Bridging the Gap between Label- and Reference-based Synthesis in Multi-attribute Image-to-Image Translation

Qiusheng Huang[1], Zhilin Zheng[2], Xueqi Hu[1], Li Sun[1]*, Qingli Li[1]
[1]Shanghai Key Laboratory of Multidimensional Information Processing, East China Normal University
[2]PingAn Technology

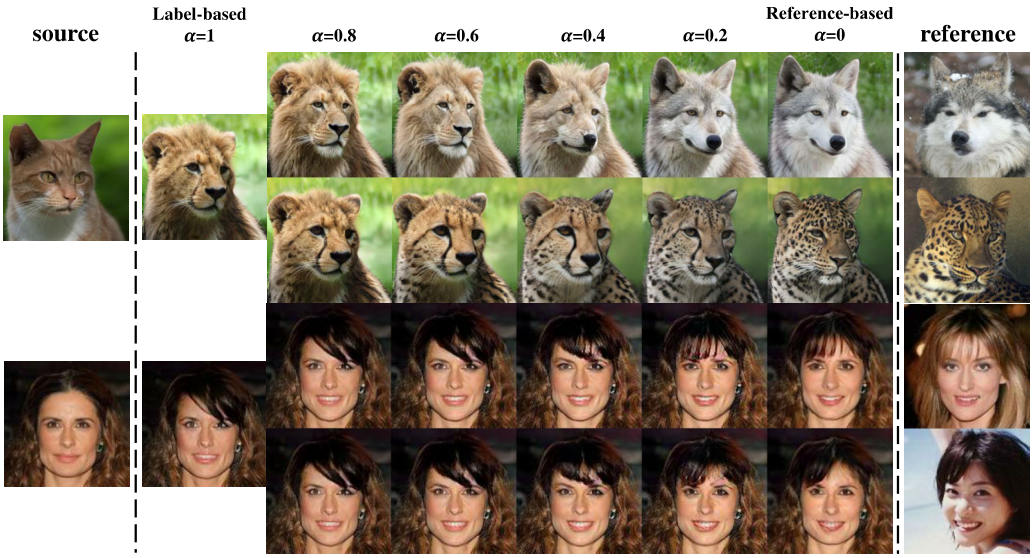

Figure 1: **Interpolation results between label- and reference-based synthesis.** For the species translation, a cat face can be turned into other wild animals. For the face editing, the model converts "bangs" and "mouth open" at the same time. α is the interpolation rate. $\alpha = 1$ means the result is entirely label-based, while $\alpha = 0$ takes the style from the reference.

Abstract

The image-to-image translation (I2IT) model takes a target label or a reference image as the input, and changes a source into the specified target domain. The two types of synthesis, either label- or reference-based, have substantial differences. Particularly, the label-based synthesis reflects the common characteristics of the target domain, and the reference-based shows the specific style similar to the reference. This paper intends to bridge the gap between them in the task of multi-attribute I2IT. We design the label- and reference-based encoding modules (LEM and REM) to compare the domain differences. They first transfer the source image and target label (or reference) into a common embedding space, by providing the opposite directions through the attribute difference vector. Then the two embeddings are simply fused together to form the latent code S_{rand} (or S_{ref}), reflecting the domain style differences, which is injected into each layer of the generator by SPADE. To link LEM and REM, so that two types of results benefit each other, we encourage the two latent codes to be close, and set up the cycle consistency between the forward and backward translations on them. Moreover, the interpolation between the S_{rand} and S_{ref} is also used to synthesize an extra image. Experiments show that label- and reference-based synthesis are indeed mutually promoted, so that we can have the diverse results from LEM, and high quality results with the similar style of the reference. Code will be available at https://github.com/huangqiusheng/BridgeGAN.

1. Introduction

Image-to-image translation (I2IT) aims to learn mapping functions among different domains. These domains are ei-

*Corresponding author, email: sunli@ee.ecnu.edu.cn. This work is supported by the Science and Technology Commission of Shanghai Municipality (No.19511120800).

ther defined by a single attribute [17, 44], therefore, they are mutually exclusive, *e.g.* changing a cat face into a dog. Or they are specified by multiple attributes, so one domain may be overlapped with others with respect to a different attribute, *e.g.* the hair color and the gender are different attributes. Naturally, a domain of black hair has intersection with the male domain. An ideal I2IT model should be able to change the source images into the required target domain, while keeping the content of the source without excessive modifications. For multi-attribute I2IT, since the model needs to complete the translations according to multiple source-to-target requirements, it becomes important it has the ability to accurately edit the individual attribute-related domain, while making other unrelated domains stable. In practice [7, 22], the intended domain labels usually participate the generation, so we name such results the label-based synthesis.

On the other hand, images are often with various styles even if they are in the same domain. *E.g.*, bangs or beards may look quite different. Therefore, it is expected to synthesize multiple results within the same domain, and the styles of them must be under users' control. By providing a reference and asking the model to synthesize an image in the similar appearance with it, we can get diverse multi-modal results [46, 16, 21, 8]. However, the reference-based synthesis is difficult to be realized in the multi-attribute I2IT. Most of the existing works only deal with a single attribute, which means that one domain is not overlapped with others. Only a few works [6, 37, 25] aim at the multi-attribute setting, but their results are often poor, compared with the label-based synthesis.

This paper aims to build a single model to bridge the gap between the label- and reference-based synthesis for the multi-attribute I2IT, as is shown in Fig.2(a). Primarily, our model translates the source image X_s into the target domain, through either the label difference vector att_{diff}, or the reference image X_r lying in a domain different from the source. The results X_g^l from the former (label-based synthesis) are usually of high quality and in the correct required domain. But they have only a single mode and lack diversity. The latter X_g^r (reference-based synthesis) can potentially generate multi-modal images, but are often of low quality and in the wrong domain. Our idea is to utilize the two types of synthesis, and make them guide each other, so that the final results from both of them get promoted.

Specifically, we design two units which are Label- and Reference-based Encoding Module, referred as LEM and REM in Fig.2(a). Their outputs are given to the common main branch of the generator G to synthesize X_g^l and X_g^r. Both modules have two branches. In REM, they process the source X_s and reference X_r, respectively. One branch encodes X_s into a latent code along the direction specified by the difference between source and target domain labels, while the other encodes X_r in the similar way, but in the opposite direction. The two branches intend to find a pair of latent codes from X_s and X_r, respectively. And they are all used by G, which are finally translated into a target domain X_g^r, with its style being similar to X_r. The LEM mimics the design of REM. It also encodes X_s in the direction of the target domain, in the same way as REM. However, since it has no specific reference as the input, we sample a random noise vector to replace it, and design a separate module to map the noise into the latent code. The results from both branches are fused together, and given to G for the X_g^l.

To further bridge the performance gap between the results of X_g^l and X_g^r, the model not only outputs the two of them, but also a translated image based on the interpolation of the two latent codes from the LEM and REM. Moreover, we design a constraining loss directly on the two codes. Intuitively, both of them are used by the common module G to translate the same X_s, so they should be close to each other. To better connect the LEM and REM, and encourage the X_g^l to have diverse styles, we explicitly assign a unique noise vector to each reference X_r during training, and minimize the distance between two latent codes from LEM and REM for the same pair of X_s and X_r. In addition, X_g^l and X_g^r are fed back into the REM as a reference or a source, therefore, the cycle consistency can be applied. We do extensive ablations on each objective loss terms. Fig.1 shows our impressive visual results.

2. Related Works

I2IT for a single attribute. The topic of I2IT is first proposed in [17]. The pix2pix and its later version pix2pixHD [34] are made of autoencoders, which translate images between two domains based on paired data. CycleGAN [44] extends it to the unpaired data. However, these models only generate single-modal results.

To encourage diverse styles, one way is to change the latent code from deterministic to probabilistic, usually achieved by VAE [42, 18, 3, 14, 33, 9, 43, 39, 5]. BicycleGAN [46] gives the multi-modal results by employing the VAE structure. UNIT [23] uses two VAE encoders, mapping images from different domains into a shared space. MUNIT [16] and DRIT [20, 21] disentangle between the content and the style code to encourage the diverse styles. They also add an extra encoder, specifying a style code which is injected into the generator by AdaIN [15]. With a similar structure, FUNIT [24] extends previous works in the few shot scenarios, and the model can work for multiple domains, with each one having only a few examples. StarGAN-V2 [8] and TUNIT [2] also aim at multi-domain translation, but they can not perform well in the multi-attribute setting.

I2IT for multi-attributes. Domains defined by different attributes are inevitably overlapped with each other. Star-

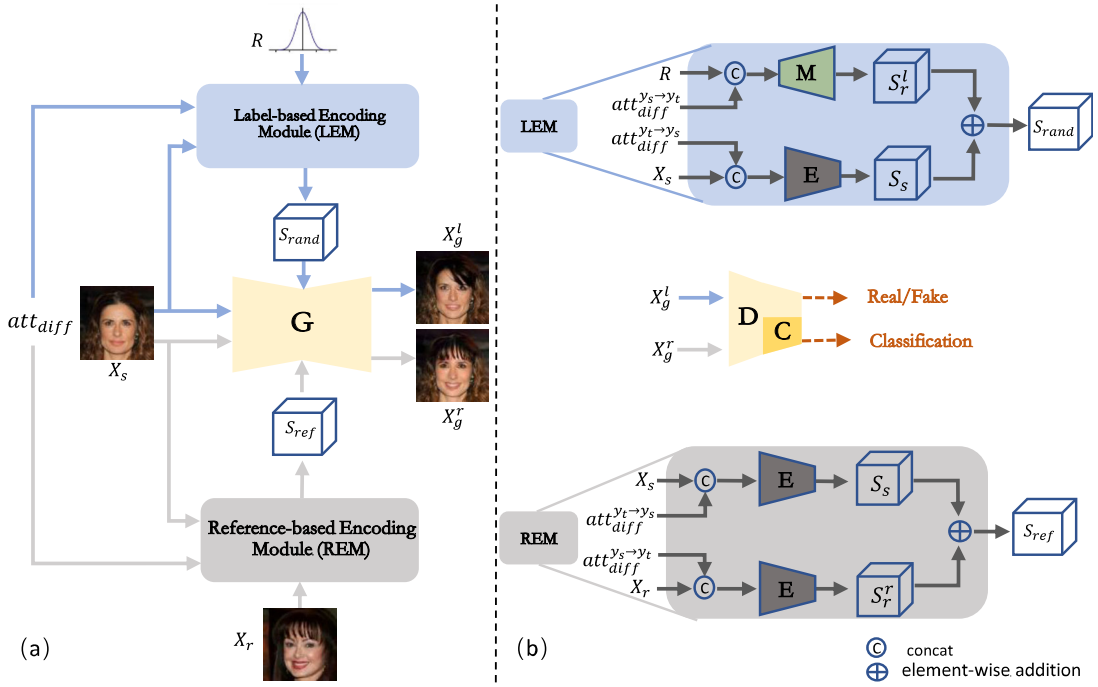

Figure 2: **The overall structure based on LEM and REM.** On the left, we give the overview of our model. On the right, details about LEM and REM are provided. All encoders (E) share the model parameters.

GAN [7] and AttGAN [12] are two similar models for label-based synthesis. The source image and the required labels are processed together to give the results. STGAN [22] refines the structure by setting up connections from encoder to decoder. RelGAN [35] uses the label difference to replace the source and target labels. However, results from them are single-modal. To increase the diversity, SMIT [31] simply incorporates the random noises on the style code to augment the label-based synthesis, making a tradeoff between the quality and diversity. ELEGANT [37] and HomoGAN [6] are pure reference-based models. Some works like GMM-UNIT [25] and DMIT [40] build a single model to support both the label- or reference-based synthesis by introducing the probabilistic encoder. The two types of synthesis are obtained by sampling from the prior or the posterior. Nonetheless, modeling distribution for every domain is difficult particularly when considering large number of attributes.

3. Proposed Method

3.1. Problem Formulation

Our model aims to translate an image $X_s \in \mathbb{R}^{H \times W \times 3}$, with its multi-attribute binary label $Y_s \in \{0,1\}^n$, into an image X_g in a different domain specified by a target label $Y_t \in \{0,1\}^n$. The reference image X_r is optionally provided during the inference, specifying a particular target domain style for X_g. Note that this is a typical unpaired generation task in which we do not have the groundtruth for X_g during training. Here n is the number of the attributes, and each one defines two non-overlapped visual domains, meaning with or without a specific attribute. In total, there are 2^n different domains. $\text{att}_{diff}^{Y_s \to Y_t} \in \{-1, 0, +1\}^n = Y_t - Y_s$ is also an n element vector, representing the direction from source to target. It is employed by the LEM and REM as the input condition. Fig.2 illustrates the specific architecture of our model, consisting of a mapping network M, an encoder network E, a generator network G and a discriminator D with an extra multi-attribute domain classifier C [29]. The two types of synthesis X_g^l and X_g^r are built on LEM and REM modules, respectively. In summary, given following inputs: an image pair X_s and X_r, a noise vector R, and two opposite directions $\text{att}_{diff}^{Y_s \to Y_t}$ and $\text{att}_{diff}^{Y_t \to Y_s}$, the LEM and REM are designed to output the latent codes for the label- and reference-based synthesis, X_g^l and X_g^r.

3.2. Pipelines for Two Types of Synthesis

The two modules, LEM and REM, support the two types of synthesis X_g^l and X_g^r by injecting their outputs S_{rand} and S_{ref} into G. They essentially compare the two inputs from different domains, and encode their differences into a style code. Note that both modules are composed of two branches, where each branch maps its input into an inter-

mediate latent code, and then they are combined together. These processes are summarized in (1) and (2). Details are illustrated in following subsections.

$$\begin{aligned} S_r^l &= \mathrm{M}(R, \mathrm{att}_{diff}^{Y_t \to Y_s}) \quad S_r^r = \mathrm{E}(X_r, \mathrm{att}_{diff}^{Y_t \to Y_s}) \\ & \qquad\qquad S_s = \mathrm{E}(X_s, \mathrm{att}_{diff}^{Y_s \to Y_t}) \end{aligned} \tag{1}$$

$$\begin{aligned} S_{rand} &= \mathrm{LEM}(X_s, R, \mathrm{att}_{diff}) = S_s + S_r^l \\ S_{ref} &= \mathrm{REM}(X_s, X_r, \mathrm{att}_{diff}) = S_s + S_r^r \end{aligned} \tag{2}$$

LEM for label-based synthesis. The mapping network M encodes the random noise $R \in \mathbb{R}^d$ together with $\mathrm{att}_{diff}^{Y_t \to Y_s}$, and gradually increases the spatial size until $S_r^l \in \mathbb{R}^{\frac{H}{k} \times \frac{W}{k} \times C}$. In practice, we concatenate R with $\mathrm{att}_{diff}^{Y_t \to Y_s}$ before giving it to M, as is shown in (1). Similarly, the source X_s is encoded by E in the opposite direction of $\mathrm{att}_{diff}^{Y_s \to Y_t}$ to form another intermediate code S_s with the same size as S_r^l. Then, S_r^l and S_s are added together to form S_{rand} like (2), which reflects the domain style differences on the specified attributes. In terms of purpose, this is equivalent to using the same att_{diff} to generate S_s and S_r^l, and then get S_{rand} by $|S_r^l - S_s|$. The synthesis X_g^l based on S_{rand} shows the common characteristics in the target domain, but it often lacks diversity even if we can sample a random R as the input. So we need REM to take the effect and give the multi-modal results. Furthermore, we emphasize that incorporating with $\mathrm{att}_{diff}^{Y_s \to Y_t}$ can help to locate the attributes to be transferred, at the same time, maintain the remaining attributes specified by 0 in att_{diff}.

REM for reference-based synthesis. REM has the same structure as LEM to process X_s, mapping it into S_s by E in the direction of $\mathrm{att}_{diff}^{Y_s \to Y_t}$. As is shown in (1), E also encodes X_r to get a code S_r^r. Then, S_r^r and S_s are added to form S_{ref} like in (2). Note that the result X_g^r based on S_{ref} not only lies in the target domain but also has the similar style as X_r.

The generator G for two types of synthesis. S_{rand} and S_{ref} are employed by a common G to give the final results X_g^l and X_g^r, respectively. G is an auto-encoder, which first encodes X_s into an embedding space, then decodes it back into an image. S_{rand} and S_{ref} are employed by the main branch of decoder G through SPADE [30]. Note that the parameters of G and SPADE are shared for S_{rand} and S_{ref}.

3.3. Training Objectives

Noise processing and hidden layer objective. To further bridge the gap between the label- and reference-based synthesis, we intend to link the implicit $R \sim N(0, I)$ in LEM and the explicit reference X_r in REM. Particularly, we allocate a random R for each training sample X_r and make them into pairs $\{R, X_r\}$. During training, the pairs keep fixed. We use the constraint defined in (3) for the optimization.

$$L_{sty} = \| S_{rand} - S_{ref} \|_1 \tag{3}$$

Inspired by [36, 4], we adopt a two-step strategy. In the first step, model parameters in LEM and REM are fixed, only R gets updated. Then, the revised R is given to LEM again for the new S_{rand}. The new L_{sty} is computed to update the parameters in LEM and REM. This penalty allows LEM to learn different attribute styles and improve diversity. It also makes the conversion of REM more accurately.

Adversarial objective. We employ the adversarial loss [10] for the generation fidelity, formulated as (4).

$$L_{adv} = \mathbb{E}_{X_s \sim p_d}[\mathrm{D}(X_s)] - \mathbb{E}_{X_g \in \{X_g^l, X_g^r, X_g^i\}}[\mathrm{D}(X_g)] \tag{4}$$

Here D is the discriminator constrained by 1-Lipschitz continuity following WGAN [1] and WGAN-GP [11]. Besides X_g^l and X_g^r, we randomly interpolate between S_{rand} and S_{ref}, and give the results to G for X_g^i,

$$X_g^i = \mathrm{G}(X_s, (\alpha S_{rand} + (1 - \alpha) S_{ref})), \tag{5}$$

where α is a scalar and $\alpha \sim U(0, 1)$.

Attribute classification. We employ a classifier C to ensure that generated images have accurate attributes [28], formulated as (6).

$$L_{cls} = -\frac{1}{N_C} \sum_{i=0}^{N_C} [y_i \log \mathrm{C}_i(X) + (1 - y_i) \log(1 - \mathrm{C}_i(X))] \tag{6}$$

Here N_C is the number of attributes. X is an image, including the translations X_g^l, X_g^r and X_g^i, and the real image X_s. C_i is the classifier that predicts the i^{th} attribute of X. y_i is the i^{th} value of attribute label Y.

Source reconstruction. We adopt the reconstruction loss in (7) as a regularization.

$$L_{rec} = \| X_s - \mathrm{G}(X_s, \mathrm{LEM}|\mathrm{REM}(X_s, R, 0)) \|_1 \tag{7}$$

By setting $\mathrm{att}_{diff} = 0$, it can ensure that the style information of the generated image comes from LEM or REM.

Latent cycle consistency. We employ a cycle consistency on the latent code S_{rand} and S_{ref}, which can make G utilize the style code S when generating X_g. The idea is to feed back the synthesis X_g^l or X_g^r as the reference input in REM, so that we can have the style code $\tilde{S}$ in (8).

$$L_{cyc} = \| S_{rand} - \tilde{S}_{rand} \|_1 + \| S_{ref} - \tilde{S}_{ref} \|_1, \tag{8}$$

Here S_{rand} and S_{ref} are the style codes for X_g^l and X_g^r, respectively. Both $\tilde{S}_{rand}$ and $\tilde{S}_{ref}$ are computed from REM, to which we feed X_s as the source input, and X_g^l or X_g^r as the reference. The L_{cyc} in (8) reflects the distance between the first and second time style code, which is similar to [8, 45]. Note that this penalty only affects E and M, but not G.

Mode seeking objective. To encourage the images with diverse styles, we use the mode seeking loss [27, 38] in (9),

$$L_{ms} = \frac{1}{\| \mathrm{G}(X_s, S_{rand}) - \mathrm{G}(X_s, S'_{rand}) \|_1} \| R - R' \|_1, \tag{9}$$

where R specifies S_{rand} as (2), and $R' \sim N(0, I)$ is a different input noise vector, giving S'_{rand}.

Attribute keeping constraint. In multi-attribute I2IT model, only the specified attributes need to be translated. The remaining ones are expected to be the same as the source. We use E to extract features for unspecified attributes from both the original X_s and edited X_g^l, constraining them to be close. The formula is as follows.

$$L_{ak} = \| \mathrm{E}(X_s, \mathrm{att}_{ak}^{Y_s\downarrow}) - \mathrm{E}(\mathrm{G}(X_s, R, \mathrm{att}_{diff}^{Y_s\to Y_t}), \mathrm{att}_{ak}^{Y_s\downarrow}) \|_1 \tag{10}$$

In (10), we only extract the features that need to be retained, so we calculate $\mathrm{att}_{ak}^{Y_s\downarrow} \in \{-1, 0, +1\}^n$ in (11), in which attributes without editing from Y_s to Y_t are obtained.

$$\mathrm{att}_{ak}^{Y_s\downarrow} = (1 - 2Y_s)(1 - |\mathrm{att}_{diff}^{Y_s\to Y_t}|). \tag{11}$$

For example, if there are four attributes, given $Y_s = [1, 0, 1, 0]$ and $Y_t = [1, 1, 0, 0]$, we can obtain $\mathrm{att}_{diff}^{Y_s\to Y_t} = [0, 1, -1, 0]$ and $\mathrm{att}_{ak}^{Y_s\downarrow} = [-1, 0, 0, 1]$. In both X_s and X_g^l, the features represented by the first and last attribute are extracted, and they are constrained to be close. The value -1 and 1 in $\mathrm{att}_{ak}^{Y_s\downarrow}$, corresponding to the attributes in Y_s and Y_t are 1 and 0, respectively. Please see the appendix, for a more detailed explanation of (11).

Full objective. Finally, we train our M, E, G, D, C, and r to minimize following objectives.

$$L_{\mathrm{G}} = L_{adv} + \lambda_{cls} L_{cls} + \lambda_{rec} L_{rec} + \lambda_{sty} L_{sty} + \lambda_{ms} L_{ms} + \lambda_{ak} L_{ak}$$
$$L_{\mathrm{ME}} = L_{\mathrm{G}} + \lambda_{cyc} L_{cyc}$$
$$L_{\mathrm{DC}} = -L_{adv} + \lambda_{cls} L_{cls} \qquad L_r = \lambda_{sty} L_{sty}$$

where λ_{cls}, λ_{rec}, λ_{cyc}, λ_{ms}, λ_{sty}, and λ_{ak} are hyperparameters for each term.

4. Experiments

Datasets. When converting multiple attributes simultaneously, a large amount of training data is needed. So, we adopt CelebA [26] to evaluate our method. Fourteen attributes are selected in our experiments, including Young, Mouth Slightly Open, Smiling, Black Hair, Blond Hair, Brown Hair, Gray Hair, Receding Hairline, Bangs, Male, No Beard, Mustache, Goatee, and Sideburns. Besides, 182,000 images are used as the training set, and 19,962 as the testing set. We crop each image to 170×170 and resize it to 128×128. In order to prove the effectiveness of our method on the single attribute and non-face data, we also evaluate it on the AFHQ published by StarGAN-V2, and follow StarGAN-V2's setting. In this part, there are three domains that need to be converted to each other, including cats, dogs, and wild animals. Each domain is provided with 5000 images. These images are resized to 256×256 resolution for training.

Evaluation metrics. We use Frechét inception distance (FID) [13] and Inception Score (IS)[32] to evaluate the visual quality, and evaluate the diversity of generated images by the learned perceptual image patch similarity (LPIPS) [41, 19]. Besides, the domain Accuracy of the generated image is evaluated by a pre-trianed multi-attribute classifier. We compute metrics for all attributes on the test data and report their averages. Please see the appendix for the accuracy of each attribute and other implementation details.

4.1. Qualitative and Quantitative Results

Label-based synthesis. Fig.3 shows representative examples, demonstrating that our method can generate high-quality images, and accurately control the attributes to translate. Moreover, the $9th$ to $12th$ columns in Fig.3 are results of simultaneous conversion of multiple attributes. We find it does not degrade the image quality and change the irrelevant attributes. Fig.4 shows a visual comparison among 4 models, including StarGAN, STGAN, SMIT and ours. Note that since label-based synthesis is relatively easy, all models accomplish the task. However, StarGAN seriously changes the background. Compared with our results, STGAN cannot accurately edit certain attributes while keeping others unchanged. *E.g.*, when changing the hair color, the eyebrow color is altered obviously. When editing the gender, the hair style changes notably. SMIT cannot edit on beards, and the quality for gender conversion is poor.

Tab.1 lists FID, Accuracy, and LPIPS of all competing methods. The performance of StarGAN, STGAN, and SMIT is obviously worse than our method (model G). As we all know, for label-based synthesis, the diversity and attribute accuracy of the generated images are conflicting to a certain extent. Few works can pursue both at the same time except ours. In fact, StarGAN-V2 achieve good results in face conversion, but it cannot complete the task of multi-attribute conversion, so we cannot compare with it.

Reference-based synthesis. In Fig.5, we visually compare the results of HomoGAN, ELEGANT and ours. Obviously, HomoGAN achieves the low quality, and cannot keep irrelevant factors such as background and skin color. ELEGANT has high quality but often fails to change appropriately. In our model, LEM can assist REM to locate the relevant attributes more accurately. Tab.1 also lists the quantitative metrics, including FID and Accuracy. Our model outperforms the HomoGAN and ELEGANT undoubtedly on these two metrics.

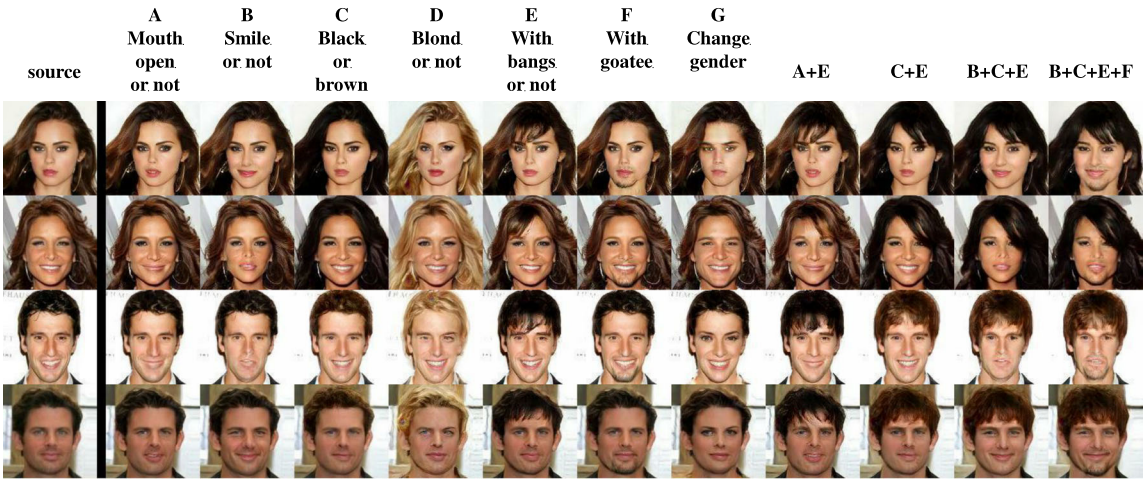

Figure 3: **Label-based synthesis of our model.** The last 4 columns are results by editing more than one attributes.

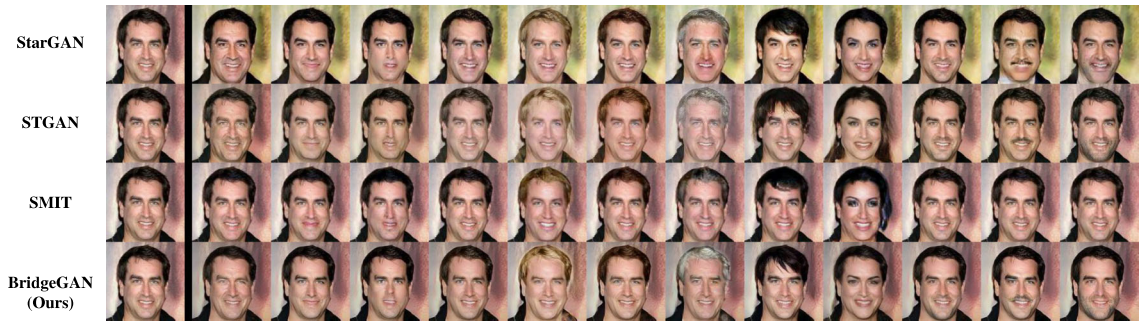

Figure 4: **Label-based synthesis of 4 models.** From left to right: source, young, mouth slightly open, smiling, black, blond, brown, gray hair, bangs, male, no beard, mustache, and sideburns.

We further test our model on the synthesis from multiple references. Particularly, we use the encoder E to extract the features S_r^r from different references, and average on S_r^r to give the result, as is shown in Fig.7. The model succeeds taking the relevant attributes from different references. In addition, we also perform the feature mixing on S_r^r. Here, S_r^r is mixed along the image height by the original codes from different references, and the results is shown in Fig.8. The model also takes the relevant attributes and their styles from corresponding references.

User Study. In Fig.9, we conduct a user study to evaluate different models under human perception. For label-based synthesis, we randomly choose images to translate for each attribute. Users are asked to select the best editing result from all competing methods. For reference-based synthesis, we randomly generate translated results . Users are required to choose the best among all competing methods, according to whether the converted attributes between the synthesis and the reference image are similar, and whether other irrelevant attributes remain the same.

Experiments on single attribute. In Fig.6, we show the visually results on the AFHQ datasets. So far, StarGAN-V2 seems to be the best in the AHFQ, so we only compared with it. In comparison, we are better than StarGAN-V2 in maintaining the background information of the original image. In terms of the similarity of the reference image, our results show a higher degree of stability. In addition, Tab.2 lists FID, IS and LPIPS of all methods. On the one hand, we achieved lower FID score and higher IS in the quality of images. On the other hand, with similar LPIPS scores, we can maintain background information to a certain extent. This proves that our method can capture the characteristics of domains and edit more accurately.

Interpolation results. In Fig.1, we use the method in Eq.(5) to synthesize the interpolation between the two types of latent codes, S_{rand} and S_{ref}. The interpolated images between the two methods are natural and achieve smooth transitions. Our model has the ability to give diverse results with the specific domain style linearly approaching to the reference.

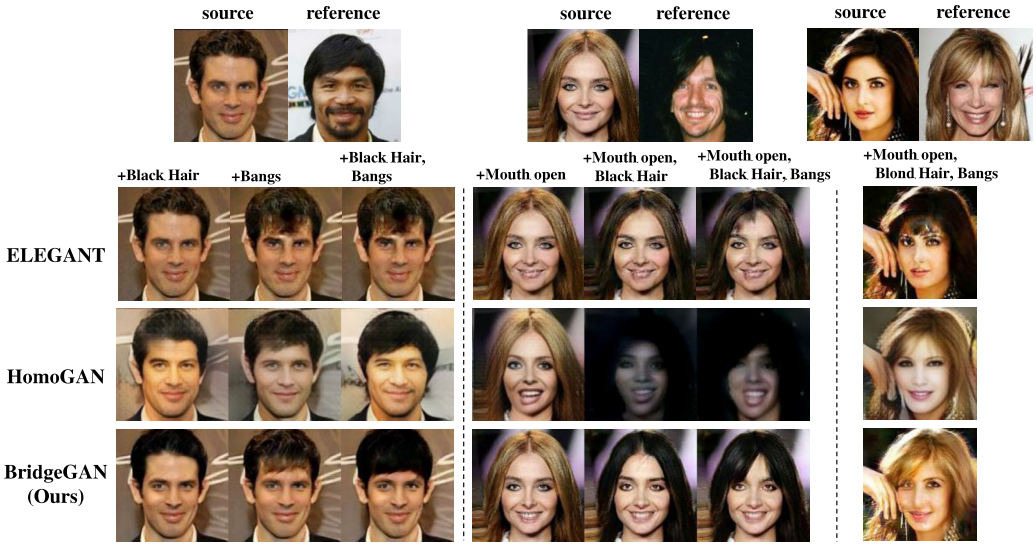

Figure 5: **Reference-based synthesis results.** We show 3 data pairs, and their results on single or multiple attributes editing.

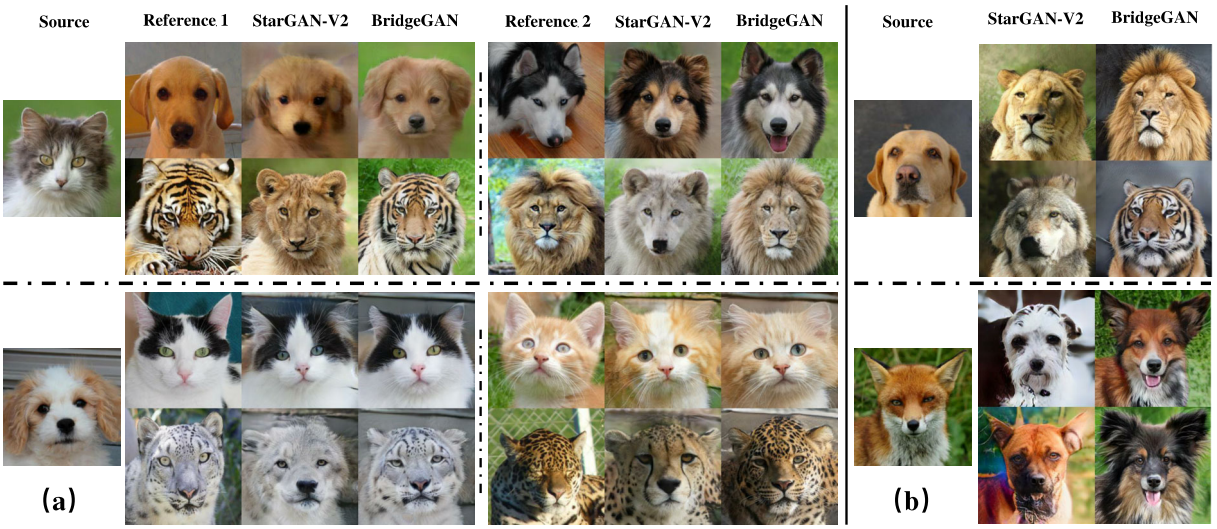

Figure 6: **Qualitative comparison of reference-based and label-based synthesis results on the AFHQ datasets.** On the left, we show the reference-based synthesis of different methods, each source image is equipped with 4 reference images with different styles. On the right, we show the label-based synthesis results, including dog-to-wild, and wild-to-dog. Each source image is equipped with two different sampling noise vectors. Please zoom in for more details.

4.2. Ablation Study

Tab.1 lists the metrics for ablation study. The baseline is the model A, in which the network has only the LEM (OLEM). Therefore, we remove the whole REM and the S_s in LEM, which means $S_{rand} = S_r^l$. We use L_{adv} in (4), L_{cls} in (6) and L_{rec} (7) for training. Intuitively, we can also have a model with only the REM (OREM) for reference-base synthesis. But such a model can not take the source content.

Then we add individual component to A. In model B, we combine the OLEM with the OREM, and add L_{cyc} into the objectives. Obviously, once we start to establish the connection between LEM and REM, the reference-based result becomes better. Then we add the encoder E to give S_s in model C. This means that the domain difference between the original source and the reference is constructed

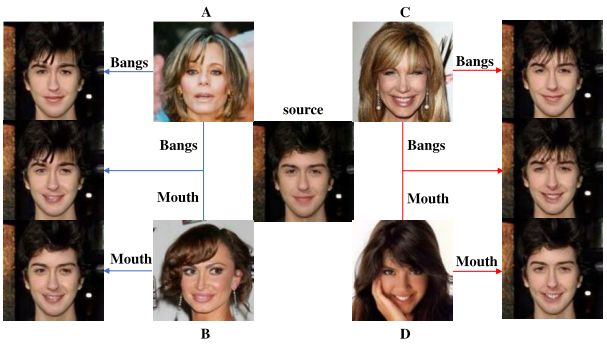

Figure 7: **Reference-based multi-attribute editing by latent code averaging on S_r^r.** The results on the left are provided by reference image A and B, while the right are provided by C and D.

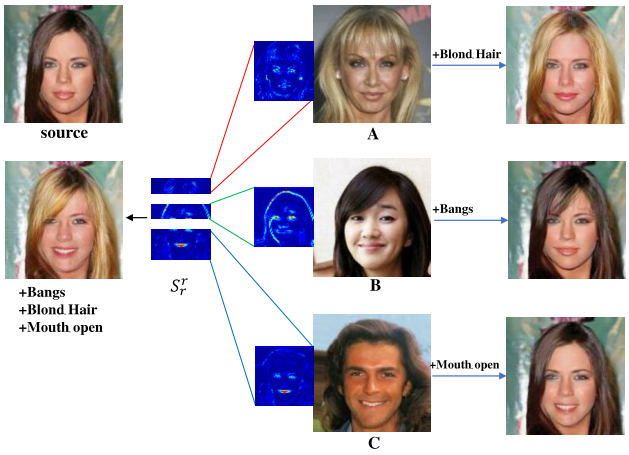

Figure 8: **Reference-based multi-attribute editing by latent code mixing.** On the left, we mix S_r^r from 3 different references (A: blond hair, B: bangs, C: mouth open). On the right, we take a single reference and make the translation.

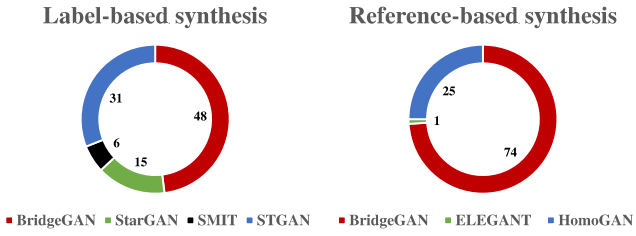

Figure 9: **User study results of two types of synthesis.**

by S_{rand} and S_{ref} in the hidden layer. The metric of FID shows it greatly improves image quality while keeping others the same. We further add the interpolation image X_g^i to L_{adv} for training in model D. It increases the Accuracy

method	FID↓		Accuracy↑		LPIPS↑
StarGAN [7]	25.96	-	69.5	-	-
STGAN [22]	16.11	-	80.6	-	-
SMIT [31]	**12.14**	-	27.4	-	0.029
ELEGANT [37]	-	68.88	-	26.6	-
HomoGAN [6]	-	17.96	-	30.5	-
A:OLEM	25.10	-	84.3	-	0.020
B:+OREM	17.07	13.91	83.5	47.6	0.017
C:+S_s	**12.14**	9.52	82.8	47.5	0.006
D:+interp	17.32	10.09	86.2	42.1	0.012
E:+L_{ms}	14.03	**9.28**	82.9	42.7	0.030
F:+L_{sty}	16.82	12.95	85.1	39.5	0.034
G:+L_{ak}	14.26	11.38	**87.0**	**54.0**	**0.043**

Table 1: **Quantitative comparisons and ablation studies by the metrics.** For FID and Accuracy we measure them on two types of synthesis. On the left of the separator | is the value of label-based synthesis, while the right side is reference-based. For LPIPS, we only evaluate on the probabilistic models by random sampling on the noise input.

method	FID↓		LPIPS↑		IS↑
StarGAN-V2 [8]	31.07	33.32	0.478	**0.437**	4.112
BridgeGAN	**25.87**	**26.68**	**0.479**	0.432	**4.666**

Table 2: **Quantitative comparisons on the AFHQ datasets by the metrics.** We measure these on two types of synthesis, the result of IS is the average of them. In order to measure the image quality in general, we use the real images of the test set to calculate the FID.

and LPIPS for label-based synthesis. For diverse results, we add L_{ms} in (9) in the setting E, and LPIPS for label-based synthesis is improved greatly. Moreover, this loss has a good impact on the reference-based synthesis. We then add L_{sty} in (3) to model F to give a tight link between LEM and REM. Compared to method E, Accuracy and LPIPS of Label-based synthesis are further improved. Finally, we add L_{ak} (10) in the last model G, which aims to keep irrelevant attributes from being converted. Note that it can guide both LEM and REM, particularly it gives a higher Accuracy.

5. Conclusion

This paper constructs a new architecture for multi-attribute I2I translation. Our model bridges the gap between label- and reference-based synthesis, so that both of them get improved. For the label-based image synthesis, we can simultaneously obtain diverse and high-accuracy translated images. For the reference-based synthesis, our model is able to take the specific style similar to the reference. The results show that the proposed model remarkably outperforms the previous ones.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.

[2] Kyungjune Baek, Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Hyunjung Shim. Rethinking the truly unsupervised image-to-image translation. *arXiv preprint arXiv:2006.06500*, 2020.

[3] Jianmin Bao, Dong Chen, Fang Wen, Houqiang Li, and Gang Hua. Cvae-gan: fine-grained image generation through asymmetric training. In *Proceedings of the IEEE international conference on computer vision*, pages 2745–2754, 2017.

[4] Sam Bond-Taylor and Chris G Willcocks. Gradient origin networks. *arXiv preprint arXiv:2007.02798*, 2020.

[5] Ricky TQ Chen, Xuechen Li, Roger Grosse, and David Duvenaud. Isolating sources of disentanglement in variational autoencoders. *arXiv preprint arXiv:1802.04942*, 2018.

[6] Ying-Cong Chen, Xiaogang Xu, Zhuotao Tian, and Jiaya Jia. Homomorphic latent space interpolation for unpaired image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2408–2416, 2019.

[7] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul 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*, pages 8789–8797, 2018.

[8] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8188–8197, 2020.

[9] Patrick Esser, Ekaterina Sutter, and Björn Ommer. A variational u-net for conditional appearance and shape generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8857–8866, 2018.

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27, pages 2672–2680. Curran Associates, Inc., 2014.

[11] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, pages 5767–5777. Curran Associates, Inc., 2017.

[12] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. Attgan: Facial attribute editing by only changing what you want. *IEEE Transactions on Image Processing*, 28(11):5464–5478, 2019.

[13] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, pages 6626–6637. Curran Associates, Inc., 2017.

[14] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. 2016.

[15] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017.

[16] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 172–189, 2018.

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

[18] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[19] Alex Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[20] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. In *Proceedings of the European conference on computer vision (ECCV)*, pages 35–51, 2018.

[21] Hsin-Ying Lee, Hung-Yu Tseng, Qi Mao, Jia-Bin Huang, Yu-Ding Lu, Maneesh Singh, and Ming-Hsuan Yang. Drit++: Diverse image-to-image translation via disentangled representations. *International Journal of Computer Vision*, pages 1–16, 2020.

[22] Ming Liu, Yukang Ding, Min Xia, Xiao Liu, Errui Ding, Wangmeng Zuo, and Shilei Wen. Stgan: A unified selective transfer network for arbitrary image attribute editing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3673–3682, 2019.

[23] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. In *Advances in neural information processing systems*, pages 700–708, 2017.

[24] Ming-Yu Liu, Xun Huang, Arun Mallya, Tero Karras, Timo Aila, Jaakko Lehtinen, and Jan Kautz. Few-shot unsupervised image-to-image translation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 10551–10560, 2019.

[25] Yahui Liu, Marco De Nadai, Jian Yao, Nicu Sebe, Bruno Lepri, and Xavier Alameda-Pineda. Gmm-unit: Unsupervised multi-domain and multi-modal image-to-image translation via attribute gaussian mixture modeling. *arXiv preprint arXiv:2003.06788*, 2020.

[26] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. 2014.

[27] Qi Mao, Hsin-Ying Lee, Hung-Yu Tseng, Siwei Ma, and Ming-Hsuan Yang. Mode seeking generative adversarial networks for diverse image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[28] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *Computer ence*, pages 2672–2680, 2014.

[29] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *International conference on machine learning*, pages 2642–2651. PMLR, 2017.

[30] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[31] Andrés Romero, Pablo Arbeláez, Luc Van Gool, and Radu Timofte. Smit: Stochastic multi-label image-to-image translation. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[32] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *arXiv preprint arXiv:1606.03498*, 2016.

[33] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28:3483–3491, 2015.

[34] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.

[35] Po-Wei Wu, Yu-Jing Lin, Che-Han Chang, Edward Y Chang, and Shih-Wei Liao. Relgan: Multi-domain image-to-image translation via relative attributes. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5914–5922, 2019.

[36] Yan Wu, Jeff Donahue, David Balduzzi, Karen Simonyan, and Timothy Lillicrap. Logan: Latent optimisation for generative adversarial networks. *arXiv preprint arXiv:1912.00953*, 2019.

[37] Taihong Xiao, Jiapeng Hong, and Jinwen Ma. Elegant: Exchanging latent encodings with gan for transferring multiple face attributes. In *Proceedings of the European conference on computer vision (ECCV)*, pages 168–184, 2018.

[38] Dingdong Yang, Seunghoon Hong, Yunseok Jang, Tianchen Zhao, and Honglak Lee. Diversity-sensitive conditional generative adversarial networks. *CoRR*, abs/1901.09024, 2019.

[39] Mingyu Yin, Li Sun, and Qingli Li. Novel view synthesis on unpaired data by conditional deformable variational auto-encoder. In *European Conference on Computer Vision*, pages 87–103. Springer, 2020.

[40] Xiaoming Yu, Yuanqi Chen, Shan Liu, Thomas Li, and Ge Li. Multi-mapping image-to-image translation via learning disentanglement. In *Advances in Neural Information Processing Systems*, pages 2994–3004, 2019.

[41] 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 (CVPR)*, June 2018.

[42] Ziye Zhang, Li Sun, Zhilin Zheng, and Qingli Li. Disentangling the spatial structure and style in conditional vae. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 1626–1630. IEEE, 2020.

[43] Zhilin Zheng and Li Sun. Disentangling latent space for vae by label relevant/irrelevant dimensions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12192–12201, 2019.

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

[45] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Multimodal image-to-image translation by enforcing bi-cycle consistency. In *Advances in neural information processing systems*, pages 465–476, 2017.

[46] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *Advances in neural information processing systems*, pages 465–476, 2017.