Continuous and Diverse Image-to-Image Translation via Signed Attribute Vectors

Qi Mao · Hsin-Ying Lee · Hung-Yu Tseng · Jia-Bin Huang · Siwei Ma · Ming-Hsuan Yang

Abstract Recent image-to-image (I2I) translation algorithms focus on learning the mapping from a source to a target domain. However, the continuous translation problem that synthesizes intermediate results between the two domains has not been well-studied in the literature. Generating a smooth sequence of intermediate results bridges the gap of two different domains, facilitating the morphing effect across domains. Existing I2I approaches are limited to either intra-domain or deterministic inter-domain continuous translation. In this work, we present an effective signed attribute vector, which enables continuous translation on diverse mapping paths across various domains. In particular, utilizing the sign operation to encode the domain information, we introduce a unified attribute space shared by all domains, thereby allowing the interpolation on attribute vectors of different domains. To enhance the visual quality of continuous translation results, we generate a trajectory between two sign-symmetrical attribute vectors and leverage the domain information of the interpolated results along the trajectory for adversarial training. We evaluate the proposed method on a wide range of I2I translation tasks. Both qualitative and quantitative results demonstrate that the proposed framework generates more high-quality continuous translation results against the state-of-the-art methods.

1 Introduction

Image-to-Image (I2I) translation [16] aims to learn the mapping function between different visual domains. It can be applied to a wide range of tasks such as semantic image synthesis [33,30], photo enhancement [40], style transfer [23,10], season transfer [15,22,37], domain adaptation [13,6], and object transfiguration [36,8,26,9]. Given an image from the source domain, we can render it into an image with the target domain’s style. However, it is unclear whether one can generate smooth and continuous translated images using existing methods designed for single images. As shown in Fig. 1, continuous translation enables applications such as image morphing. Furthermore, modeling the intermediate results facilitates a better understanding of the translation process between the two domains.

Recent I2I translation approaches mainly focus on two multi-modal and multi-domain translation. However, these methods are limited in rendering continuous translated images across domains. Existing multi-modal methods [41,15,22] use a domain-specific attribute representation to model the image variation of a particular domain. While these algorithms can generate continuous translation by performing linear interpolation between different attribute vectors, such schemes are limited to intra-domain due to separate attribute spaces of different domains. On the other hand, multi-
domain methods [8, 35] map the image to multiple domains by taking an additional domain label as input. Several recent approaches [10, 35] generate intermediate results by performing interpolation between two discrete domain labels to handle continuous translation. Nevertheless, since such an interpolation is conducted on the domain label (rather than attribute vectors), it can only produce a deterministic translation path. Namely, these approaches synthesize an identical continuous translation result for the same input image given two particular domains.

In this paper, we focus on continuous image-to-image translation on multiple mapping paths (multi-modal) across various visual domains (multi-domain) and address two challenging issues. First, previous approaches such as MUNIT [15] and DRIT [23] adopt separate attribute spaces for different domains, thus cannot model the continuous variation across domains. Second, there are no ground-truth intermediate samples for learning continuous translation results in-between two domains.

To address the issues mentioned above, we propose a signed-attribute-vector-based image-to-image translation framework (SAV-I2I). We disentangle images into the content and the attribute representations by a content encoder and an attribute encoder. To enable inter-domain continuous translation, we introduce a unified attribute space containing domain-specific attributes of all domains. We consider each attribute dimension from the prior Gaussian distribution as independent and identically distributed random variables and draw samples. Then, the sign operation is applied to make the values of the attributes in a particular domain positive and those in other domains negative. The proposed signed attribute vectors (SAVs) and the content representations are fed into the generator to synthesize corresponding domain images. Furthermore, we adopt the maximum mean discrepancy (MMD) constraint to align the distribution of SAVs with the attribute encoder embedding distribution. Thus, the attribute representation can either be sampled from the signed attribute space or extracted by the given reference image.

There are two advantages to the proposed method. First, it facilitates continuous translation across various domains within a unified attribute space. Second, owing to the sign information, we propose a translation trajectory between the SAV of the source domain and its sign-symmetrical attribute vector of the target domain. We leverage the domain information on interpolated results along the trajectory during the training and apply the adversarial loss to enhance the quality.

The main contributions of this work are summarized as follows:

- We propose a simple yet effective SAV to construct a unified attribute space for all domains, enabling continuous and diverse translation paths across various visual domains.
- We design a sign-symmetrical operation to create a translation trajectory between two domains during training. By leveraging the domain information of intermediate results along the trajectory, we apply the adversarial loss to enhance the quality.
- Extensive experiments validate that the proposed method can synthesize continuous and diverse translation results on a wide range of I2I translation tasks. Qualitative and quantitative results demonstrate that the proposed framework generates more
Fig. 2: **Comparisons of multi-modal translation methods.** Denote $x_{y_i}$ as the image in domain $y_i$, the flow of encoder and generator are represented in the solid lines and dashed lines, respectively. The same module uses lines with the same color.

Table 1: **Comparisons of image-to-image translation networks.** Our model renders continuous translation on diverse mapping paths across multiple domains.

| Methods             | MUNIT [15] | DRIT [22] | DMIT [37] | DRIT++ [23] | StarGAN-v2 [9] | DLOW [10] | Ours |
|---------------------|------------|-----------|-----------|-------------|---------------|-----------|------|
| Multi-modal         | ✓          | ✓         | ✓         | ✓           | ✓             | -         | ✓    |
| Multi-domain        | -          | -         | ✓         | ✓           | ✓             | ✓         | ✓    |
| Continuous translation | -        | -         | -         | -           | -             | ✓         | ✓    |

High-quality continuous translation results against the state-of-the-art approaches.

### 2 Related Work

**Image-to-image translation.** Image-to-image (I2I) translation [16] aims to learn the mapping images between different domains. Isola et al. [16] propose Pix2Pix to address the problem with paired data. Numerous I2I methods [40, 25, 19] exploit the idea of cycle-consistency to train the model with unpaired data. However, these approaches can only perform one-to-one (i.e., uni-modal) mapping translation.

Recent methods achieve one-to-many mapping from different perspectives: multi-modal [41, 15, 22], multi-domain [8, 3, 26], and both [37, 23, 9]. Nevertheless, these schemes focus on generating images of the target distribution, ignoring the continuous translation process that produces intermediate results across domains. Although the DLOW method [10] aims to address continuous translation, it is built upon the multi-modal translation model Cycle-GAN [40] by introducing an additional domain variable, which only synthesizes one deterministic continuous translation path between the source and target domains for the same input image.

To achieve multi-modality, MUNIT [15] and DRIT [22] disentangle images into domain-invariant content representations and domain-specific attribute representations (Fig. 2(a)). However, due to the separate construction of domain-specific attribute space of different domains, they can only perform continuous interpolation within intra-domain. Several approaches, such as DRIT++ [23], DMIT [37], and StarGAN-v2 [9], integrate the attribute (style) representations and domain labels in a single framework for multi-modal and multi-domain translations. Nevertheless, they still do not perform well in continuous translation across multiple domains. The attribute encoder and the generator of DRIT++ [23] require an additional domain label as input (Fig. 2(b)), which prevents interpolation on attribute vectors from different domains. DMIT [23] disentangles images into the domain-invariant content space, the domain-shared style space, and the domain-specific attribute space represented by domain labels (Fig. 2(c)). We note that directly interpolating vectors of domain-shared style space do not lead to continuous translations from the source domain to the target one. Despite that StarGAN-v2 [9] introduces the unified style encoder to encode style information of differ-
ent domains (Fig. 2(d)), the design of multiple embedding branches still separates attribute space of different domains. We note that applying interpolation straightforwardly to style vectors of different domains does not generate smooth translated images, as shown in our experimental results.

In contrast to existing methods, we present a unified attribute representation that contains domain-specific attributes from all domains. Then, the signed attribute vectors are proposed utilizing the sign operation to embed the domain information, as shown in Fig. 2(e), facilitating continuous translation on diverse mapping paths across various domains. Table 1 summarizes the differences between recent unsupervised I2I translation frameworks.

**Facial Attribute Editing.** Facial attribute editing is one application of the image-to-image translation task. Compared to domain labels, it requires fine-grained attribute annotations. Several facial attribute editing approaches can continuously manipulate one specific attribute such as the smile attribute on face images, by varying the value of the annotated attribute label.

Fader networks [21] use binary attribute labels for training. The model can then be generalized to use continuous float values to manipulate the specific facial attribute strength during inference. RelGAN [35] proposes a relative-attribute-based approach for facial attribute editing. Guided by an interpolation discriminator, the framework learns to interpolate a specific attribute during training, leading to a smoother continuous facial attribute manipulation. Instead of linearly interpolating the annotated attribute labels, the recent approach HomoInterpGAN [7] embeds images into latent representations by an encoder and learn an interpolator network to interpolate the latent features. The interpolator network is then trained with a homomorphic loss to make the manipulation of latent representations consistent with that of the annotated attributes.

However, since these approaches require the fine-grained binary attribute annotations during the training stage, they are not directly applicable to the continuous I2I task, which only provides few domain labels. In contrast, our method learns a unified signed attribute space from images of different domains. Even without fine-grained binary attribute labels, it can realize high-quality continuous manipulation by interpolating the learned attribute vectors.

**Image Morphing.** Image morphing [34] aims to change from an image to another through a seamless transition. Existing approaches [34] usually accomplish the task via multiple steps: determining the corresponding mapping between the images on specific feature space [24], applying a 2D geometric transformation to warp the image for retaining geometric alignment on the feature space, interpolating the color space to blend the texture.

With the advances of generative adversarial networks (GANs), numerous methods [1,2] manipulating the latent space of state-of-the-arts GAN models such as StyleGAN [1], StyleGAN-v2 [2], and BigGAN [5] have been demonstrated to be effective for image morphing. In particular, they project images into these pre-trained GAN models’ latent space and then perform the morphing effect by interpolating latent vectors of different images.

Unlike interpolating latent features embedded by pre-trained GANs, we learn to disentangle images into the content and attribute representations. By interpolating attribute vectors of different domains, we generate morphing effects to the target domain while preserving the content of the source image.

### 3 Proposed Method

Our goal is to learn continuous and diverse image-to-image translation across visual domains while preserving the domain-invariant content. As illustrated in Fig. 3(a), given an image $x$, we use a content encoder $E_c$ to obtain the domain-invariant content representation $c$, and a unified attribute encoder $E_a$ to extract the attribute vector $z$. We can then generate continuous translation by interpolating two different attribute vectors. However, existing approaches [15,22,23] are more effective for intra-domain interpolation due to separate attribute spaces for different domains. To enable inter-domain continuous translation, we propose a unified attribute space shared by all domains, denoted as

$$z = [z_1^1, z_1^2, \cdots, z_2^1, z_2^2, \cdots, z_d^1, \cdots, z_2^N, z_2^N, \cdots, z_d^N],$$

$$z \in \mathbb{R}^{d \cdot N},$$

where $d$ is the attribute vector’s dimension for each domain, and $N$ represents the number of visual domains. In the following, we present a sign operation to encode the domain information into the unified attribute vector and present the training strategies for the framework.

#### 3.1 Signed Attribute Vectors

In this work, we encode the domain information into the unified attribute vector. The prior of the translation between two domains lies in one domain that has some more prominent attributes than the other domain.
Given a source image $\mathbf{x}$ of the domain $y$, we assume that the attribute values corresponding to the domain $y$ should be large, while those from other domains should be relatively small. For instance, the beard is usually longer on male faces compared to female faces. Therefore, we propose to use the sign operation to formulate this assumption.

First, we sample a vector $\mathbf{z}^p \in \mathbb{R}^{d \cdot N}$ from a prior distribution. Each attribute dimension of the vector is i.i.d. sampled from the prior Gaussian distribution $\mathcal{N}(0, 1)$. For a domain label $y$, we use the sign operation to compute the SAVs:

$$\mathbf{z}^s = O_y(\mathbf{z}^p) \sim \mathcal{N}(0, I), y \in \{1...N\}. \tag{2}$$

Specifically, the sign operation $O_y$ makes the attribute values $\{z_i\}_{i=y}^{d}$ of domain $y$ positive, and those for other domains $\{z_i\}_{i \neq y}^{d}$ negative by,

$$O_y(\mathbf{z}^p) = [-|z_1|, -|z_2|, \cdots, -|z_{d}], \cdots, +|z_{y}|, +|z_{y+1}|, \cdots, +|z_{d}], \cdots, -|z_{1}|, -|z_{2}|, \cdots, -|z_{y}|]. \tag{3}$$

We show an example of the proposed sign operation on an attribute vector of the target domain $\hat{y}$ in the gray block of Fig. 3(b).

Then, the SAV $\mathbf{z}^s$ can be applied to align the distribution of the unified attribute vector $\mathbf{z}$ extracted by the attribute encoder $E_a$ using the MMD [39] constraint:

$$L_{MMD} = \mathbb{E}_{p(\mathbf{z}^s), q(z)}[k(\mathbf{z}^s, \mathbf{z})] + \mathbb{E}_{q(\mathbf{z}^s), q(z)}[k(\mathbf{z}^s, \mathbf{z})] - 2\mathbb{E}_{p(\mathbf{z}^s), q(z)}[k(\mathbf{z}^s, \mathbf{z})], \tag{4}$$

where $k$ is the Gaussian kernel $k(\mathbf{z}^s, \mathbf{z}) = e^{-\frac{||\mathbf{z}^s - \mathbf{z}||^2}{2\sigma^2}}$.

As presented in Fig. 3(b), the content representation extracted by the source image along with the attribute representation, which is either sampled by the signed attribute space or extracted by the reference target image, are fed into the generator to synthesize the translated images of target domain $\hat{y}$. We use a multi-task discriminator $D$ [29, 26, 9] with multi-branch outputs for all domains during the training stage to ensure that the translated images belong to the corresponding domain. For the translation target domain $\hat{y}$, the corresponding
branch $D_y$ is learned by the domain adversarial loss,

$$L_{\text{adv}} = E[\log D_y(x_y) + \log(1 - D_y(G(c, z))],$$  \hspace{1cm} (5)

where $x_y$ is the real image of the target domain, $c = E_c(x)$ is the content representation of the source image, and $z$ is the attribute vector which can be derived from $E_a(x_y)$ (reference-guided) or $O_a(z^p)$ (latent-guided).

We embed only the sign information in the attribute vectors sampled from a prior distribution in this work. For the unified attribute vectors embedded by the attribute encoder $E_a$, the sign information is learned via the domain adversarial loss $L_{\text{adv}}$ and the MMD constraint $L_{\text{MMD}}$. Furthermore, we adopt the style reconstruction loss [17] to ensure the style consistency of the translated images with the reference image:

$$L_{\text{style}} = E[|\text{Gram}(\phi(G(E_c(x), E_a(x_i)))) - \text{Gram}(\phi(x_i))|_1],$$  \hspace{1cm} (6)

where Gram is gram matrix, and $\phi$ is embedding feature space. In particular, we use ReLU3_1 VGG features.

With such a simple sign operation on SAVs, we construct a unified attribute space incorporating the domain information, thus allowing continuous translation across domains.

3.2 Improving Quality by Sign-Symmetrical Attribute Vectors

As illustrated in Fig. 3(c), we can achieve continuous translation across domains by interpolating the proposed SAVs of two different domains. However, improving the quality of intermediate results remains a challenge. Due to the lack of real interpolation samples, we cannot directly apply the domain adversarial loss in Eq. 5 to ensure the realism of the interpolated images. Therefore, we propose two sign-symmetrical attribute vectors to create a continuous translation trajectory for conducting the domain adversarial training.

Fig. 4: **Illustration of interpolation between two sign-symmetrical points.** Using a 2D-plane, we show that the signed attribute vector contains attributes of two domains, and each domain has one attribute dimension. (a) The sign-symmetrical attribute vector is obtained by reversing the sign of the attributes in the source and target domains. (b) When $\beta < 0.5$, the interpolated point lies in the source domain. (c) When $\beta = 0.5$, the interpolated point is an intermediate state. (d) When $\beta > 0.5$, the interpolated point belongs to the target domain.

$$z_{\text{sym}} = O^r(z^p) = [\cdot \cdot \cdot , +|z^p_1|, +|z^p_2|, \cdot \cdot \cdot , +|z^p_d|, \cdot \cdot \cdot , -|z^p_1|, -|z^p_2|, \cdot \cdot \cdot , -|z^p_d|, \cdot \cdot \cdot ].$$  \hspace{1cm} (7)

For the translated image $\hat{x}$ using the sign-symmetrical attribute vector $z_{\text{sym}}$, we first apply the domain adversarial loss to assure its domain membership belongs to the target domain $\hat{y}$ and name it as the reverse sign domain adversarial loss:

$$L_{\text{adv}} = E[\log D_y(x_y) + \log(1 - D_y(G(x))].$$  \hspace{1cm} (8)

where $\hat{x} = G(c, z_{\text{sym}})$, and $c = E_c(x)$.

Since traversing these two vectors $z$ and $z_{\text{sym}}$ forms a continuous translation trajectory across domains, we leverage this path during the training stage. Specifically, we sample the interpolation coefficient from a uniform distribution, i.e., $\beta \in (0, 1)$, and conduct linear interpolation on the two sign-symmetrical vectors $z^*$ and $z_{\text{sym}}$ in domain $y$ and $\hat{y}$, respectively. The interpolated attribute vector can be formulated as $z^i = (1 - \beta) \cdot z^* + \beta \cdot z_{\text{sym}}$. Combining a content representation $c$, we generate the interpolated translated result $x^i = G(c, z^i)$. Based on the sign information of the attribute vector, there exist three cases of $x^i$: 

(a) Sign-symmetrical attribute vectors
(b) $\beta < 0.5$ : Source domain
(c) $\beta = 0.5$ : Intermediate state
(d) $\beta > 0.5$ : Target domain

![Diagram showing interpolation between two sign-symmetrical points](image-url)
where $\beta < 0.5$, the interpolated attribute vector still locates in the domain $y$, as illustrated in Fig. 4(b). Therefore, the generated interpolated image belongs to the domain $y$.

- 2) When $\beta > 0.5$, the interpolated attribute vector lies in the target domain, as demonstrated in Fig. 4(d). At this time, the interpolated image belongs to the domain $\hat{y}$.

- 3) When $\beta = 0.5$, this is the intermediate state, as shown in Fig. 4(c). In this case, we regard the interpolated result indistinguishable from neither the domain $y$ nor the domain $\hat{y}$. Compared to $\beta \in (0, 0.5)$ and $\beta \in (0.5, 1)$, this point is much less sampled. As a result, we can apply the domain adversarial loss to the images generated along this trajectory to ensure the quality of the continuous translation results, which is defined as the interpolated domain adversarial loss and summarized as follows

$$L_{\text{adv}}^{\text{interp}} = \begin{cases} \mathbb{E}[\log D_y(x) + \log(1 - D_y(x')])] & \beta < 0.5, \\ \frac{1}{2}\mathbb{E}[\log D_y(x) + \log(1 - D_y(x'))] + \\ \frac{1}{2}\mathbb{E}[\log D_y(x) + \log(1 - D_y(x'))] & \beta = 0.5, \\ \frac{1}{2}\mathbb{E}[\log D_y(x) + \log(1 - D_y(x'))] & \beta > 0.5. \end{cases}$$

(9)

Although we only exploit this specific interpolation path during training, the model is generalized to ensure interpolation results between any two attribute vectors at the inference stage.

3.3 Other Loss Objectives

In addition to the above loss functions, we also apply several loss objectives commonly used in the I2I translation approaches to train the proposed model.

Content adversarial loss. To further disentangle the content and attribute representations, we adopt the content discriminator $D^c$ [22, 23, 37] to distinguish the content representations belong to different domains. On the other hand, the content encoder $E^c$ aims to generate the content representations fool the content discriminator $D^c$. Then, the content adversarial loss is defined by

$$L_{\text{adv}}^{\text{content}} = \mathbb{E}_x [\frac{1}{2} \log D^c(E^c(x)) + \frac{1}{2} \log(1 - D^c(E^c(x)))].$$

(10)

Cycle-consistency loss. To preserve the consistency of domain-invariant characteristics of generated images, we impose the cycle-consistency loss [9, 40],

$$L_{\text{cc}} = \mathbb{E} ||x - G(E_c(x), E_a(x))||_{1},$$

(11)

where $\hat{x} = G(E_c(x), E_a(x_{\hat{y}}))$ or $\hat{x} = G(E_c(x), \mathcal{O}_y(z^p))$.

### Table 2: Details of network architectures.

(a) Content encoder $E_a$ architecture.

| Layer Type | Norm/Activation | Input Size | Output Size |
|------------|-----------------|------------|-------------|
| Conv(3,1,1) | -               | 256 x 256 x 3 | 256 x 256 x 64 |
| DownResBlock | IN LReLU | 256 x 256 x 64 128 x 128 x 128 | 256 x 256 x 64 128 x 128 x 128 |
| ResBlock | IN LReLU | 64 x 64 x 256 64 x 64 x 256 | 64 x 64 x 256 64 x 64 x 256 |
| ResBlock | IN LReLU | 64 x 64 x 256 32 x 32 x 512 | 64 x 64 x 256 32 x 32 x 512 |
| ResBlock | IN LReLU | 32 x 32 x 512 32 x 32 x 512 | 32 x 32 x 512 32 x 32 x 512 |

(b) Generator $G$ architecture. Style translation tasks directly concatenate the attribute vector, while shape-variation translation tasks adopt AdaIN to inject the attribute vector.

| Layer Type | Norm/Activation | Input Size | Output Size |
|------------|-----------------|------------|-------------|
| Conv(3,1,1) | -               | 256 x 256 x 3 | 256 x 256 x 64 |
| Conv(3,1,1) | IN LReLU | 256 x 256 x 64 128 x 128 x 128 | 256 x 256 x 64 128 x 128 x 128 |
| DownResBlock | IN LReLU | 128 x 128 x 128 64 x 64 x 256 | 128 x 128 x 128 64 x 64 x 256 |
| DownResBlock | IN LReLU | 64 x 64 x 256 32 x 32 x 512 | 64 x 64 x 256 32 x 32 x 512 |
| DownResBlock | IN LReLU | 32 x 32 x 512 32 x 32 x 512 | 32 x 32 x 512 32 x 32 x 512 |
| Conv(1,1,0) | IN LReLU | 256 x 256 x 64 256 x 256 x 3 | 256 x 256 x 3 |

(e) Attribute encoder $E_a$ and discriminator $D$ architecture. The $d$ is set to 8 and 1 for $E_a$ and $D$ respectively.

| Layer Type | Norm/Activation | Input Size | Output Size |
|------------|-----------------|------------|-------------|
| Conv(3,1,1) | -               | 256 x 256 x 3 | 256 x 256 x 64 |
| DownResBlock | IN LReLU | 256 x 256 x 64 128 x 128 x 128 | 256 x 256 x 64 128 x 128 x 128 |
| DownResBlock | IN LReLU | 128 x 128 x 128 64 x 64 x 256 | 128 x 128 x 128 64 x 64 x 256 |
| DownResBlock | IN LReLU | 64 x 64 x 256 32 x 32 x 512 | 64 x 64 x 256 32 x 32 x 512 |
| DownResBlock | IN LReLU | 32 x 32 x 512 32 x 32 x 512 | 32 x 32 x 512 32 x 32 x 512 |
| Conv(4,1,0) | IN LReLU | 8 x 8 x 512 4 x 4 x 512 | 8 x 8 x 512 4 x 4 x 512 |
| Reshape | - | 1 x 1 x 512 | 512 |
| PC | - | 512 | $d \cdot N$ |

(d) Content discriminator $D_c$ architecture. The $c$ is set to 256 and 512 for style translation, and shape-variation translation tasks, respectively, and shape-variation translation tasks do not contain the first DownResBlock.

| Layer Type | Norm/Activation | Input Size | Output Size |
|------------|-----------------|------------|-------------|
| DownResBlock | -               | 64 x 64 x 32 x 32 x c | 64 x 64 x 32 x 32 x c |
| DownResBlock | -               | 32 x 32 x 16 x 16 x c | 32 x 32 x 16 x 16 x c |
| DownResBlock | -               | 16 x 16 x 8 x 8 x c | 16 x 16 x 8 x 8 x c |
| DownResBlock | -               | 8 x 8 x 4 x 4 x c | 8 x 8 x 4 x 4 x c |
| Conv(4,1,0) | -               | 4 x 4 x 1 x 1 c | 4 x 4 x 1 x 1 c |
| Conv(1,1,0) | -               | 1 x 1 x 1 x 1 c | 1 x 1 x 1 x 1 c |

(e) Fusing network $F$ architecture.

| Layer Type | Norm/Activation | Input Size | Output Size |
|------------|-----------------|------------|-------------|
| FC | -               | ReLU | 8 $\cdot$ N | 512 |
| PC | -               | ReLU | 512 | 512 |
| PC | -               | ReLU | 512 | 64 |
Self-reconstruction loss. We reconstruct the original image \([22,23]\) using the encoded content representation and attribute representation as
\[
\mathcal{L}_{\text{recon}}^1 = \mathbb{E}[\|G(E_c(x), E_a(x)) - x\|_1].
\] (12)

Latent regression loss \([41,22,23,37,9]\) is adopted to further encourage the invertible mapping between the generated image and the signed attribute space. We reconstruct the signed attribute vector \(z^s\) as
\[
\mathcal{L}_{\text{latent}}^1 = \mathbb{E}[\|E_a(G(E_c(x), z^s)) - z^s\|_1].
\] (13)

Mode seeking loss. To alleviate the mode collapse problem and improve the diversity of generated images. We introduce another SAV \(z^{s_2}\) to calculate the mode seeking loss \([28]\) as
\[
\max \mathcal{L}_{\text{ms}} = \mathbb{E}[\|G(E_c(x), z^{s_2}) - G(E_c(x), z^{s_1})\|_1].
\] (14)

The objective function of our framework is
\[
\begin{align*}
\mathcal{L}_{D,D_c} & = \lambda_{\text{content}} \mathcal{L}_{\text{content}} + \lambda_{\text{domain}} \mathcal{L}_{\text{domain}} + \lambda_{\text{adv}} \mathcal{L}_{\text{adv}} \\
& \quad + \lambda_{\text{adv}} \mathcal{L}_{\text{rvs}} + \lambda_{\text{adv}} \mathcal{L}_{\text{interp}} \\
\mathcal{L}_{G,E_c,E_a} & = -\lambda_{\text{adv}} \mathcal{L}_{\text{content}} + \lambda_{\text{domain}} \mathcal{L}_{\text{domain}} \\
& \quad + \lambda_{\text{adv}} \mathcal{L}_{\text{rvs}} + \lambda_{\text{adv}} \mathcal{L}_{\text{interp}} \\
& \quad + \lambda_{\text{MMD}} \mathcal{L}_{\text{MMD}} + \lambda_{\text{style}} \mathcal{L}_{\text{style}} + \lambda_{\text{cc}} \mathcal{L}_{\text{cc}} \\
& \quad + \lambda_{\text{recon}} \mathcal{L}_{\text{recon}} + \lambda_{\text{latent}} \mathcal{L}_{\text{latent}} + \lambda_{\text{ms}} \mathcal{L}_{\text{ms}},
\end{align*}
\] (15)

where the term \(\lambda\) controls the importance of each loss function.

4 Implementation Details

The proposed model is implemented in Pytorch \([31]\), and the source code will be made publicly available at https://github.com/HelenMao/SAVI2I. More implementation details and experimental results can be found on this website.

Datasets We evaluate the proposed method on four representative datasets, including the low-level style translation and shape-variation translation tasks.

For style translation, the Yosemite \([40]\) dataset includes the summer and the winter two domains. The Photo2Artwork dataset \([40]\) contains the photo, Monet, Van Gogh, and Ukiyo-e domains. For shape-variation translation, the CelebA-HQ \([18]\) dataset in which we split the male and female domains for translation. The AFHQ \([9]\) dataset consists of animal faces with the cat, dog, and wildlife domains.

Network Architecture. The proposed model consists of a content encoder \(E_c\), an attribute encoder \(E_a\), a generator \(G\), a discriminator \(D\), and a content discriminator \(D_c\). We set the size of an attribute vector to
Fig. 6: **One input and multiple targets in the AFHQ dataset.** Each row continuously translates the source image to the target domain using the attribute vector extracted by (a) a reference image or (b) an SAV of the target domain. Every two rows from the top present the continuous translation in the following order: cat→dog and cat→wildlife.

$z \in \mathbb{R}^{8 \cdot N}$, where $N$ represents the number of visual domains in the dataset. We feed the attribute vectors into a fusing network $F$ with three-layer MLP before feeding them into the generator. Since the style translation tasks require more content preservation than shape-variation translation tasks, we adopt different network architecture choices for these two tasks. Table 2 shows the network configuration details where Conv$(k, s, p)$ and DeConv$(k, s, p)$ denote the convolutional layer and transposed convolutional layer with $k$ as kernel size, $s$ as stride, and $p$ as padding; DownResBlock and UpResBlock adopt the average pooling for down-sampling and the nearest-neighbor interpolation for up-sampling respectively; AdaIN is the adaptive instance normalization [14]; LN is the layer normalization [4] and IN is the instance normalization [32]; and LReLU indicates leaky ReLU [27] with a negative slope of 0.2.

**Training Process.** The resolution of each image is 256 × 256 pixels for all the experiments. We adopt the following hyper-parameters for the training in all the experiments: $\lambda_{\text{content}} = 1$, $\lambda_{\text{domain}} = 1$, $\lambda_{\text{adv}} = 1$, $\lambda_{\text{interp}} = 1$, $\lambda_{\text{style}} = 1$, $\lambda_{\text{cc}} = 10$, $\lambda_{\text{recon}} = 10$, $\lambda_{\text{latent}} = 10$, $\lambda_{\text{ms}} = 1$. For AHFQ, we employ $\lambda_{\text{MMD}} = 10$, and $\lambda_{\text{MMD}} = 1$ for other datasets. We use the batch size of 1 as well as the Adam [20] optimizer with a learning rate of $10^{-4}$ and exponential decay rates $(\beta_1, \beta_2) = (0, 0.99)$. We adopt the non-saturating adversarial loss [11] with $R_1$ regularization [29] with $\gamma = 10$ for style translation tasks, and $\gamma = 1$ for shape-variation translation tasks. All of the models are trained on two NVIDIA Tesla-P100 GPUs with 16GB memory.

5 Experiments

5.1 Continuous and Diverse Image-to-Image Translation

We present diverse continuous translation paths from the source domain to the target one. An input image $I_s$ in the source domain can be continuously translated to multiple images $I_{t_1}, I_{t_2}, \ldots, I_{t_N}$ in the target domain.
We name this type as one input and multiple targets. Furthermore, given a source image $I_s$ and a target image $I_t$, we can obtain multiple continuous translation paths by passing different intermediate attribute vectors, called as one input, one target, and multiple intermediate points.

**One input and multiple targets.** We first embed an image of the source domain to obtain the content representation and the source attribute vector. Then, we compute a target attribute vector by either extracting a reference image sampled from the target domain (reference-guided) or randomly generate an SAV of the target domain (latent-guided), as illustrated in Fig. 3(c). We apply linear interpolation of the source and target attribute vectors to generate continuous translation results. Fig. 5 and Fig. 6 show continuous and diverse translation results with different reference images and SAVs conditioned on the same source image in style translation and shape-variation translation tasks, respectively.

**One input, one target, and multiple intermediate points.** Given a source attribute vector and a target attribute vector, we can also apply interpolation with multiple intermediate points to generate multiple translation paths, as presented in Fig. 7.

5.2 Comparisons with the State-of-the-arts

We present qualitative and quantitative evaluations with the state-of-the-art approaches on the male→female translations.

**State-of-the-art methods.** We evaluate the proposed method against the state-of-the-art I2I translation approaches, including StarGAN-v2 [9], FUNIT [26], and DLOW [10], and facial attribute editing schemes, including HomoInterpGAN [7], RelGAN [35], and Fader networks [21]. For the I2I translation approaches, we adopt the pre-trained StarGAN-v2 model provided by the authors. Although the FUNIT model is designed to address the few-shot setting of I2I translation, we use the same training protocols to train the FUNIT model for fair comparisons. We train DLOW using the codes provided by the authors. For facial attribute editing schemes, we only use domain labels to train the models using the codes provided by the authors instead of fine-grained attribute annotations.

**Evaluation metrics.** For quantitative evaluation, we assess two critical features of the interpolated images: realism and smoothness. Given a set of interpolated images $\{x_0, x_1, \ldots, x_{t-1}, x_t\}$, we measure the visual quality of the interpolation results by calculating the Fréchet Inception Distance (FID) [12] between the interpolated images and the images collected from different domains, denoted as I-FID. Lower I-FID means better realism of
Fig. 8: Qualitative comparisons of reference-guided continuous translation results on the CelebA-HQ dataset.

Table 3: Quantitative comparisons on the CelebA-HQ (male→female) dataset. Case 1: reference-guided continuous translation. Case 2: latent-guided continuous translation. Bold and Underline indicate the lowest and highest values, respectively.

|                  | Male→Female | Female→Male |
|------------------|-------------|-------------|
|                  | I-FID | δ-N-LPIPS | I-FID | δ-N-LPIPS |
| StarGAN-v2       | 51.61 | 0.1308 | 47.39 | 0.1299 |
| HomoInterpGAN    | 46.65 | **0.0025** | 39.19 | **0.0022** |
| FUNIT            | 40.75 | 0.0180 | 33.58 | 0.0170 |
| Ours             | **15.13** | 0.0236 | **14.53** | 0.0244 |
| StarGAN-v2       | 50.32 | 0.1233 | 41.22 | 0.1193 |
| DLOW             | 33.96 | 0.0041 | 20.82 | 0.0167 |
| RelGAN           | 31.05 | 0.0457 | 19.63 | 0.0344 |
| Fader networks   | **62.92** | **0.0038** | **59.33** | **0.0034** |
| Ours             | **15.59** | 0.0425 | **14.18** | 0.0358 |

interpolated images. For smoothness, we compute the standard deviation of the normalized LPIPS [38] distance between $x_{i-1}$ and $x_i$ as

$$\sigma\left(\frac{\text{LPIPS}(x_{i-1}, x_i)}{\text{LPIPS}(x_0, x_t)}\right),$$

where $\sigma(\cdot)$ implies the standard deviation. We denote it as $\delta$-N-LPIPS, and a smaller standard deviation indicates a smoother interpolation. In particular, we adopt $t = 20$ and use images in the source domain’s test set to generate interpolated images. The interpolated images and the training images of both the source and target domains are used to calculate the I-FID score. For the $\delta$-N-LPIPS metric, we first compute all the LPIPS scores between any two consecutive interpolated images. We then compute the standard deviation of the LPIPS scores in the same test case (i.e., the same source image and target image). We report the average values of all test cases’ standard deviation as the final $\delta$-N-LPIPS score.

Reference-guided continuous translation. We evaluate the proposed method against the StarGAN-v2 [9], FUNIT [26], and HomoInterpGAN [7] models. For the StarGAN-v2 model, we feed the source and target image into the style encoder [9] and obtain style vectors from the source and target domain branch. Then, we apply linear interpolation on these two style vectors. Similarly, we feed the source and target image into the class encoder of FUNIT [26] and interpolate the
class codes to generate interpolated results. HomoInterpGAN [7] embeds the source and target image into the unified latent feature space by the encoder, and we adjust the value of the control vector $v \in [0, 1]^{c \times 1}$ on the domain branch to control the interpolation results.

**Latent-guided continuous translation.** We evaluate the proposed method against the StarGAN-v2 [9], DLOW [10], RelGAN [35], and Fader networks [21] models. The DLOW, RelGAN, and Fader networks interpolate discrete domain labels to generate intermediate results. For StarGAN-v2, we randomly sample a latent vector from the Gaussian distribution and feed it into the mapping network to acquire the target domain’s style vector from the corresponding domain branch [9], and apply interpolation between source and target style vectors.

**Analysis.** Fig. 8, Fig. 9 and Table 3 show qualitative and quantitative comparisons of these methods.

Fig. 9: Qualitative comparisons of latent-guided continuous translation results on the CelebA-HQ dataset.

**Table 4: Ablation study on the male→female translation.** Bold indicates the best performance.

| SAV | $\mathcal{L}_{\text{adv}}$ | $\mathcal{L}_{\text{interp}}$ | $\mathcal{L}_{\text{MMD}}$ | $\mathcal{L}_{\text{style}}$ | F | FID ↓ | $\delta$-N-LPIPS ↓ |
|-----|----------------|----------------|----------------|----------------|---|-------|----------------|
| ×   | ×      | ×      | ✓       | ✓       | ✓       | 22.31 | 0.0190 |
| ✓   | ×      | ×      | ✓       | ✓       | ✓       | 18.21 | 0.0327 |
| ✓   | ✓      | ×      | ✓       | ✓       | ✓       | 17.52 | 0.0335 |
| ✓   | ✓      | ✓      | ✓       | ✓       | ✓       | 16.25 | 0.0191 |
| ✓   | ✓      | ✓      | ✓       | ✓       | ✓       | 22.30 | 0.0224 |
| ✓   | ✓      | ✓      | ✓       | ✓       | ✓       | 19.18 | **0.0165** |
| ✓   | ✓      | ✓      | ✓       | ✓       | ✓       | **15.13** | 0.0236 |

For reference-guided continuous translation, StarGAN-v2 does not generate intermediate results well due to the separate attribute spaces for different domains. Thus, it has the highest $\delta$-N-LPIPS value, as shown in Table 3. In contrast, our method can generate smooth intermediate results across domains with the proposed
signed attribute vector. The class code of the FUNIT model cannot capture the style of images well when trained with only two classes. Although HomoInterpGAN has the lowest value of δ-N-LPIPS, the translated images contain limited variations, as shown in Fig. 8. Furthermore, the translated results do not capture the style of the reference target image. For latent-guided continuous translation, we note the translated images by DLOW, RelGAN, and Fader networks contain only local variations (e.g., makeup and beard). They do not exhibit the hairstyles in Fig. 9. Overall, our approach synthesizes images with desired attributes such as hairstyle, makeup, and skin-color, leading to the lowest value of I-FID among all the evaluated methods. Our method achieves a comparable δ-N-LPIPS score against RelGAN but with interpolated images of more considerable variations, demonstrating the proposed model can accomplish both the high-quality interpolation and smooth transition.

**User study.** We conduct a user study from reference-guided and latent-guided these two perspectives to evaluate the user preference among the proposed method and the state-of-the-art approaches. We ask participants to select their preferred continuous translation results through pairwise comparisons. We randomly select 20 images from the test set of the CelebA-HQ dataset and collect the results from 21 participants. Fig. 10 shows that the majority of participants prefer our method (from 74.3% to 97.1% in the reference-guided setting and 81.9% to 93.8% in the latent-guided setting) when compared to all the other evaluated approaches.

### 5.3 Ablation Study

We present the ablation study on the male→female translation in Fig. 11 and Table 4 to analyze the three proposed components: SAV, reverse sign domain adversarial loss $L_{adv}^{rv}$, and interpolated domain adversarial loss $L_{adv}^{interp}$. In addition, we also demonstrate the efficiency of the MMD constraint $L_{mmd}$, the style reconstruction loss $L_{style}$, and the fusing network $F$.

Both quantitative and qualitative results demonstrate that the proposed SAV plays an essential role in the continuous translation across domains. Without the sign operation, all interpolation results belong to the female domain, as illustrated in Fig. 11. Since the dataset contains more female images than male images in the training set (17K VS 9K), the model tends to learn more female domain attributes without the sign information embedding. The $L_{adv}^{rv}$ ensures the domain-membership of the translated images using the sign-symmetrical attribute vector. As shown in Fig. 11, without $L_{adv}$, the translated target image cannot preserve the pose of the source image well. In particular, the pose of interpolated faces images varies from right to left. Training without $L_{interp}^{adv}$ leads to the highest δ-N-LPIPS score. Furthermore, Fig. 11 shows that it does not generate the intermediate results between two domains well. Therefore, applying $L_{adv}^{interp}$ of interpolated results on the trajectory between sign-symmetrical attribute vectors is essential to improve the quality and smoothness. We observe that training without $L_{style}$ cannot capture the style of the reference image for the translated image, as presented in Fig. 11. By aligning the distribution under the constraint of $L_{mmd}$, the attribute vector extracted by $E_a$ can better embed the sign information, further improving the quality of interpolated results. Table 4 shows that feeding the attribute vector into a fusing network before the generator enhances the quality of interpolated results.

### 6 Conclusions

In this paper, we present a signed attribute vector to enable continuous and diverse image-to-image translation across domains. To enhance the continuous translation quality, we propose to use the sign-symmetrical attribute vectors to form a translation trajectory between different domains. Then, we leverage the domain information of intermediate results for adversarial training.
Fig. 11: Ablation study on the male→female translation.

We evaluate our methods on a wide range of image-to-image translation tasks. Both qualitative and quantitative results demonstrate that the proposed method achieves high-quality and diverse continuous translation across domains.

References

1. Abdal, R., Qin, Y., Wonka, P.: Image2StyleGAN: How to embed images into the StyleGAN latent space? In: ICCV (2019) 4
2. Abdal, R., Qin, Y., Wonka, P.: Image2StyleGAN++: How to edit the embedded images? In: CVPR (2020) 4
3. Anoosheh, A., Agustsson, E., Timofte, R., Gool, L.V.: ComboGAN: Unrestrained scalability for image domain translation. In: CVPR Workshop (2018) 3
4. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016) 9
5. Brock, A., Donahue, J., Simonyan, K.: Large scale gan training for high fidelity natural image synthesis. In: ICLR (2018) 4
6. Chen, Y.C., Lin, Y.Y., Yang, M.H., Huang, J.B.: Crdoco: Pixel-level domain transfer with cross-domain consistency. In: CVPR (2019) 1
7. Chen, Y.C., Xu, X., Tian, Z., Jia, J.: Homomorphic latent space interpolation for unpaired image-to-image translation. In: CVPR (2019) 4, 10, 11, 12
8. Choi, Y., Choi, M., Kim, M., Ha, J.W., Kim, S., Choo, J.: StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In: CVPR (2018) 1, 2, 3
9. Choi, Y., Uh, Y., Yoo, J., Ha, J.W.: StarGAN v2: Diverse image synthesis for multiple domains. In: CVPR (2020) 1, 3, 5, 7, 8, 10, 11, 12
10. Gong, R., Li, W., Chen, Y., Gool, L.V.: DLOW: Domain flow for adaptation and generalization. In: CVPR (2019) 1, 2, 3, 10, 12
11. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NIPS (2014) 9
12. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: GANs trained by a two time-scale update rule converge to a local nash equilibrium. In: NIPS (2017) 10
13. Hoffman, J., Tzeng, E., Park, T., Zhu, J.Y., Isola, P., Saenko, K., Efros, A., Darrell, T.: Cycada: Cycle-consistent adversarial domain adaptation. In: ICM (2018) 1
14. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: ICCV (2017) 9
15. Huang, X., Liu, M.Y., Belongie, S., Kautz, J.: Multimodal unsupervised image-to-image translation. In: ECCV (2018) 1, 2, 3, 4
16. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR (2017) 1, 3
17. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: ECCV (2016) 6
18. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of GANs for improved quality, stability, and variation. ICLR (2018) 8
19. Kim, T., Cha, M., Kim, H., Lee, J.K., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. In: ICML (2017) 3
20. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015) 9
21. Lample, G., Zeghidour, N., Usunier, N., Bordes, A., Denoyer, L., Ranzato, M.: Fader networks: Manipulating images by sliding attributes. In: NIPS (2017) 4, 10, 12
22. Lee, H.Y., Tseng, H.Y., Huang, J.B., Singh, M.K., Yang, M.H.: Diverse image-to-image translation via disentangled representations. In: ECCV (2018) 1, 3, 4, 7, 8
23. Lee, H.Y., Tseng, H.Y., Mao, Q., Huang, J.B., Lu, Y.D., Singh, M., Yang, M.H.: DRIT++: Diverse image-to-image translation via disentangled representations. IJCV pp. 1–16 (2020) 1, 2, 3, 4, 7, 8
24. Liao, J., Lima, R.S., Nehab, D., Hoppe, H., Sander, P.V., Yu, J.: Automating image morphing using structural similarity on a halfway domain. ACM Transactions on Graphics pp. 1–12 (2014) 4
25. Liu, M.Y., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. In: NIPS (2017) 3
26. Liu, M.Y., Huang, X., Mallya, A., Karras, T., Aila, T., Lehtinen, J., Kautz, J.: Few-shot unsupervised image-to-image translation. In: ICCV (2019) 1, 3, 5, 10, 11
27. Maas, A.L., Hannun, A.Y., Ng, A.Y.: Rectifier nonlinearities improve neural network acoustic models. In: ICML (2013) 9
28. Mao, Q., Lee, H.Y., Tseng, H.Y., Ma, S., Yang, M.H.: Mode seeking generative adversarial networks for diverse image synthesis. In: CVPR (2019) 8
29. Mescheder, L., Geiger, A., Nowozin, S.: Which training methods for gans do actually converge? In: ICML (2018) 5, 9
30. Park, T., Liu, M.Y., Wang, T.C., Zhu, J.Y.: Semantic image synthesis with spatially-adaptive normalization. In: CVPR (2019) 1
31. Paszke, A., Gross, S., Chintalal, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch. In: NIPSW (2017) 8
32. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Instance normalization: The missing ingredient for fast stylization. arXiv preprint arXiv:1607.08022 (2016) 9
33. Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional GANs. In: CVPR (2018) 1
34. Wolberg, G.: Image morphing; a survey. The Visual Computer pp. 360–372 (1998) 4
35. Wu, P.W., Lin, Y.J., Chang, C.H., Chang, E.Y., Liao, S.W.: RelGAN: Multi-domain image-to-image translation via relative attributes. In: ICCV (2019) 2, 4, 10, 12
36. Wu, W., Cao, K., Li, C., Qian, C., Loy, C.C.: TransGaGa: Geometry-aware unsupervised image-to-image translation. In: CVPR (2019) 1
37. Yu, X., Chen, Y., Liu, S., Li, T., Li, G.: Multi-mapping image-to-image translation via learning disentanglement. In: NeurIPS (2019) 1, 3, 7, 8
38. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR (2018) 11
39. Zhao, S., Song, J., Ermon, S.: InfoVAE: Information maximizing variational autoencoders. arXiv preprint arXiv:1706.02202 (2017) 5
40. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: ICCV (2017) 1, 3, 7, 8
41. Zhu, J.Y., Zhang, R., Pathak, D., Darrell, T., Efros, A.A., Wang, O., Shechtman, E.: Toward multimodal image-to-image translation. In: NIPS (2017) 1, 3, 8