STGAN: A Unified Selective Transfer Network for Arbitrary Image Attribute Editing

Ming Liu¹, Yukang Ding², Min Xia¹, Xiao Liu², Errui Ding², Wangmeng Zuo²,³ Shilei Wen²
¹Harbin Institute of Technology, ²Department of Computer Vision Technology (VIS), Baidu Inc., ³Peng Cheng Laboratory, Shenzhen
cmliu@outlook.com, csxm@baidu.com, wmzuo@hit.edu.cn,
{dingyukang, liuxiaol2, dingerrui, wenshilei}@baidu.com

Abstract

Arbitrary attribute editing generally can be tackled by incorporating encoder-decoder and generative adversarial networks. However, the bottleneck layer in encoder-decoder usually gives rise to blurry and low quality editing result. And adding skip connections improves image quality at the cost of weakened attribute manipulation ability. Moreover, existing methods exploit target attribute vector to guide the flexible translation to desired target domain. In this work, we suggest to address these issues from selective transfer perspective. Considering that specific editing task is certainly only related to the changed attributes instead of all target attributes, our model selectively takes the difference between target and source attribute vectors as input. Furthermore, selective transfer units are incorporated with encoder-decoder to adaptively select and modify encoder feature for enhanced attribute editing. Experiments show that our method (i.e., STGAN) simultaneously improves attribute manipulation accuracy as well as perception quality, and performs favorably against state-of-the-arts in arbitrary facial attribute editing and season translation.

Figure 1: High resolution (384 × 384) results of STGAN for facial attribute editing, and more results are given in the suppl.

1. Introduction

Image attribute editing, aiming at manipulating an image to possess desired attributes, is an interesting but challenging problem with many real-world vision applications. On one hand, it is impracticable to collect paired images with and without desirable attributes (e.g., female and male face images of the same person). Thus, unsupervised generative learning models, e.g., generative adversarial networks (GANs) [9], have attracted upsurging attention in attribute editing. On the other hand, arbitrary attribute editing actually is a multi-domain image-to-image translation task. Learning single translation model for each specific attribute editing task may achieve limited success [20, 29, 34]. But it is ineffective in exploiting the entire training data, and the learned models grows exponentially along with the number of attributes. To handle this issue, several arbitrary attribute editing approaches [7, 11, 26] have been developed, which usually (i) use encoder-decoder architecture, and (ii) take both source image and target attribute vector as input.

Albeit their extensive deployment, encoder-decoder networks remain insufficient for high quality attribute editing. Attribute can be of either local, global, or abstract characteristic of the image. In order to properly manipulate image attribute, spatial pooling or downsampling generally are required to obtain high-level abstraction of image content.

*Work done during an internship at Baidu.
and attributes. For example, auto-encoder architecture is adopted in [11, 17, 26], and shallow encoder-decoder with residual blocks is used in [7, 34]. However, the introduction of bottleneck layer, i.e., the innermost feature map with minimal spatial size, gives rise to blurry and low quality editing result. As a remedy, some researchers suggest to add one [11] or multiple [13] skip connections between encoder and decoder layers. Unfortunately, as further shown in Sec. 3.1, the deployment of skip connections improves image quality of editing result but is harmful to attribute manipulation ability of learned model. Another possible solution is to employ spatial attention network to allow attribute-specific region editing [32], which, however, is effective only for local attributes and not designed for arbitrary attribute editing.

Moreover, most existing methods exploit both source image and target attribute vector for arbitrary attribute editing. In particular, the encoders in [11, 17] only take source image as input to produce latent code, and then the decoders utilize both latent code and target attribute vector to generate editing result. In contrast, StarGAN [7] directly takes source image and target attribute vector as input. However, for arbitrary attribute editing only the attributes to be changed are required, taking full target attribute vector as input may even have adverse effect on editing result. As shown in Fig. 2, although all attributes keep unchanged, unwanted changes and visual degradation can be observed in the results by AttGAN [11] and StarGAN [7], mainly ascribing to the limitation of encoder-decoder and the use of target attribute vector as input.

To address the above issues, this work investigates arbitrary attribute editing from selective transfer perspective and presents a STGAN model. In terms of selective, our STGAN is suggested to (i) only consider the attributes to be changed, and (ii) selectively concatenate encoder feature in editing attribute irrelevant regions with decoder feature. In terms of transfer, our STGAN is expected to adaptively modify encoder feature to match the requirement of varying editing task, thereby providing a unified model for handling both local and global attributes.

To this end, instead of full target attribute vector, our STGAN takes the difference between target and source attribute vectors as input to encoder-decoder. Subsequently, selective transfer units (STUs) are proposed to adaptively select and modify encoder feature, which is further concatenated with decoder feature for enhancing both image quality and attribute manipulation ability. In particular, STU is added to each pair of encoder and decoder layers, and takes both encoder feature, inner state, and difference attribute vector into consideration for exploiting cross-layer consistency and task specificity. From Figs. 1 and 2, our STGAN can generate high quality and photo-realistic results for arbitrary attribute editing, and obtain near-ideal reconstruction when the target and source attributes are the same. To sum up, the contribution of this work involves:

- Instead of all target attributes, difference attribute vector is taken as input to enhance the flexible translation of attributes and ease the training procedure.
- Selective transfer units are presented and incorporated with encoder-decoder for simultaneously improving attribute manipulation ability and image quality.
- Experimental results show that our STGAN performs favorably against state-of-the-arts in arbitrary facial attribute editing and season translation.

2. Related Work

Encoder-Decoder Architecture. In their pioneer work [12], Hinton and Zemel proposed an autoencoder network, which consists of an encoder to map the input into latent code and a decoder to recover from the latent code. Subsequently, denoising autoencoders [30] are presented to learn representation robust to partial corruption. Kingma and Welling [16] suggested a Variational Autoencoder (VAE), which validates the feasibility of encoder-decoder architecture to generate unseen images. Recent studies show that skip connections [14, 28] between encoder and decoder layers usually benefit the training stability and visual quality of generated images. However, as discussed in Sec. 3.1, skip connections actually improves image quality at the cost of weakened attribute manipulation ability, and should be carefully used in arbitrary attribute editing.

Generative Adversarial Networks. GAN [9, 27] is originally proposed to generate images from random noise, and generally consists of a generator and a discriminator which are trained in an adversarial manner and suffer from the mode collapse problem. Recently, enormous efforts have been devoted to improving the stability of learning. In [3, 10], Wasserstein-1 distance and gradient penalty are suggested to improve stability of the optimization process. In [18], the VAE decoder and GAN generator are collapsed into one model and optimized by both reconstruction and adversarial loss. Conditional GAN (cGAN) [14, 24] takes conditional variable as input to the generator and discriminator to generate image with desired properties. As a re-
AttGAN-ED

A further assesses the facial attribute generation ac-

dependence between latent code and attributes. Modu-

larGAN [50x81] Image-to-image translation

Are then developed to improve the visual quality of syn-

thesized images. As for unpaired image-to-image trans-

lation, additional constraints, e.g., cycle consistency [34] and

shared latent space [22], are suggested to alleviate the in-

herent ill-posedness of the task. Nonetheless, arbitrary at-

tribute editing actually is a multi-domain image-to-image

translation problem, and cannot be solved with scalability

by aforementioned methods. To address this issue, [2] and

and [13] decouple generators by learning domain-specific

encoders/decoders with shared latent space, but are still lim-

ited in scaling to change multiple attributes of an image.

Facial Attribute Editing. Facial attribute editing is an in-

teresting multi-domain image-to-image translation problem

and has received considerable recent attention. While sev-

eral methods have been proposed to learn single translation

model for each specific attribute editing task [5, 20, 29, 32],

they suffer from the limitation of image-to-image trans-

lation and cannot well scale to arbitrary attribute editing.

Therefore, researchers resort to learning a single model for

arbitrary attribute editing. IcGAN [26] adopts an encoder to

generate latent code of an image, and a cGAN to de-

code latent code conditioned on target attributes. However,

IcGAN first trains the cGAN model followed by the en-

coders, greatly restricting its reconstruction ability. Lam-

ple et al. [17] trained the FaderNet in an end-to-end man-

ner by imposing adversarial constraint to enforce the in-

dependence between latent code and attributes. ModularGAN [33] presents a feasible solution to connect specific

attribute editing to arbitrary attribute editing, but its computa-
tion time gradually increases along with the number of at-
tributes to be changed. StarGAN [7] and AttGAN [11] elab-

orately tackle arbitrary attribute editing by taking target at-
tribute vector as input to the transform model. In this work,

we analyze the limitation of StarGAN [7] and AttGAN [11],

and further develop a STGAN for simultaneously enhan-
ing the attribute manipulation ability and image quality.

3. Proposed Method

This section presents our proposed STGAN for arbitrary

attribute editing. To begin with, we use AttGAN as an ex-

ample to analyze the limitation of skip connections. Then,

we formulate STGAN by taking difference attribute vec-
tor as input and incorporating selective transfer units into

encoder-decoder structure. Finally, network architecture

(see Fig. 5) and model objective of STGAN are provided.

3.1. Limitation of Skip Connections in AttGAN


structure, where spatial pooling or downsampling are es-
tential to obtain high level abstract representation for at-

tribute manipulation. Unfortunately, downsampling irre-

versibly diminishes spatial resolution and fine details of fea-
ture map, which cannot be completely recovered by trans-

posed convolutions and the results are prone to blurring or

missing details. To enhance image quality of editing result,

AttGAN [11] applies one skip connection between encoder

decoder, but we will show that it is still limited.

To analyze the effect and limitation of skip connections,

we test four variants of AttGAN on the test set: (i) AttGAN

w/o skip connection (AttGAN-ED), (ii) AttGAN model re-

leased by He et al. [11] with one skip connection (AttGAN),

(iii) AttGAN with two skip connections (AttGAN-2s), and
(iv) AttGAN with all symmetric skip connections [28]

(AttGAN-UNet). Table 1 lists the PSNR/SSIM results of

reconstruction by keeping target attribute vector the same as

the source one, and Fig. 3 shows the reconstruction results

of an image. It can be seen that adding skip connections
does benefit the reconstruction of fine details, and better re-

sult can be obtained with the increase of skip connections.

By setting target attribute vector different from source one,

Fig. 4 further assesses the facial attribute generation ac-

curacy via a facial attribute classification model1. While

We train the model on CelebA [23] dataset which can achieve 94.5% mean accuracy on the 13 attributes we use.
adding one skip connection, i.e., AttGAN, only slightly decreases generation accuracy for most attributes, notable degradation can be observed by adding multiple skip connections. Thus, the deployment of skip connections improves reconstruction image quality at the cost of weakened attribute manipulation ability, mainly attributing to that skip connection directly concatenates encoder and decoder features. To circumvent this dilemma, we present our STGAN to employ selective transfer units to adaptively transform encoder features guided by attributes to be changed.

3.2. Taking Difference Attribute Vector as Input

Both StarGAN [7] and AttGAN [11] take target attribution vector \( \text{att}_t \) and source image \( x \) as input to the generator. Actually, the use of full target attribution vector is redundant and may be harmful to editing result. In Fig. 2, the target attribution vector \( \text{att}_t \) is exactly the same as on the source one \( \text{att}_s \), but StarGAN [7] and AttGAN [11] may manipulate some unchanged attributes by mistake. From Fig. 2, after editing the face image with blond hair becomes more blond. Moreover, they even incorrectly adjust hair length of a source image with the attribute female.

For arbitrary image attribute editing, instead of full target attribute vector, only the attributes to be changed should be considered to preserve more information of source image. So we define the difference attribute vector as the difference between target and source attribute vectors,

\[
\text{att}_{\text{diff}} = \text{att}_t - \text{att}_s.
\]

Taking \( \text{att}_{\text{diff}} \) as input can bring several distinctive merits. First, the attributes to be changed are only a small set of attribute vector, and the use of \( \text{att}_{\text{diff}} \) usually makes the model easier to train. Second, in comparison to \( \text{att}_t \), \( \text{att}_{\text{diff}} \) can provide more valuable information for guiding image attribute editing, including whether an attribute is required to edit or not, toward which direction an attribute should be changed. The information can then be utilized to design proper model to transform and concatenate encoder feature with decoder feature, and improve image reconstruction quality without sacrifice of attribute manipulation accuracy. Finally, in practice \( \text{att}_{\text{diff}} \) actually is more convenient to be provided by user. When taking \( \text{att}_t \) as input, the user is required to either manually supply all target attributes, or modify source attributes provided by some attribute prediction method.

3.3. Selective Transfer Units

Fig. 5 shows the overall architecture of our STGAN. Instead of directly concatenating encoder with decoder features via skip connection, we present selective transfer unit (STU) to selectively transform encoder feature, making it compatible and complementary to decoder feature. Naturally, the transform is required to be adaptive to the changed attributes, and be consistent among different encoder layers. Thus, we modify the structure of GRU [6, 8] to build STUs for passing information from inner layers to outer layers.

Without loss of generality, we use the \( l \)-th encoder layer as an example. Denote by \( f^l_{\text{enc}} \) the encoder feature of the \( l \)-th layer, and \( s^{l+1} \) the hidden state from the \( l+1 \)-th layer. For convenience, the difference attribute vector \( \text{att}_{\text{diff}} \) is stretched to have the same spatial size of \( s^{l+1} \). Different from sequence modeling, feature maps across layers are of different spatial size. So we first use transposed convolution to upsample hidden state \( s^{l+1} \),

\[
\hat{s}^{l+1} = W_t * T[s^{l+1}, \text{att}_{\text{diff}}],
\]

where \( [\cdot, \cdot] \) denotes the concatenation operation, and \( *T \) denotes transposed convolution. Then, STU adopts the mathematical model of GRU to update the hidden state \( s^l \) and transformed encoder feature \( f^l_t \),

\[
\begin{align*}
    r^l &= \sigma(W_r * [f^l_{\text{enc}}, \hat{s}^{l+1}]), \\
    z^l &= \sigma(W_z * [f^l_{\text{enc}}, \hat{s}^{l+1}]), \\
    s^l &= r^l \odot \hat{s}^{l+1}, \\
    \hat{f}^l_t &= \tanh(W_h * [f^l_{\text{enc}}, s^l]), \\
    f^l_t &= (1 - z^l) \odot s^{l+1} + z^l \odot \hat{f}^l_t,
\end{align*}
\]

where \( \odot \) denotes the convolution operation, \( \odot \) denotes entry-wise product, and \( \sigma(\cdot) \) stands for the sigmoid function.

The introduction of the reset gate \( r^l \) and update gate \( z^l \) allows us control the contribution of hidden state, difference attribute vector, and encoder feature in a selective manner. Moreover, the convolution transform and linear interpolation in Eqns. (6) and (7) provide an adaptive means for the transfer of encoder feature and its combination with hidden state. In comparison to GRU where \( f^l_t \) is adopted as the output of hidden state, we take \( s^l \) as the output of hidden state and \( f^l_t \) as the output of transformed encoder feature. And experiments empirically validate that such modification can bring moderate gains on attribute generation accuracy.

3.4. Network Architecture

Our STGAN is comprised of two components, i.e., a generator \( G \) and a discriminator \( D \). Fig. 5 illustrates the network structure of \( G \) consisting of an encoder \( G_{\text{enc}} \) for abstract latent representation and a decoder \( G_{\text{dec}} \) for target image generation. The encoder \( G_{\text{enc}} \) contains five convolution layers with kernel size 4 and stride 2, while the decoder \( G_{\text{dec}} \) has five transposed convolution layers. Besides, STU is applied right after each of the first four encoder layers, denoted by \( (f^l_t, s^l) = G_{\text{st}}(f^l_{\text{enc}}, s^{l+1}, \text{att}_{\text{diff}}) \).
Thus the reconstruction loss is defined as,

\[ \mathcal{L}_{\text{rec}} = \| x - G(x, 0) \|_1, \]  

where the \( \ell_1 \)-norm \( \cdot \|_1 \) is adopted for preserving the sharpness of reconstruction result.

**Adversarial loss.** When the target attributes are different from source ones, i.e., \( \mathbf{att}_{\text{diff}} \neq 0 \), the ground-truth of editing result will be unavailable. Therefore, adversarial loss \([9]\) is employed for constraining the editing result to be indistinguishable from real images. In particular, we follow Wasserstein GAN (WGAN) \([3]\) and WGAN-GP \([10]\), and define the losses for training \( D_{\text{adv}} \) and \( G \) as,

\[
\begin{align*}
\max_{D_{\text{adv}}} \mathcal{L}_{D_{\text{adv}}} &= \mathbb{E}_x D_{\text{adv}}(x) - \mathbb{E}_{\hat{y}} D_{\text{adv}}(\hat{y}) + \\
& \quad \lambda \mathbb{E}_x \left( \| \nabla_x D_{\text{adv}}(x) \|_2 - 1 \right)^2, \\
\max_{G} \mathcal{L}_{G_{\text{adv}}} &= \mathbb{E}_{x, \mathbf{att}_{\text{diff}}} D_{\text{adv}}(G(x, \mathbf{att}_{\text{diff}})),
\end{align*}
\]

where \( \hat{x} \) is sampled along lines between pairs of real and generated images.  

**Attribute manipulation loss.** Even the ground-truth is missing, we can require the editing result to possess the desired target attributes. Thus, we introduce an attribute classifier \( D_{\text{att}} \) which shares the convolution layers with \( D_{\text{adv}} \), and define the following attribute manipulation losses for training \( D_{\text{att}} \) and generator \( G \),

\[
\begin{align*}
\mathcal{L}_{D_{\text{att}}} &= - \sum_{i=1}^{c} | \mathbf{att}_i^{(i)} | \log D_{\text{att}}^{(i)}(x) + \\
& \quad \left( 1 - \mathbf{att}_i^{(i)} \right) \log \left( 1 - D_{\text{att}}^{(i)}(x) \right), \\
\mathcal{L}_{G_{\text{att}}} &= - \sum_{i=1}^{c} | \mathbf{att}_i^{(i)} | \log D_{\text{att}}^{(i)}(\hat{y}) + \\
& \quad \left( 1 - \mathbf{att}_i^{(i)} \right) \log \left( 1 - D_{\text{att}}^{(i)}(\hat{y}) \right),
\end{align*}
\]

where \( \mathbf{att}_i^{(i)} \) (\( D_{\text{att}}^{(i)}(x) \)) denotes the \( i \)-th attribute value of \( \mathbf{att}_{s/t} (D_{\text{att}}(x)) \).
Model Objective. Taking the above losses into account, the objective to train the discriminator $D$ can be formulated as,

$$\min_D \mathcal{L}_D = -\mathcal{L}_{D_{adv}} + \lambda_1 \mathcal{L}_{D_{att}},$$

(17)

and that for the generator $G$ is,

$$\min_G \mathcal{L}_G = -\mathcal{L}_{G_{adv}} + \lambda_2 \mathcal{L}_{G_{att}} + \lambda_3 \mathcal{L}_{rec}.$$  

(18)

where $\lambda_1$, $\lambda_2$, and $\lambda_3$ are the model tradeoff parameters.

4. Experiments

We train the model by the ADAM [15] optimizer with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The learning rate is initialized as $2 \times 10^{-4}$ and decays to $2 \times 10^{-5}$ for fine-tuning after 100 epochs. In all experiments, the tradeoff parameters in Eqns. (17) and (18) are set to $\lambda_1 = 1$, $\lambda_2 = 10$ and $\lambda_3 = 100$. All the experiments are conducted in the TensorFlow [1] environment with cuDNN 7.1 running on a PC with Intel(R) Xeon(R) E3-1230v5 CPU 3.40GHz and Nvidia GTX1080Ti GPU. The source code can be found at https://github.com/csmliu/STGAN.git.

4.1. Facial Attribute Editing

Following [7, 11], we first evaluate our STGAN for arbitrary facial attribute editing on the CelebA dataset [23] which has been adopted by most relevant works [7, 11, 26].

Dataset and preprocessing. The CelebA dataset [23] contains 202,599 aligned facial images cropped to 178 × 218, with 40 with/without attribute labels for each image. The images are divided into training set, validation set and test set. We take 1,000 images from the validation set to assess the training process, use the rest of the validation set and the training set to train our STGAN model, and utilize the test set for performance evaluation. We consider 13 attributes, including Bald, Bangs, Black Hair, Blond Hair, Brown Hair, Bushy Eyebrows, Eyeglasses, Male, Mouth Slightly Open, Mustache, No Beard, Pale Skin and Young, due to that they are more distinctive in appearance and cover most attributes used by the relevant works. In our experiment, the central 170 × 170 region of each image is cropped and resized to 128 × 128 by bicubic interpolation. Training and inference time please refer to the suppl.

Qualitative results. We compare STGAN with four competing methods, i.e., IcGAN [26], FaderNet [17], AttGAN [11] and StarGAN [7]. The qualitative results are shown in Fig. 6. The results of AttGAN are generated by the released model, and we retrain other models for a fair comparison. It can be observed from Fig. 6, all the competing methods are still limited in manipulating complex attributes, e.g., Bald, Hair, and Age, and are prone to over-smoothing results. Besides, their results are more likely to be insufficiently modified and photo non-realistic when dealing with complex and/or multiple attributes. In comparison, our STGAN is effective in correctly manipulating the desired attributes, and can produce results with high image quality. More editing results are given in the suppl.

Quantitative evaluation. The performance of attribute editing can be evaluated from two aspects, i.e., image quality and attribute generation accuracy. Due to the unavailability of editing result, we resort to two alternative measures for quantitative evaluation of our STGAN. First, we use the training set of STGAN to train a deep attribute classification model which can attain an accuracy of 94.5% for the 13 attributes on the test set. Then Fig. 7 shows the attribute generation accuracy, i.e., classification accuracies on the changed attributes of editing results. It can be seen that
our STGAN outperforms all the competing methods with a large margin. For the attributes Bald, Black Hair, Brown Hair, and Eyebrows, STGAN achieves 20% accuracy gains against the competing methods.

As for image quality, we keep target attribute vector the same as the source one, and give the the PSNR/SSIM results of reconstruction in Table 2. Benefited from the STUs and difference attribute vector, our STGAN achieves much better reconstruction (> 7 dB by PSNR) in comparison to AttGAN and STGAN. The result is consistent with Fig. 2. The reconstruction ability of IcGAN is very limited due to the training procedure. FaderNet obtains better reconstruction results, mainly ascribing to that each FaderNet model is trained to deal with only one attribute.

User study. User study on a crowdsourcing platform is conducted to evaluate the generation quality of three top-performance methods, i.e., AttGAN, StarGAN and STGAN. We consider 11 tasks for 13 attributes, as the transfer among Blond Hair, Black Hair and Brown Hair are merged into Hair Color. For each task, 50 validated people participate in and each of them is given 50 questions. In each question, people are given a source image randomly selected from test set and the editing results by AttGAN, StarGAN and STGAN. For a fair comparison, the results are shown in a random order. The users are instructed to choose the best result which changes the attribute more successfully, is of higher image quality and better preserves the identity and fine details of source image. The results are shown in Table 3, and STGAN has higher probability to be selected as the best method on all the 11 tasks.

4.2. Season Translation

We further train our STGAN for image-to-image translation between summer and winter using the dataset released by CycleGAN [34]. The dataset contains photos of Yosemite, including 1,231 summer and 962 winter images in the training set, and 309 summer and 238 winter images for testing. We also randomly select 100 images from the training set to validation. All images are used as the original size of 256 × 256.

We compare our STGAN with AttGAN [11], StarGAN [7], and CycleGAN released by Zhu et al. [34]. Note that CycleGAN uses two generators respectively for summer→winter and winter→summer translation, while the other three methods conduct the two tasks with a single model. Fig. 8 shows several examples of translation results. It can be seen that STGAN performs favorably against the competing methods. We also conduct a user study using the same setting for facial attribute editing. From Table 4, our STGAN has a probability of more than 50% to win among the four competing methods.

5. Ablation Study

Using facial attribute editing, we implement several variants of STGAN, and evaluate them on CelebA [25] to assess the role of difference attribute vector and STUs. Concretely, we consider six variants, i.e., (i) STGAN: original STGAN, (ii) STGAN-dst: substituting difference attribute vector with target attribute vector, (iii) STGAN-conv: instead of STU, applying a convolution operator by taking encoder feature and difference attribute vector as input to modify encoder feature, (iv) STGAN-conv-res: adopting
the residual learning formulation to learn the convolution operator in STGAN-conv, (v) STGAN-gru: replacing STU with GRU in STGAN, (vi) STGAN-res: adopting the residual learning formulation to learn the STU in STGAN. We also train AttGAN and StarGAN models with difference attribute vector, denoted by AttGAN-diff and StarGAN-diff. Figs. 9 and 10 show their results on attribute manipulation. Please refer to the suppl. for qualitative results.

**Difference attribute vector vs. target attribute vector.** In Fig. 9, we present the comparison results of AttGAN, StarGAN and STGAN-dst with their counterparts (i.e., AttGAN-diff, StarGAN-diff and STGAN) by using difference attribute vector. One can see that difference attribute vector generally benefit attribute generation accuracy for all the three models. Moreover, empirical studies show that the use of difference attribute vector gives rise to training stability as well as image reconstruction performance. Note that while AttGAN-diff and StarGAN-diff perform better than AttGAN and StarGAN, they still suffer from the poor image quality.

**Selective Transfer Unit vs. its variants.** Fig. 10 reports the attribute generation accuracy of several STGAN variants for transforming encoder feature conditioned on difference attribute vector. The two convolutional methods, i.e., STGAN-conv and STGAN-conv-res, are significantly inferior to STGAN, indicating that they are limited in selective transfer of encoder feature. In comparison to STGAN-conv, STGAN-conv-res achieves relatively higher attribute generation accuracy. So we also compare STGAN with STGAN-res to check whether STU can be improved via residual learning. However, due to the selective ability of STUs, further deployment of residual learning cannot bring any gains for most attributes, and performs worse for several global (e.g., Gender, Age) and fine (e.g., Mustache, Beard) attributes. Finally, STGAN is compared with STGAN-gru by using transformed feature as hidden state. Although STGAN-gru performs better on Bald, STGAN is slightly superior to STGAN-gru for most attributes and the gain is notable for attributes Gender and Mustache.

**6. Conclusion**

In this paper, we study the problem of arbitrary image attribute editing for selective transfer perspective, and present a STGAN model by incorporating difference attribute vector and selective transfer units (STUs) in encoder-decoder network. By taking difference attribute vector rather than target attribute vector as model input, our STGAN can focus on editing the attributes to be changed, which greatly improves the image reconstruction quality and enhances the flexible translation of attributes. Furthermore, STUs are presented to adaptively select and modify encoder feature tailored to specific attribute editing task, thereby improving attribute manipulation ability and image quality simultaneously. Experiments on arbitrary facial attribute editing and season translation show that our STGAN performs favorably against state-of-the-arts in terms of attribute generation accuracy and image quality of editing results.

**Acknowledgement.** This work was supported in part by the National Natural Science Foundation of China under grant No. 61671182 and 61872118.
References


