image pairs regarding the interested attributes only. With an effective ranker, the generator would "win" the adversarial game by producing high-quality images that present smooth changes. The experiments on two face datasets and one shoe dataset demonstrate that our CTAP achieves state-of-art results in generating high-fidelity images which exhibit smooth changes over the interested attributes.**

***Impact Statement*—Continuous image-to-image translation has gained increasing attention with its wide applications such as film production [1], aiding in medical diagnosis [2]. Unlike many existing works that rely on binary attributes, capturing only category-level differences, we utilize Relative Attributes (RAs), which can describe subtle distinctions in attributes. More importantly, we propose a new framework equipped with adversarial training between a ranker and a generator to reconcile the goal for continuous translation and the goal for high-quality generation under the context of RAs. This addresses the research gap that the attribute models of previous approaches are only trained using RAs from real data and do not generalize well to unseen generated data, resulting in either failure of high-quality generation or failure of smooth translation. In addition, our proposed method establishes a new adversarial paradigm by extending the adversarial game on classification to ranking.**

***Index Terms*—Adversarial Preferences, Continuous Image-to-image Translation, Generative Adversarial Network (GAN), Relative Attributes.**

This work was supported by the Agency for Science, Technology and Research (A*STAR) Centre for Frontier AI Research, the A*STAR GAP project (Grant No. I23D1AG079), NSFC (Grant No. 62250710682), and the Program for Guangdong Provincial Key Laboratory (Grant No. 2020B121201001).
Yinghua Yao, Yuangang Pan and Ivor W. Tsang are with the Centre for Frontier AI Research, Agency for Science, Technology and Research (A*STAR), 138632, Singapore and also with the Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), 138632, Singapore. Xin Yao is with the School of Data Science, Lingnan University, Hong Kong and also with the School of Computer Science, University of Birmingham, B15 2TT Birmingham, UK. Email: eva.yh.yao@gmail.com, yuangang.pan@gmail.com, ivor.tsang@gmail.com, xinyao@ln.edu.hk. (Corresponding author: Ivor W. Tsang and Xin Yao)

I. Introduction

IMAGE-TO-IMAGE (I2I) translation [3] aims to translate an input image into the desired ones with changes in some specified attributes. Current literature can be classified into two categories: binary translation [4], [5], [6], [7], e.g., translating an image from "no smile" to "smile"; continuous translation [8], [9], e.g., generating a series of images with smooth changes from "no smile" to "smile". In this work, we focus on high-quality continuous I2I translation, namely, generating a series of realistic versions of the input image with smooth changes on the specified attributes. The desired high-quality images in our context are two folds: first, the generated images look as realistic as training images, named as *realistic quality*; second, the generated images are modified exclusively in terms of the specified attributes, named as *attribute-specified quality*.

Most existing methods [8], [9], [10], [11], [12], [13], [14], [15] resort to binary attributes that indicate the presence (or absence) of certain attributes in an image. However, their continuous I2I translation is unsatisfactory as the binary attributes have restrictive description capacity [16], [17]. Instead, relative attributes (RAs), referring to the preferences of pairs of images w.r.t. the strength of the interested attributes, have more potentials in continuous translation because of its richer semantic information [16]. For instance, in terms of the "smile" attribute, it is coarse to simply classify whether a face image is either smiling or not smiling since there exist many images that are difficult to categorize. The comparison between a pair of image in terms of the "smile" attribute can describe more subtle attribute changes, such as the minor variation of the smile extent. We compare continuous image-to-image translation based on binary attributes and relative attributes in Fig. 1. It shows that with relative attributes, (1) the alterations are more evenly distributed throughout the image sequences; (2) the changes are only specified to the region regarding the "smile" attribute.

Existing works based on RAs [17], [18] for continuous I2I translation proposed their model within the framework of generative adversarial networks (GANs) [19] targeting for the following two goals: the goal of interpolating RAs on generated images for continuous translation and the goal for

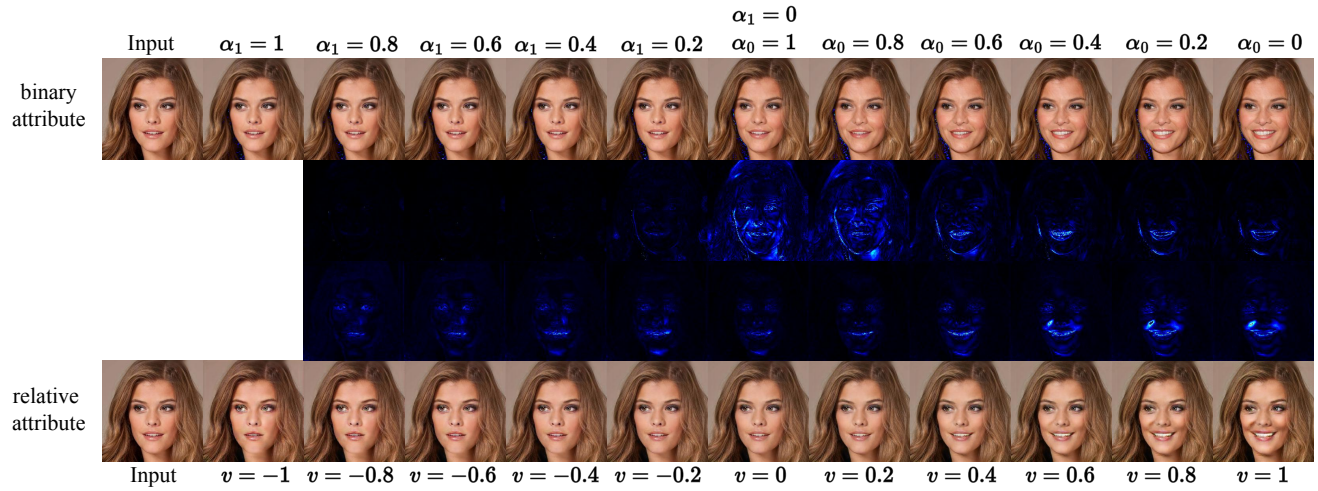


Fig. 1. Binary attribute (VecGAN [8]) vs. relative attribute (our CTAP) for continuous I2I translation (from left to right: larger "smile"). The second and third row are heat map to visualize the difference between two adjacent translated images in the first and fourth row, respectively. We expect the alterations are evenly distributed throughout the image sequences.

good-quality generation. However, these methods suffer from the conflict between the two goals. Specifically, the methods model the RAs of real images by capturing the nuanced difference over the attributes using an extra attribute model, e.g., a classification/ranking model, and count on the learned attribute model to generalize the learned attribute preferences to generated images using interpolated RAs. However, the learned attribute model never functions as expected since it generalizes poorly to unseen generated images, making the generation deviate from the real data distribution. In a trade-off with the quality goal enforced by a discriminator, the generated images either fail to change smoothly over the interested attributes or suffer from low-quality issues (See results in Section IV-A).

In this paper, we propose a new framework equipped with adversarial training between a tailor-designed ranker and a generator for high-quality continuous translation (CTAP). In particular, CTAP introduces the generated image into the training of the ranker, which distills the discrepancy from RAs via ranking loss, and formulates an adversarial ranking process using adversarial preferences (Eq. (4)). Specifically, the ranker keeps ignoring the relative changes between the generated image and the input image regarding the interested attributes; while the generator aims to achieve the agreement from the ranker that the generated image holds the desired difference compared to the input. Competition between the ranker and the generator drives both two modules to improve themselves. Since our ranker can critic the generated images during its whole training process, and it no doubt can generalize well to generated images to ensure reliable fine-grained control over the target attributes. Contributions are summarized as follows:

- We propose a method named Continuous Translation via Adversarial Preferences (CTAP) consisting of a ranker and a generator for a high-quality continuous translation. This is the first generative framework that reconciles the goal for continuous translation and the goal for high-quality generation under the context of relative attributes.
- Our CTAP broadens the principle of adversarial training by extending the adversarial game on classification to ranking via adversarial preferences. The adversarial preferences evoke the adversarial training between the ranker and the generator, which enhances the generalization of the ranker to the generated data and encourages a better generator.
- Our ranker enforces a linearizedly continuous change between the generated image and the input image, which promotes a better fine-grained control over the interested attributes.
- Empirical results show that our CTAP achieves the state-of-art results on the high-quality continuous I2I translation task. We further extend CTAP to the continuous I2I translation of multiple attributes. A case study demonstrates the efficacy of our CTAP in terms of disentangling multiple attributes and manipulating them simultaneously.

II. Related Works

A. Binary Attributes based

Many studies on continuous I2I translation employ the interpolation of binary attributes, widely developed in auto-encoder-based frameworks [10], [11], [12], [20], flow-based frameworks [13], generative adversarial networks (GANs) [21], [8], [9], [14], [15], diffusion models [22]. However, the interpolation quality is unsatisfactory since the models are only trained on binary-valued attributes and thus the interpolation is ill-defined [17], [23]. Methods that apply regularization of interpolation consistency, wherein interpolated images are predicted with corresponding interpolated attribute values [24], [8], obtain better interpolation quality. But they are still inferior due to the limited description capacity of binary attributes. Please note VecGAN [8] is the state-of-art work in this line of methods (included in the comparison in Fig. 1), which failed to obtain promising results on smooth translation.

Another line of research leverages pretrained deep generative models for image attribute manipulation [25], [26], [27], [28]. They learn interpretable latent directions based on binary attributes and each single attribute of the image can be modified by movement along a specified direction. However, this kind of methods require inverting a given image back into the latent space of the pretrained generative model, which is challenging and could be imperfect, resulting in poor translation

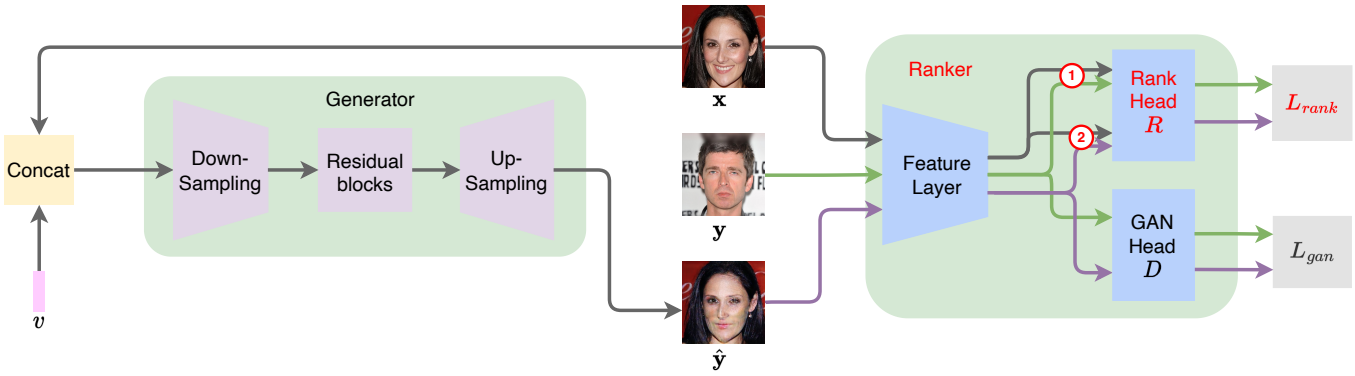


Fig. 2. The network structure of CTAP. ① denotes real image pairs $(\mathbf{x}, \mathbf{y})$ (Fig. 3) and ② denotes generated image pairs $(\mathbf{x}, \hat{\mathbf{y}})$ (Fig. 4).

performance [29]. In addition, relevant studies have identified the existence of reconstruction-editability trade-offs in these methods [8], [30].

B. Other Priors based

Deng *et al.* [31] embedded 3D priors into adversarial learning. However, it relies on available priors for attributes, which limits the practicality. Do *et al.* [32] utilized 3D-based pretrained GANs for fine-grained manipulation of attributes, the performance of which is restricted by that of the 3D GANs. Alharbi and Wonka [33] proposed an unsupervised disentanglement method. It controls specified parts of the generated images by injecting structure noises, which makes global or local features changed in a disentangled way. However, it is unclear how global or local features are related to facial attributes. Thus, it cannot meet the requirement of changing specified attributes. Some studies [34], [35], [36] achieved fine-grained facial expression generation via Action Units (AU) annotations, which describe the anatomical facial movements for a human expression in a continuous manifold. However, such a prior for AU is not always available for all applications.

C. Relative Attributes based

Two works applied GAN as a base for continuous I2I translation using relative attributes, which can generate images with nuanced changes on the interested attributes. The main differences lie in the strategies of incorporating the preference over the attributes into the image generation process. RC-GAN [18] adopts two critics consisting of a ranker, learning from the relative attributes of real images, and a discriminator, ensuring the image quality. Then the combination of two critics is aimed to guide the generator to produce high-quality images while interpolating RAs on the generation. However, since the ranker is only trained with real images and generalizes poorly to generated images with interpolated RAs, the ranker would induce the generator to generate out-of-data manifold images which is opposite to the target of the discriminator, resulting in poor-quality images. RelGAN [17] applies a matching-aware discriminator, which learns the joint data distribution of the triplet constructed with a pair of images and a discrete numerical label (i.e., relative attribute)[1]. In order to enable a smooth translation, RelGAN further interpolates the RAs on generated images while enforcing a constraint for the quality of interpolated images. However, such a constraint destroys smooth interpolation (Fig. 9).

Our model is designed within the framework of GAN by utilizing RAs. Therefore, we consider the two most related work RCGAN [18] and RelGAN [23] as our baselines.

III. CTAP for Continuous I2I translation

In this section, we propose a new model, named Translation via Adversarial Preferences (CTAP) for high-quality continuous image-to-image (I2I) translation via relative attributes (RAs). In general, RA is annotated in a discrete way, namely, $r \in \{+1, 0, -1\}$. Because human usually judges, e.g., one face is more smiling (annotated as +1) or similarly smiling (annotated as 0) or less smiling (annotated as -1) than another, but not by how much [18], [16].

We open our description with the one-attribute case. The whole structure of CTAP is shown in Fig. 2, which consists of a generator and a ranker.

The generator takes as input an image $\mathbf{x}$ along with a continuous latent variable $v \in [-1, 1]$ that controls the change of the attribute, and outputs the desired image $\hat{\mathbf{y}}$; while the ranker delivers information in terms of image quality and the preference over the attribute to guide the learning of the generator. We implement the generator with a standard encoder-decoder architecture following previous works [14], [17], [37].

The ranker consists of a feature layer with two heads: a rank head and a GAN head. Particularly, the rank head is introduced to model the relative attributes, whose generalization in ranking performance will be further boosted by adversarial preferences. Meanwhile, the rank head will be enforced with a linearizedly continuous change between the generated image and the input image to promote a better fine-grained attribute manipulation. In addition, the GAN head is introduced to improve the generation quality.

In the following, we first detail the design principle of the ranker. Then, we propose our CTAP model and extend it to the multiple-attribute case.

[1]The matching-aware discriminator is modeled as a binary classifier as original GAN [19]. The difference is that the matching-aware discriminator distinguishes real triplets from fake triplets while the discriminator in original GAN distinguishes real images from fake images.

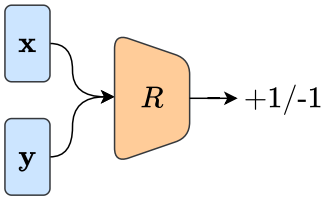


Fig. 3. The ranker model to learn relative attributes for real image pairs.

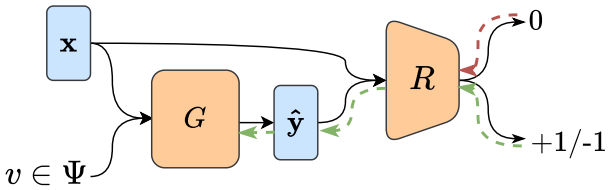


Fig. 4. Adversarial ranking process between the ranker and the generator, which ensures the ranker generalizes well to the generated pairs. Ψ denotes $[-1, 0) \cup (0, 1]$. The green/red dash lines denote gradient back-propagation.

A. Ranker for Relative Attributes

Relative attributes (RAs) are the most representative and effective in describing the information related to the relative emphasis of the attribute, owing to its simplicity and easy construction [16], [18]. For a pair of images $(\mathbf{x}, \mathbf{y})$, RAs $\mathbf{y} \succ \mathbf{x}$ represents that $\mathbf{y}$ shows a greater strength than $\mathbf{x}$ on the target attribute.

Pairwise learning to rank is a widely-adopted technique to model the relative attributes [16]. Given a pair of images $(\mathbf{x}, \mathbf{y})$ and its relative attribute, the pairwise learning to rank technique is formulated as a binary classification [38] (Fig. 3), namely,

$$R(\mathbf{x}, \mathbf{y}) = \begin{cases} +1 & \mathbf{y} \succ \mathbf{x}; \\ -1 & \mathbf{y} \prec \mathbf{x}, \end{cases} \tag{1}$$

where $R(\mathbf{x}, \mathbf{y})$ is the ranking prediction for the image pair $(\mathbf{x}, \mathbf{y})$ (Fig. 3).

Further, the attribute discrepancy implied in RAs, distilled by the ranker, can then be used to guide the generator to translate the input image into the desired one.

However, the ranker is trained on the real image pairs, which only focuses on the modeling of preference over the attribute but ignores image quality. To achieve the agreement with the ranker, the generator possibly produces unrealistic images.

B. Ranking Generalization by Adversarial Preferences

According to the above analysis, we consider incorporating the generated image pairs into the modeling of RAs, along with the real image pairs, to reconcile the goal with respect to ranking and data quality. Consequently, the resultant ranker will not only generalize well to the generated pairs but also avoid providing untrustworthy feedback by distinguishing the unrealistic pairs from the real pairs.

Motivated by adversarial training paradigms [19], [39], we introduce an adversarial ranking process between a ranker and a generator (Fig. 4) to incorporate the generated pairs into the training of ranker. To be specific,

- *Ranker.* Inspired by semi-supervised GAN [40], we assign a pseudo label to the generated pairs. In order to avoid a biased influence on the ranking decision over real image pairs, i.e., positive (+1) or negative (-1), the pseudo label is designed to be zero. Note that a generated pair consists of a synthetic image and its input, facilitating the connection between the ranking prediction and the latent variable. Let Δ be a placeholder that can be either a real image $\mathbf{y}$ or a generated image $\hat{\mathbf{y}}$. Then, we have

$$R(\mathbf{x}, \Delta) = \begin{cases} +1 & (\Delta = \mathbf{y}) \wedge (\mathbf{y} \succ \mathbf{x}); \\ -1 & (\Delta = \mathbf{y}) \wedge (\mathbf{y} \prec \mathbf{x}); \\ 0 & \Delta = \hat{\mathbf{y}}, \end{cases} \tag{2}$$

  where $\hat{\mathbf{y}} = G(\mathbf{x}, v)$ denotes the output of the generator given the input $\mathbf{x}$ and an attribute variable v, which controls the change of attribute for the generated image.
- *Generator.* The goal of the generator is to achieve the consistency between the ranking prediction $R(\mathbf{x}, \hat{\mathbf{y}})$ and the corresponding latent variable v. When $v > 0$, the ranker is supposed to believe that the generated image $\hat{\mathbf{y}}$ has a larger strength of the specified attribute than the input $\mathbf{x}$, i.e., $R(\mathbf{x}, \hat{\mathbf{y}}) = +1$; and vice versa. Namely,

$$R(\mathbf{x}, \hat{\mathbf{y}}) = \begin{cases} +1 & v > 0; \\ -1 & v < 0. \end{cases} \tag{3}$$

According to Eq. (2) and Eq. (3), it is easy to obtain target preferences for the generated pairs $(\mathbf{x}, \hat{\mathbf{y}})$ to train the ranker R and the generator G, respectively:

$$\begin{aligned} &\text{target preference for } R \quad \hat{\mathbf{y}} = \mathbf{x}, \\ &\text{target preference for } G \quad \begin{cases} \hat{\mathbf{y}} \succ \mathbf{x} & v > 0; \\ \hat{\mathbf{y}} \prec \mathbf{x} & v < 0, \end{cases} \end{aligned} \tag{4}$$

which is dubbed as *adversarial preference* since it triggers the opposite goals between the ranker and the generator w.r.t. the generated pairs.

The ranker is promoted by the novel adversarial ranking process in terms of the following aspects: (1) The function of the ranker on the real image pairs is not changed. First, the ranker learns to correctly rank real image pairs by optimizing Eq. (1). Second, the generated image pairs are uniformly sampled regarding their latent variables $v \in [-1, 1]$, which contain positive rankings ($\text{sgn}(v) = +1$, where sgn is the sign function) when conditioned on $v > 0$ and negative rankings ($\text{sgn}(v) = -1$) when conditioned on $v < 0$. By assigning label "0", the ranking output for all generated image pairs can thus be neutralized while ensuring the ranking performance on the real image pairs. (2) The ranker avoids providing biased ranking prediction for unrealistic image pairs. As we constrain the generated pairs at the decision boundary, i.e, $R(\mathbf{x}, \hat{\mathbf{y}}) = 0$, the ranker is invariant against variances of generated pairs [41]. Especially, the influence of unrealistic features contained in the generated pairs on the ranking decision is suppressed. (3) The ranker can capture the exclusive difference over the specified attribute through the adversarial process. Since the ranker rejects to give effective feedback for unrealistic image pairs, only the realistic image pairs can attract the attention of the ranker. Therefore, the ranker only passes the effective information related to the target attribute to the generator.

We broaden the principle of adversarial training by extending the adversarial game on classification to ranking. The novel adversarial ranking between the ranker and the generator is

defined on comparisons between pairs of samples, which can trigger a high-quality generation conditioned on a continuous variable. By contrast, the adversarial classification between the discriminator and the generator is defined on single samples, which can only trigger a high-quality unconditional generation.

Remark 1. *We follow [16] to assign label 0 to pairs* $\{(\mathbf{x},\mathbf{y})|\mathbf{y}=\mathbf{x}\}$*, called "ties" in learning to rank [42]. Label 0 denotes that there is no difference between* $\mathbf{x}$ *and* $\mathbf{y}$ *w.r.t. the specified attribute. The learning of these pairs can further improve the ranking prediction [42]. Label 0 for generated pairs* $(\mathbf{x},\hat{\mathbf{y}})$ *in Eq. (2) is to make the ranker think no difference on the generated pairs w.r.t. the specified attribute. Thus, assigning label 0 to similar real pairs and generated pairs have a same physical meaning.*

C. Linearizing Ranking Output for Smooth Translation

Eq. (3) models the relative attributes of the generated pairs $(\mathbf{x},\hat{\mathbf{y}})$ as a binary classification, which would fail to enable a continuous translation since the nuanced changes implied by the latent variable are not distinguished by the ranker. For example, given $v_1 > v_2 > 0$, the ranker gives same feedbacks, i.e., $\text{sgn}(v_1) = \text{sgn}(v_2) = +1$, for $(\mathbf{x},\hat{\mathbf{y}}_1)$ and $(\mathbf{x},\hat{\mathbf{y}}_2)$, where $\hat{\mathbf{y}}_* = G(\mathbf{x},v_*)$. This loses the discrimination between the two pairs. To achieve continuous translation, we linearize the ranking outputs for the generated pairs so as to align the ranking predictions with the latent variables. We thus reformulate the binary classification as the regression:

$$R(\mathbf{x},\hat{\mathbf{y}}) = v. \tag{5}$$

Given two latent variables $1 > v_2 > v_1 > 0$, the ranking prediction for the pair generated from v_2 should be larger than that from v_1, i.e., $1 > R(\mathbf{x},\hat{\mathbf{y}}_2) > R(\mathbf{x},\hat{\mathbf{y}}_1) > 0$. The ranking outputs for the generated pairs would be linearly correlated to the corresponding latent variables. Since the ranking output for a generated pair, i.e., $R(\mathbf{x},\hat{\mathbf{y}})$, reflects the difference between the generated image and the input image, the generated image $\hat{\mathbf{y}} = G(\mathbf{x},v)$ can change smoothly over the input image $\mathbf{x}$ according to the latent variable v.

As explained in Section III-B, the ranker's generalization to the generated pairs is enhanced by adversarial preferences (Eq. (4)). To further generalize the ranker to have linearized ranking outputs for the generated pairs (Eq. (5)), we tailor-design the structure for the ranker and adopt the least square loss for the ranking predictions shown in Eq. (6). Specifically, (1) the ranker is designed to output continuous values for real image pairs $(\mathbf{x},\mathbf{y})$, i.e., the last layer being the linear dense layer (Table 1 in Supplementary). Denoting the mapping before the last layer as f and the parameter of the last layer as w, the term $(R(\mathbf{x},\mathbf{y})-r)^2$ in Eq. (6a) can be reformulated as $(w^T f(\mathbf{x},\mathbf{y})-r)^2$, where $r = \begin{cases} 1 & \mathbf{y} \succ \mathbf{x} \\ 0 & \mathbf{y} = \mathbf{x} \\ -1 & \mathbf{y} \prec \mathbf{x} \end{cases}$ denotes the relative attribute. This actually learns the prediction of discrete RAs as a regression problem, which makes it easier to generalize the ranker to have a continuous ranking predictions. (2) The least square loss allows our ranker to produce soft predictions for real discrete RAs r since the loss has a small penalty when the prediction is close to the ground truth (Fig. 2[2] in [43]). Such flexibility enables our ranker to capture the discrepancy of different real image pairs $(\mathbf{x},\mathbf{y})$ w.r.t. the specified attribute. As a result, our ranker is capable to provide a continuous ranking predictions $R(\mathbf{x},\mathbf{y}) \in \mathbb{R}$, instead of discrete values in $\{-1,0,1\}$[3].

TABLE I
THE FUNCTION OF DIFFERENT LOSSES IN OUR CTAP.

Loss	Smooth translation	High-quality generation	
		Realistic	Attribute-specified
Rank head (Eq. (6))	✓	✓	✓
GAN head (Eq. (7))	✗	✓	✗

Remark 2. *Benefiting from the tailor-designed ranker, the ranker's outputs for the generated pairs is designed to be linearly correlate to the corresponding latent variable* v*. This novel loss guides the generator of our CTAP for superior smooth translation. See Section IV-C for more details.*

D. Continuous Translation via Adversarial Preferences (CTAP)

In the following, we introduce the loss functions for the ranker. The overall network structure can be seen in Fig. 2.

Loss of rank head R: the loss function for the rank head and the generator is defined as:

$$L^R_{rank} = \mathbb{E}_{p(\mathbf{x},\mathbf{y},r)}\left[\left(R\left(\mathbf{x},\mathbf{y}\right)-r\right)^2\right] + \lambda\mathbb{E}_{p(\mathbf{x})p(v)}\left[\left(R\left(\mathbf{x},G(\mathbf{x},v)\right)-0\right)^2\right]; \tag{6a}$$

$$L^G_{rank} = \mathbb{E}_{p(\mathbf{x})p(v)}\left[\left(R\left(\mathbf{x},G(\mathbf{x},v)\right)-v\right)^2\right], \tag{6b}$$

where $p(\mathbf{x},\mathbf{y},r)$ are the joint distribution of real image pairs $(\mathbf{x},\mathbf{y})$ and the relative attributes r. $p(\mathbf{x})$ is the distribution of the training images. $p(v)$ is a uniform distribution on $[-1,1]$. λ determines the strength of adversarial training between the ranker and the generator. Note we simplify $R\left(F(*_1),F(*_2)\right)$ as $R(*_1,*_2)$, where F is the feature layer in the ranker.

The rank head loss (Eq. (6)) is effective for smooth translation, which will be empirically validated in Section IV-D. Nevertheless, the ranking loss focuses on the difference of a pair of samples, so it is somehow inferior for the relativistic quality of single samples. We empirically find that our adversarial ranking loss can guarantee a superior attribute-specified quality but only boost a relatively good realistic quality (Section IV-D). To further improve the realistic quality, we introduce a parallel GAN head following the feature layer to ensure the image quality together with the rank head. The GAN head loss focuses on the quality of single samples and can further improve the ranker's ability to distill realistic features.

Loss of GAN head D: to be consistent with the above rank head and also ensure a stable training[4], a regular least square

[2]Comparison among commonly used losses (including the least square loss and the cross-entropy loss) for binary classification.

[3]In Fig. 12, the ranking outputs for the real image pairs, i.e., $R(\mathbf{x},\mathbf{y})$, locate at a neighborhood area of -1, 0, and 1 when convergence (iteration 99K), which almost covers the range from -1 to 1.

[4]The least square GAN loss is demonstrated to possess superior training stability compared to the original GAN loss in [44].

Algorithm 1 Continuous Image-to-Image Translation with Adversarial Preferences

Input: Training data $\mathcal{X} = \{\mathbf{x}_n\}_{n=1}^{N}$, triplet set $\mathcal{S} = \{(\mathbf{x}_m, \mathbf{y}_m, r_m)|\mathbf{x}_m, \mathbf{y}_m \in \mathcal{X}, r_m \in \{-1, 0, 1\}\}_{m=1}^{M}$.
Hyperparameter: batch size B, training iterations T, parameters λ, λ_g and λ_{gp}.
Initialize: the generator and the ranker (the shared feature layer, the GAN head and the rank head).
for $t = 1, 2, \cdots, T$ **do**
  Sample a mini-batch of the triplet set $\mathbf{S} = \{(\mathbf{x}_i, \mathbf{y}_i, r_i)\}_{i=1}^{B}$ from $\mathcal{S}$.
  Sample a mini-batch of samples $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{B}$ from $\mathcal{X}$.
  Get the fake samples $\hat{\mathbf{Y}} = \{G(\mathbf{x}_i, v_i)|v_i \sim U[-1, 1]\}_{i=1}^{B}$ from the generator G.
  Get the cycle reconstructed samples $\hat{\mathbf{X}} = \{G\left(G(\mathbf{x}_i, v_i), -v_i\right)|v_i \sim U[-1, 1]\}_{i=1}^{B}$.
  Construct the fake sample pair $\hat{\mathbf{S}} = \{(\mathbf{x}_i, G(\mathbf{x}_i, v_i))\}_{i=1}^{B}$.
  Train the ranker using the loss $L_{\text{rank}}^{R}\left[\mathbf{S}, \hat{\mathbf{S}}\right] + \lambda_g L_{\text{gan}}^{R}\left[\mathbf{X}, \hat{\mathbf{Y}}\right] + \lambda_{gp} L_{gp}\left[\mathbf{X}, \hat{\mathbf{Y}}\right]$.
  Train the Generator using the loss $L_{\text{rank}}^{G}\left[\hat{\mathbf{S}}\right] + \lambda_g L_{gan}^{G}\left[\hat{\mathbf{Y}}\right] + \lambda_c L_{\text{cycle}}\left[\mathbf{X}, \hat{\mathbf{X}}\right]$.
end for

GAN's loss is adopted:

$$L_{gan}^{D} = \mathbb{E}_{p(\mathbf{x})}\left[\left(D\left(\mathbf{x}\right) - 1\right)^2\right] + \mathbb{E}_{p(\mathbf{x})p(v)}\left[\left(D\left(G(\mathbf{x}, v)\right) - 0\right)^2\right]; \tag{7a}$$

$$L_{gan}^{G} = \mathbb{E}_{p(\mathbf{x})p(v)}\left[\left(D\left(G(\mathbf{x}, v)\right) - 1\right)^2\right], \tag{7b}$$

where 1 denotes the real image label and 0 denotes the fake image label. Note that we simplify $D\left(F(*)\right)$ as $D(*)$.

Jointly training the rank head and the GAN head, the gradients backpropagate through the shared feature layer to the generator. Our CTAP can effectively take into account both goals of smooth translation and high-quality generation (Table I).

As the translation is conducted on the unpaired setting, the cycle consistency loss L_{cycle} is usually introduced to keep the identity of faces [4], [14], [17]. The gradient penalty loss L_{gp} and the orthogonal regularization are added to stabilize the adversarial training between the generator and the ranker [17]. The orthogonal loss can prevent the network from converging too quickly to a subset of modes by promoting orthogonality among the weights [45]. The gradient penalty loss can prevent the ranker from becoming too sensitive to small changes in the generator's output by penalizing the gradient norm of the ranker, thus stabilizing the training [46]. See the supplementary for the detailed equations and implementation. The training algorithm is summarized in Algorithm 1.

E. Extended to the Multiple Attributes

To generalize CTAP to multiple (K) attributes, we use vectors $\mathbf{v}$ and $\mathbf{r}$ with K dimension to denote the latent variable and the preference label, respectively. Each dimension controls the change of one interested attribute. In particular, the ranker consists of one GAN head and K parallel rank heads. The overall loss function is summarized as follows:

$$L_{rank}^{R} = \mathbb{E}_{p(\mathbf{x},\mathbf{y},\mathbf{r})}\sum_{k}\left[\left(R_k\left(\mathbf{x}, \mathbf{y}\right) - \mathbf{r}_k\right)^2\right]] + \lambda\mathbb{E}_{p(\mathbf{x})p(\mathbf{v})}\sum_{k}\left[\left(R_k\left(\mathbf{x}, G(\mathbf{x}, \mathbf{v})\right) - 0\right)^2\right]; \tag{8a}$$

$$L_{rank}^{G} = \mathbb{E}_{p(\mathbf{x})p(\mathbf{v})}\sum_{k}\left[\left(R_k\left(\mathbf{x}, G(\mathbf{x}, \mathbf{v})\right) - \mathbf{v}_k\right)^2\right], \tag{8b}$$

where R_k is the output of the k-th rank head. $\mathbf{v}_k$ and $\mathbf{r}_k$ are the k-th dimension of $\mathbf{v}$ and $\mathbf{r}$, respectively.

IV. Experiments

In this section, we compare our CTAP with various baselines on the task of continuous I2I translation. We then verify that our ranker can distinguish the nuanced difference in a pair of images. Thus, we propose to apply our ranker for evaluating the fine-graininess of image pairs generated by various methods. We finally extend CTAP to the translation of multiple attributes.

Datasets. All experiments are conducted in the real-world datasets with discrete RAs $r \in \{+1, 0, -1\}$. In particular, we use two face image datasets, i.e., the high quality subset of Celeb Faces Attributes Dataset (CelebA-HQ) [47] and Labeled Faces in the Wild with attributes (LFWA) [48], and one shoe image dataset, i.e., UT-Zappos50K dataset (UT-Zap50K) [49]. CelebA-HQ consists of 30K face images of celebrities with 40 binary attributes. LFWA has $13,143$ images with 73 binary attributes. UT-Zap50K contains $50,025$ catalog shoe images together with corresponding edge images [3], where the binary label is 1 when an image is a shoe image and is 0 when an image is an edge image. We resize the images to 256×256 for three datasets. The relative attributes are obtained for any two images $\mathbf{x}$ and $\mathbf{y}$ based on the binary labels, following the former work [18], [17]. Thus we can make a fair comparison with other baselines in terms of same supervision information. For instance, for "smile" attribute, we construct the comparison $\mathbf{x} > \mathbf{y}$ when the "smile" label of $\mathbf{x}$ is 1 while the "smile" label of $\mathbf{y}$ is 0, and vice versa. We find that all methods fail to achieve satisfactory results on the shoe dataset using the discrete relative attributes. To enhance the continuous translation on UT-Zap50K, we augment the training images by mixing up shoe images with the corresponding edge images [50]. Specifically, a mix-up image is obtained by $(1-\gamma)\mathbf{x} + \gamma\mathbf{y}$, where $\mathbf{y}$ is a shoe image and $\mathbf{x}$ is $\mathbf{y}$'s corresponding edge image. The corresponding mix-up label is γ, where $\gamma \in (0, 1)$.

Implementation Details. The weighting factors for L_{gan}, L_{cycle} and L_{gp} are λ_g, λ_c and λ_{gp}, respectively. Except $\lambda_g = 0.5$ for CelebA-HQ, $\lambda_g = 5$ for LFWA and $\lambda_g = 2.5$ for UT-Zap50K, we set the same parameter for all datasets. Specifically, we set $\lambda = 0.5$, $\lambda_c = 2.5, \lambda_{gp} = 150, \lambda_o = 10^{-6}$. We use the

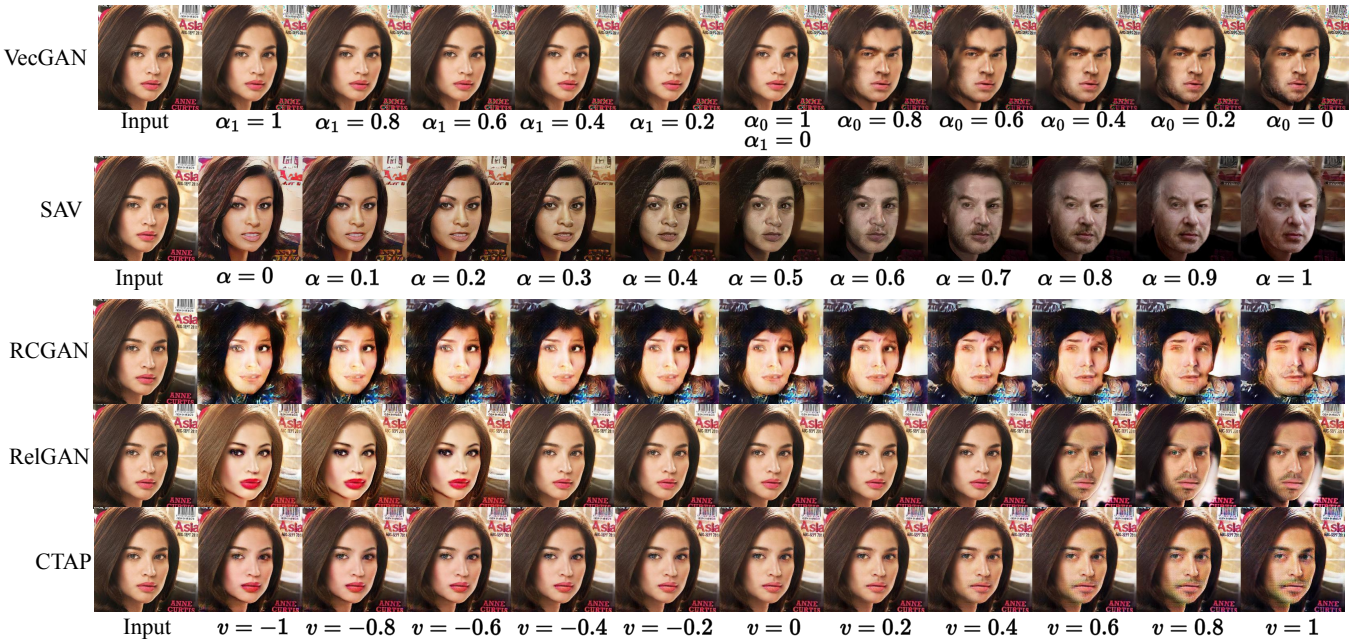


Fig. 5. Comparison of continuous facial attribute ("gender") translation on CelebA-HQ. α_1 and α_0 in VecGAN are interpolation coefficients for binary attribute 1 ("female") and 0 ("male"), respectively. Note $\alpha_1 = 0$ and $\alpha_0 = 1$ correspond to images with the same attribute strength. α in SAV is the interpolation coefficient.

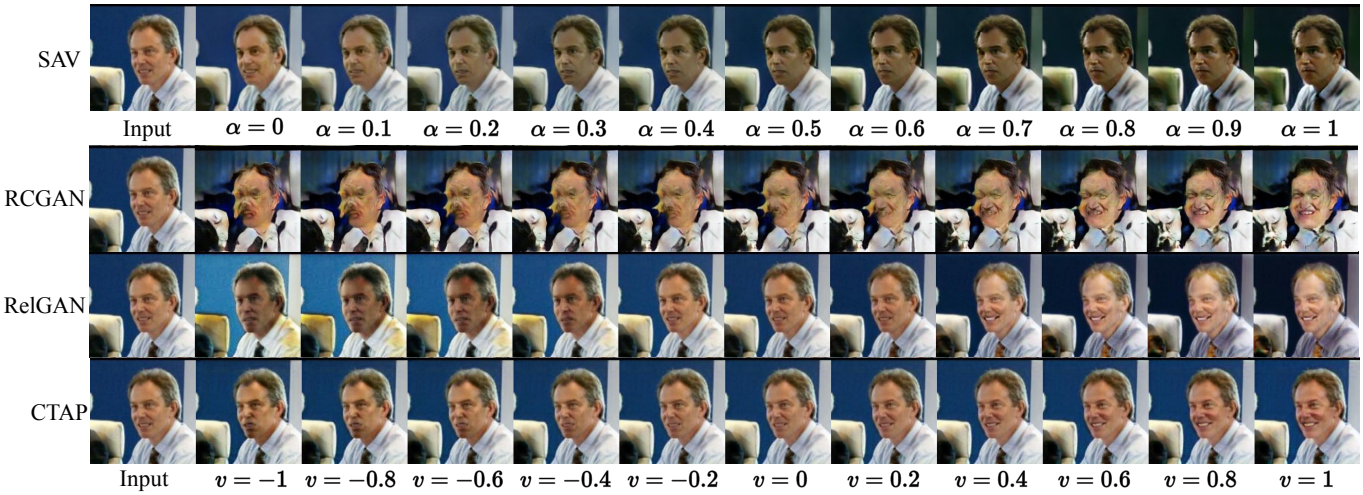


Fig. 6. Comparison of continuous facial attribute ("smile") translation on LFWA.

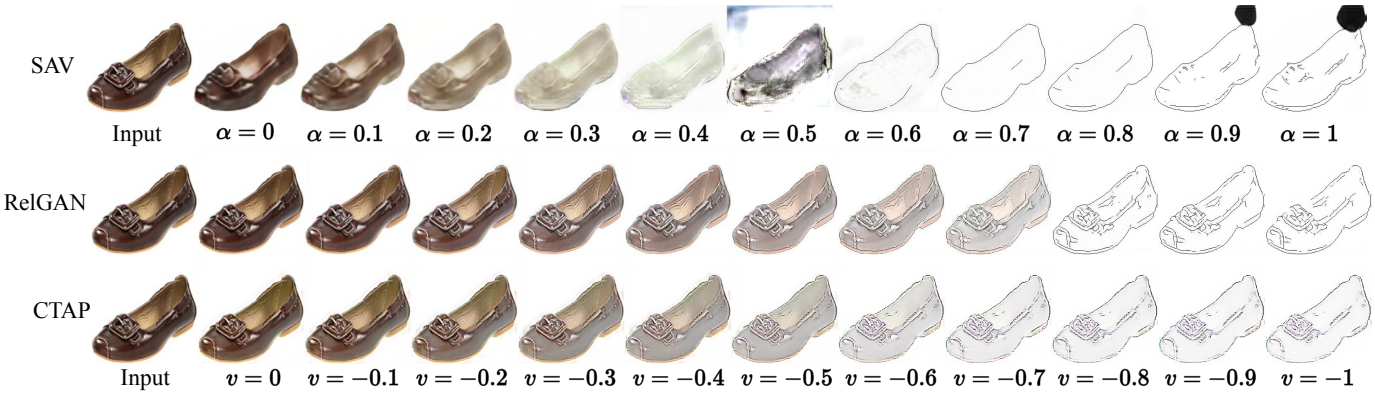


Fig. 7. Comparison of continuous style translation ("shoe→edge") on UT-Zap50K.

Adam optimizer with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The learning rate is set to 10^{-5} for the ranker and 5×10^{-5} for the generator.

TABLE II
PERFORMANCE OF CONTINUOUS TRANSLATION (DSSIM) AND IMAGE QUALITY (FID/MSE) ON CELEBA-HQ, LFWA AND UT-ZAP50K. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD. THE MISSING VALUES ARE DUE TO EITHER THE LACK OF THE TRAINING CODE (VECGAN) OR FAILURE OF THE TRANSLATION (RCGAN).

Model	Continuous Translation (DSSIM)							Image quality (FID/MSE)						
	CelebA-HQ				LFWA		UT-Zap50K	CelebA-HQ (FID)				LFWA (FID)		UT-Zap50K (MSE)
	Smile	Gender	Mouth	Cheekbones	Smile	Frown	shoe→edge	Smile	Gender	Mouth	Cheekbones	Smile	Frown	shoe→edge
VecGAN	0.0135	0.0660	-	-	-	-	-	26.34	45.37	-	-	-	-	-
SAV	0.0279	0.0452	0.0278	0.0331	0.0169	0.0173	0.0809	48.03	59.85	48.38	56.57	77.07	78.61	19053.84
RCGAN	0.0084	0.0138	-	0.0099	0.0079	0.0106	-	398.05	418.06	-	385.27	437.93	425.59	-
RelGAN	0.0679	0.0687	0.0383	0.0450	0.0172	0.0179	0.0258	**12.01**	**27.12**	10.51	**10.27**	**16.40**	**18.43**	6973.81
CTAP	**0.0021**	**0.0044**	**0.0021**	**0.0026**	**0.0005**	**0.0006**	**0.0155**	**11.07**	28.61	**8.62**	**10.52**	19.23	**18.85**	**6142.14**

The batch size is set to 4. We split the dataset into training/test with a ratio $90/10$ for face datasets. We use 40K shoe-edge image pairs (80K images in total) and the corresponding mix-up images (40K images in total) as training data and $10,025$ shoe-edge image pairs ($20,050$ images in total) as test data for UT-ZAP50K dataset. We ease the training by sequencing the learning of our CTAP. Specifically, we first pretrain our CTAP by deactivating L_{rank} to enable good generation performance from the generator. When L_{rank} is introduced to train CTAP, our ranker can predominantly concentrate on the relationship between the generated pairs and the corresponding attribute differences v, rather than simultaneously managing translation and generation quality. The experiments are run with the keras framework on 24 GB NVIDIA Quadro RTX 6000. All the results are obtained by a single run. Each run contains 100K iterations. See supplementary for details about the network architecture and more experiment setting.

Baselines. We compare CTAP with the most related methods, i.e., RA based methods RelGAN [17] and RCGAN [18]. We also include two recent binary attribute based methods VecGAN [8] and SAV [9] for comparison. We use the released codes of these methods.

Evaluation Metrics. Following [17], we use four metrics to measure the performance of continuous translation.

- *Standard Deviation of Structural SIMilarity (DSSIM) evaluates the performance of smooth translation.* We first apply the generator to produce a set of fine-grained output images $\{\mathbf{x}_1, \ldots, \mathbf{x}_l\}$ by conditioning on an input image $\mathbf{x}$ and a set of latent variable $v = -1 : 0.2 : 1$[5]. We compute the standard deviation of Structural SIMilarity (SSIM) between x_{i-1} and x_i as follows:

$$\text{DSSIM} = \sigma\left(\{\text{SSIM}\left(\mathbf{x}_{i-1}, \mathbf{x}_i\right) \mid i = 1, \cdots, l\}\right). \quad (9)$$

  We calculate DSSIM for each image in the test dataset and average them to get the final score. Smaller DSSIM denotes better smooth translation.
- *Mean Square Error (MSE) evaluates generation quality for the shoe dataset.* For shoe→edge in UT-ZAP50K, the input-output examples are paired. Namely, the input image (shoe) and the output image (edge) only differ in their style, i.e., photo style and edge style. Thus, the MSE between the translated images from generative models and the ground-truth output images can be calculated to measure the visual quality of translation. We conduct translation for all images in UT-ZAP50K dataset, obtaining $50,025$ generated pairs, and calculate the MSE. Smaller MSE denotes generation with better quality.
- *Frechet Inception Distance (FID) evaluates generation quality for the face dataset.* As the input-ouptut examples in face datasets are unpaired, we evaluate the statistical difference between the translated images and the training dataset to measure the visual quality. It is calculated with $30K$ translated images on CelebA-HQ dataset and $13,143$ translated images on LFWA dataset. Smaller FID denotes generation with better quality.
- *Accuracy of Attribute Swapping (AAS) evaluates the accuracy of binary translation.* The attribute swapping is to translate an image in a binary manner, e.g., from "smile" to "no smile". The accuracy is evaluated by a facial attribute classifier that uses the Resnet-18 architecture [51]. To obtain AAS, we first translate the test images with the trained GANs and then apply the classifier to evaluate the classification accuracy of the translated images coupled with its swapping attribute. Higher accuracy means that more images are translated as desired.

[5]The number of image sequences $l = \frac{1-(-1)+0.2}{0.2} = 11$.

A. Continuous Image-to-Image Translation

We conduct continuous I2I translation on a single attribute. On CelebA-HQ dataset, we translate images in terms of "smile", "gender", "mouth open" and "high cheekbones" attributes, respectively. On LFWA dataset, we translate images in terms of "smile" and "Frown" attributes, respectively. On UT-ZAP50K dataset, we translate the shoe images to edge images, i.e., "shoe→edge". We show that our CTAP achieves the state-of-art performance on the continuous I2I translation task comparing with various baselines, especially in terms of the continuous translation score DSSIM.

Visual results. Fig. 5, Fig. 6 and Fig. 7 show that: (1) Benefiting from the adversarial ranking process and the tailor-design of the ranker, our CTAP achieves the best visual quality, generating realistic output images that are different from the input images only in the specified attribute. (2) Regarding the translation of the "gender" attribute in Fig. 5, it would require substantial modifications to the faces. All baselines almost lose

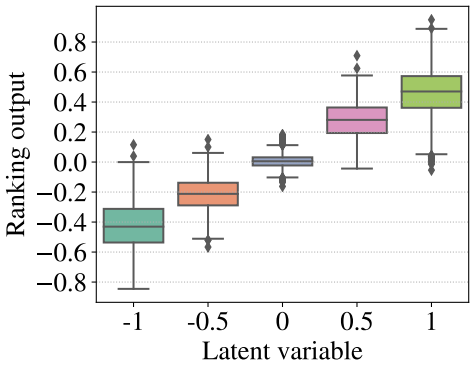


Fig. 8. The box plot of the ranking output for generated pairs $R(\mathbf{x}, \hat{\mathbf{y}})$ with different values of the latent variable v.

the facial identity during translation, with SAV and RCGAN even losing the background. In contrast, our CTAP achieves maximum modification while preserving the facial identity. (3) RelGAN's generation not only changes the specified attribute, but is also influenced by other irrelevant attributes, e.g., "hair color". This is because its matching-aware discriminator is not effective to learn attribute information. In addition, due to its quality constraint for interpolated images (see Supplementary for more details), the translation by RelGAN is not smooth, with a relatively large DSSIM. (4) RCGAN exhibits extremely poor generation results. Because its ranker that is only trained with real images generalizes poorly to the generated images with interpolated RAs. (5) The translation of binary attribute based methods SAV and VecGAN is also not smooth, with a large DSSIM since the binary attribute is not sufficient to model nuanced changes of the attribute. (6) The translation of SAV also involves the change of irrelevant attributes because its latent space has not been well disentangled, indicated by its original paper [9].

Continuous translation score. We present the quantitative evaluation of the continuous translation in Table II. Our CTAP achieves the lowest DSSIM scores for three datasets, consistent with the visual results. Note that a trivial case to obtain a low DSSIM is when the translation is failed. Namely, the generator would output the same image no matter what the latent variable is. Therefore, we further apply AAS to evaluate the I2I translation in a binary manner. All methods mostly achieve over 95% accuracy (see Supplementary). Under this condition, it guarantees that a low DSSIM indeed indicates the output images change smoothly with the latent variable.

Image quality score. Table II presents the quantitative evaluation of the image quality. Our CTAP achieves the best image quality with the lowest FID scores. RCGAN has the worst FID scores, consistent with the visual results in Fig. 5 and Fig. 6. SAV obtains a bad MSE score on "shoe→edge", due to its flaw on the translated results shown in Fig. 7.

B. Physical Meaning of Ranking Output

Fig. 8 shows that (1) for a large v, the rank head of our ranker would output a large prediction. It demonstrates that the ranker indeed generalizes to synthetic image pairs and can discriminate the nuanced change among each image pair. (2) The ranker can capture the whole ordering instead of the exact value w.r.t. the latent variable. Because the ranker assigning 0 to the generated pairs inhibits the generator's loss optimizing to zero, although our generator's objective is to ensure the ranking output (namely, the output of rank head in Fig. 2) are consistent with the latent variable v. The ranker achieves an equilibrium with the generator when converging.

From Fig. 5 and Table II, it shows that when conditioning on different latent variables v, our CTAP can translate an input image $\mathbf{x}$ into a series of output images $\hat{\mathbf{y}} = G(\mathbf{x}, v)$ that exhibit the corresponding changes over the specified attribute. We then evaluate the function of our ranker using these generated pairs $(\mathbf{x}, \hat{\mathbf{y}})$. It verifies that our ranking output $R(\mathbf{x}, \hat{\mathbf{y}})$ well-aligns to the relative change v in the pair of images.

We further evaluate continuous I2I translations w.r.t. the "smile" attribute on the test dataset of CelebA-HQ (Fig. 8). The trained generator is applied to generate a set of $G(\mathbf{x}, v)$ by taking as inputs an image $\mathbf{x}$ and $v = -1.0, -0.5, 0.0, 0.5, 1.0$, respectively. We collect the ranking output for each generated pair, i.e., $R(\mathbf{x}, \hat{\mathbf{y}})$, and plot the density in terms of different v.

C. Linear Tendency on the Latent Variable

As verified above, our ranker can reveal the changes between image pairs. We use it to evaluate the nuanced differences between the synthetic image pairs generated by RA based methods to further compare our CTAP with RCGAN and RelGAN.

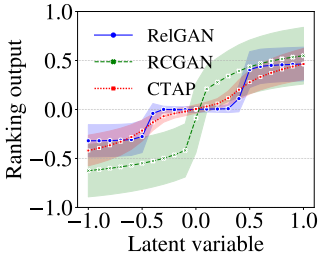
We generate the pairs with continuous translation on the test dataset of CelebA-HQ w.r.t. the "smile" attribute. Each trained model produces a series of synthetic images $\hat{\mathbf{y}}$ by taking as input a real image $\mathbf{x}$ and different latent variables v. The range of v is from -1 to 1 with step 0.1. Then the ranker, pre-trained by our CTAP, is applied to evaluate the generated pairs $(\mathbf{x}, \hat{\mathbf{y}})$ and group them in terms of different conditioned latent variables v for different models, respectively. In terms of each group, we calculate the mean and the standard deviation (std) for the ranking outputs (Fig. 9).

Fig. 9 shows that (1) the ranking output $R(\mathbf{x}, \hat{\mathbf{y}})$ of CTAP exhibits a linear trend with the lowest variance w.r.t. the latent variable v. This demonstrates that CTAP can smoothly translate the input image $\mathbf{x}$ to the desired image $\hat{\mathbf{y}} = G(\mathbf{x}, v)$ over the specified attribute along v. (2) The ranking output of RCGAN behaves like a $\tanh$ curve with a sudden change when the latent variable is around zero. It means that RCGAN cannot smoothly control the attribute strength for the input image. In addition, RCGAN has the largest variance on the ranking output due to the low quality of the generated images, which introduces noises to the ranking prediction on the generated pairs. (3) RelGAN manifests a three-step like curve, which indicates a failure of continuous generation. This is mainly because of its specific design of the interpolation loss.

D. Ablation Study

Fig. 10 and Table III show an ablation study of our model.

a) Function of GAN head (Table I): Solely training with the adversarial ranking loss, i.e., without (w/o) L_{gan}, yields relatively realistic images. The generated images change exclusively w.r.t. the "smile" attribute along with the controlled variable. However, this model obtains higher FID score and poorer



v	RelGAN	RCGAN	CTAP
$-1.$	0.165	0.268	**0.157**
-0.5	0.164	0.265	**0.107**
$0.$	**0.030**	0.354	0.042
0.5	0.183	0.273	**0.121**
$1.$	0.172	0.290	**0.159**

Fig. 9. **Left panel:** plots the ranking output for generated pairs $R(\mathbf{x}, \hat{\mathbf{y}})$ in terms of different latent variables v. The curve shows the mean of the output, while the shaded region depicts the standard deviation of the output. **Right panel:** summarizes the standard deviation of the ranking output.

images than the model with L_{gan} (i.e., CTAP), which demonstrates the effect of GAN head loss on the realistic quality.

b) Function of rank head (Table I): (1) Without L_{rank}, the generated images exhibit no change over the input image. The generator fails to learn the translation function, with an extremely low translation accuracy (AAS). This demonstrates that the function of rank head loss L_{rank} to the performance of smooth translation. (2) Adversarial ranking loss vs. vanilla ranking loss (i.e., w/o L_{gan} vs. w/o L_{gan} & $\lambda = 0$). The later model performs worse in terms of both the realistic quality and the attribute-specified quality. It produces unrealistic images, with a very high FID. Meanwhile, the generated images change in terms of another aspect (the background color of images) rather than the "smile" attribute along with the controlled variable, with a low AAS. This demonstrate the effect of rank head loss on the realistic quality and attribute-specific quality.

c) Effect of using generated pairs to train the ranker: Setting $\lambda = 0$, which means not considering the generated pairs into the training of the ranker, the performance of facial image manipulation deteriorates, obtaining a low translation accuracy (AAS).

d) Eq. (3) vs. Eq. (5): When optimizing with Eq. (3), i.e., not linearing the ranking output for the generated pairs, the fine-grained control over the attributes fails, getting a high DSSIM score.

TABLE III
QUANTITATIVE EVALUATION OF ABLATION STUDY ON CELEBA-HQ ("SMILE" ATTRIBUTE). "BIN" MEANS REPLACING EQ. (5) WITH EQ. (3), NAMELY, NOT LINEARING THE RANKING OUTPUT FOR THE GENERATED PAIRS. WITHOUT (W/O); WITH (W).

Model	w/o L_{rank}	w/o L_{gan}&$\lambda = 0$	w/o L_{gan}	$\lambda = 0$	bin	CTAP
AAS (↑)	9.73	46.2	98.47	55.7	92.37	97.69
DSSIM (↓)	1.51E-05	0.0016	0.0058	0.0011	0.0163	0.0030
FID (↓)	29.00	428.35	78.08	8.55	11.33	10.19

E. Extension to Multiple Attributes

We conduct continuous I2I translation with both "smile" and "gender" on CelebA-HQ to show that CTAP can generalize well to multiple attributes. A two-dimension variable is used to control the change of both attributes, simultaneously.

The generated images conditioning on different $\mathbf{v}$ are shown in Fig. 11. (1) CTAP can disentangle multiple attributes. When $\mathbf{v} = [-1, 0]/[1, 0]$, the generated images $\hat{\mathbf{y}}_{\text{-1,0}}/\hat{\mathbf{y}}_{1,0}$ appear "less smiling"/"more smiling" with no change on the "gender" attribute. When $\mathbf{v} = [0, -1]/[0, 1]$, the generated images $\hat{\mathbf{y}}_{0,\text{ -1}}/\hat{\mathbf{y}}_{0,1}$ appear "less masculine"/"more masculine" with no change in the "smile" attribute. In addition, a fine-grained control over the strength of a single attribute is still practical. (2) CTAP can manipulate the nuanced changes of multiple attributes simultaneously. For example, when conditioning $\mathbf{v} = [1, -1]$, the generated image $\hat{\mathbf{y}}_{1,\text{ -1}}$ appear "less smiling" and "more masculine".

F. Training Convergence of CTAP

CTAP converges to an equilibrium when the generator produces the realistic image with desired changes over the input image regarding the target attribute, and the ranker makes reliable prediction for the difference between the translated images and the input image w.r.t. the target attribute.

To justify our claim, we plot the distribution of the ranking prediction for real image pairs and generated image pairs with different RAs $-1/0/1$ using the ranker in Fig. 12. (1) At the beginning of the training, the ranker gives similar predictions for real image pairs with different RAs. The same observations can also be found on the generated image pairs. (2) After 100 iterations, the ranker learns to give the desired prediction for different kinds of pair, i.e., > 0 (averaged) for pairs with RAs $(+1)$, 0 (averaged) for pairs with RAs (0) and < 0 (averaged) for pairs with RAs (-1). (3) After $9,900$ iterations, CTAP converges. In terms of the real image pairs, the rank head outputs $+1$ for the pairs with RAs $(+1)$, 0 for the pairs with RAs (0) and -1 for the pairs with RAs (-1) in the sense of average. This verifies that our ranker can give precise ranking predictions for real image pairs. In terms of the generated pairs, the rank head outputs $+0.5$ for the pairs with RAs $(+1)$, 0 for the pairs with RAs (0) and -0.5 for the pairs with RAs (-1) in the sense of average. This is a convergence state due to adversarial preferences. We take pairs with RAs $(+1)$ as an example. The generated pairs with RAs $(+1)$ are expected to be assigned 0 when optimizing the ranker, and to be assigned $+1$ when optimizing the generator. Therefore, the convergence state should be around 0.5.

V. CONCLUSION

This paper proposes CTAP for high-quality continuous I2I translation. CTAP elegantly reconciles the goal for continuous translation and the goal for high-quality generation through adversarial ranking. It broadens the principle of adversarial training by extending the adversarial game on classification to ranking and enables generation with nuanced changes.

Although introducing the orthogonal regularization and the gradient penalty loss can help to stabilize the adversarial training following [17], we have observed that the translation performance is not always stable. Thus, we applied the Accuracy of Attribute Swapping (AAS) to choose the model checkpoint with a good translation performance as clarified in Section II-C of the supplementary. On the other hand, it would be interesting to extend our idea to diffusion models, which has a more stable training.

Fig. 10. Ablation study on CelebA-HQ ("smile"). L_{rank}, L_{gan}, "bin" and "con" refer to Eq. (6), Eq. (7), Eq. (3), and Eq. (5), respectively. The models in rows 1 to 6 correspond to the models in columns 2 to 7 of Table III, respectively.

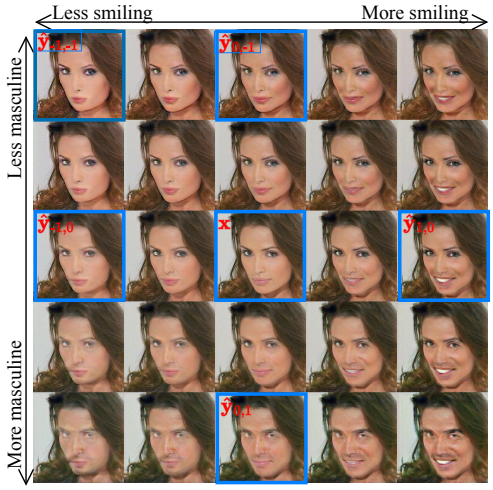


Fig. 11. Continuous I2I translation with "smile" and "gender" attributes. The middle is the input image $\mathbf{x}$ and others are the generated images $\hat{\mathbf{y}} = G(\mathbf{x}, \mathbf{v})$.

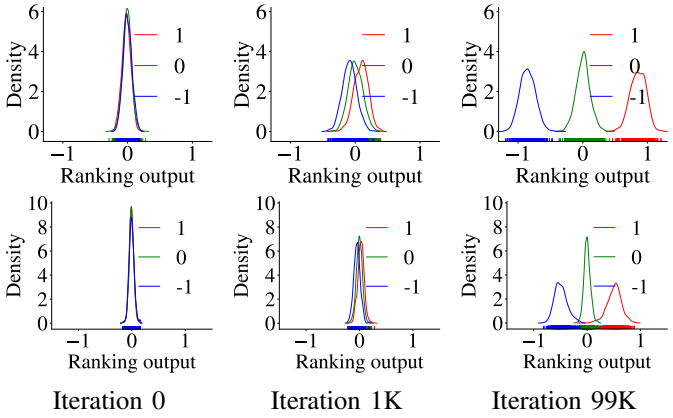


Fig. 12. The density plot of the ranking output for real image pairs $R(\mathbf{x}, \mathbf{y})$ (first row) and generated pairs $R(\mathbf{x}, \hat{\mathbf{y}})$ (second row) in terms of different relative attributes $-1/0/1$, respectively, during the training process.

REFERENCES

[1] C. Wang, C. Xu, and D. Tao, "Self-supervised pose adaptation for cross-domain image animation," *IEEE Transactions on Artificial Intelligence*, vol. 1, no. 1, pp. 34–46, 2020.

[2] L. Kong, C. Lian, D. Huang, Y. Hu, Q. Zhou *et al.*, "Breaking the dilemma of medical image-to-image translation," *Advances in Neural Information Processing Systems*, vol. 34, pp. 1964–1978, 2021.

[3] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1125–1134.

[4] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *IEEE International Conference on Computer Vision*, 2017, pp. 2223–2232.

[5] S. Huang, C. He, and R. Cheng, "Sologan: Multi-domain multimodal unpaired image-to-image translation via a single generative adversarial network," *IEEE Transactions on Artificial Intelligence*, vol. 3, no. 5, pp. 722–737, 2022.

[6] B. Li, K. Xue, B. Liu, and Y. Lai, "Bbdm: Image-to-image translation with brownian bridge diffusion models," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, 2023.

[7] G. Kwon and J. C. Ye, "Diffusion-based image translation using disentangled style and content representation," in *International conference on learning representations*, 2023.

[8] Y. Dalva, H. Pehlivan, O. I. Hatipoglu, C. Moran, and A. Dundar, "Image-to-image translation with disentangled latent vectors for face editing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[9] Q. Mao, H.-Y. Tseng, H.-Y. Lee, J.-B. Huang, S. Ma, and M.-H. Yang, "Continuous and diverse image-to-image translation via signed attribute vectors," *International Journal of Computer Vision*, vol. 130, no. 2, pp. 517–549, 2022.

[10] G. Lample, N. Zeghidour, N. Usunier, A. Bordes, L. Denoyer, and M. Ranzato, "Fader networks: Manipulating images by sliding attributes," in *Advances in Neural Information Processing Systems*, 2017, pp. 5967–5976.

[11] X. Li, C. Lin, C. Wang, and F. Guerin, "Latent space factorisation and manipulation via matrix subspace projection," *International Conference on Machine Learning*, 2020.

[12] Z. Ding, Y. Xu, W. Xu, G. Parmar, Y. Yang, M. Welling, and Z. Tu, "Guided variational autoencoder for disentanglement learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7920–7929.

[13] R. Liu, Y. Liu, X. Gong, X. Wang, and H. Li, "Conditional adversarial generative flow for controllable image synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7992–8001.

[14] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, "Attgan: Facial attribute editing by only changing what you want," *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5464–5478, 2019.

[15] J. Lin, Z. Chen, Y. Xia, S. Liu, T. Qin, and J. Luo, "Exploring explicit domain supervision for latent space disentanglement in unpaired image-to-image translation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 4, pp. 1254–1266, 2021.

[16] D. Parikh and K. Grauman, "Relative attributes," in *2011 International Conference on Computer Vision*. IEEE, 2011, pp. 503–510.

[17] P.-W. Wu, Y.-J. Lin, C.-H. Chang, E. Y. Chang, and S.-W. Liao, "Relgan: Multi-domain image-to-image translation via relative attributes," in *IEEE International Conference on Computer Vision*, 2019, pp. 5914–5922.

[18] Y. Saquil, K. I. Kim, and P. M. Hall, "Ranking cgans: Subjective control over semantic image attributes," in *British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, September 3-6, 2018*. BMVA Press, 2018, p. 131.

[19] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[20] A. H. Liu, Y.-C. Liu, Y.-Y. Yeh, and Y.-C. F. Wang, "A unified feature disentangler for multi-domain image translation and manipulation," in *Advances in neural information processing systems*, 2018, pp. 2590–2599.

[21] W. Huang, W. Luo, J. Huang, and X. Cao, "Sdgan: Disentangling semantic manipulation for facial attribute editing," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 3, 2024, pp. 2374–2381.

[22] K. Preechakul, N. Chatthee, S. Wizadwongsa, and S. Suwajanakorn, "Diffusion autoencoders: Toward a meaningful and decodable representation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 619–10 629.

[23] D. Berthelot, C. Raffel, A. Roy, and I. Goodfellow, "Understanding and improving interpolation in autoencoders via an adversarial regularizer," in *International Conference on Learning Representations*, 2018.

[24] Y.-C. Chen, X. Xu, Z. Tian, and J. Jia, "Homomorphic latent space interpolation for unpaired image-to-image translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2408–2416.

[25] Y. Shen, J. Gu, X. Tang, and B. Zhou, "Interpreting the latent space of gans for semantic face editing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9243–9252.

[26] T. Aoshima and T. Matsubara, "Deep curvilinear editing: Commutative and nonlinear image manipulation for pretrained deep generative model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 5957–5967.

[27] H. Pehlivan, Y. Dalva, and A. Dundar, "Styleres: Transforming the residuals for real image editing with stylegan," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 1828–1837.

[28] A. B. Yildirim, H. Pehlivan, B. B. Bilecen, and A. Dundar, "Diverse inpainting and editing with gan inversion," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 23 120–23 130.

[29] Y. Gao, F. Wei, J. Bao, S. Gu, D. Chen, F. Wen, and Z. Lian, "High-fidelity and arbitrary face editing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 115–16 124.

[30] O. Tov, Y. Alaluf, Y. Nitzan, O. Patashnik, and D. Cohen-Or, "Designing an encoder for stylegan image manipulation," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 4, pp. 1–14, 2021.

[31] Y. Deng, J. Yang, D. Chen, F. Wen, and X. Tong, "Disentangled and controllable face image generation via 3d imitative-contrastive learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5154–5163.

[32] H. Do, E. Yoo, T. Kim, C. Lee, and J. Y. Choi, "Quantitative manipulation of custom attributes on 3d-aware image synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 8529–8538.

[33] Y. Alharbi and P. Wonka, "Disentangled image generation through structured noise injection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5134–5142.

[34] A. Pumarola, A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer, "Ganimation: Anatomically-aware facial animation from a single image," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 818–833.

[35] J. Ling, H. Xue, L. Song, S. Yang, R. Xie, and X. Gu, "Toward fine-grained facial expression manipulation," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*. Springer, 2020, pp. 37–53.

[36] S. Jin, Z. Wang, L. Wang, P. Liu, N. Bi, and T. Nguyen, "Aueditnet: Dual-branch facial action unit intensity manipulation with implicit disentanglement," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 2104–2113.

[37] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8789–8797.

[38] Y. Cao, J. Xu, T.-Y. Liu, H. Li, Y. Huang, and H.-W. Hon, "Adapting ranking svm to document retrieval," in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 2006, pp. 186–193.

[39] Y. Yao, Y. Pan, J. Li, I. W. Tsang, and X. Yao, "Generative adversarial ranking nets," *Journal of Machine Learning Research*, vol. 25, no. 119, pp. 1–35, 2024.

[40] A. Odena, "Semi-supervised learning with generative adversarial networks," *Workshop on Data-Efficient Machine Learning (ICML)*, 2016.

[41] O. Chapelle, A. Agarwal, F. H. Sinz, and B. Schölkopf, "An analysis of inference with the universum," in *Advances in neural information processing systems*, 2008, pp. 1369–1376.

[42] K. Zhou, G.-R. Xue, H. Zha, and Y. Yu, "Learning to rank with ties," in *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2008, p. 275–282.

[43] A. Martins and R. Astudillo, "From softmax to sparsemax: A sparse model of attention and multi-label classification," in *International conference on machine learning*. PMLR, 2016, pp. 1614–1623.

[44] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, "Least squares generative adversarial networks," in *IEEE International Conference on Computer Vision*, 2017, pp. 2794–2802.

[45] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," in *International Conference on Learning Representations*, 2019.

[46] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," *Advances in neural information processing systems*, vol. 30, 2017.

[47] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," in *International Conference on Learning Representations*, 2018.

[48] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[49] A. Yu and K. Grauman, "Fine-grained visual comparisons with local learning," in *IEEE Conference on Computer Vision and Pattern Recognition*, Jun 2014.

[50] H. Zhang, M. Cissé, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *International Conference on Learning Representations*, 2018.

[51] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.