Multi-Attribute Transfer via Disentangled Representation

Jianfu Zhang,[1] Yuanyuan Huang,[1] Yaoyi Li,[1] Weijie Zhao,[2] Liqing Zhang[1]*
[1]Shanghai Jiao Tong University, [2]Versa

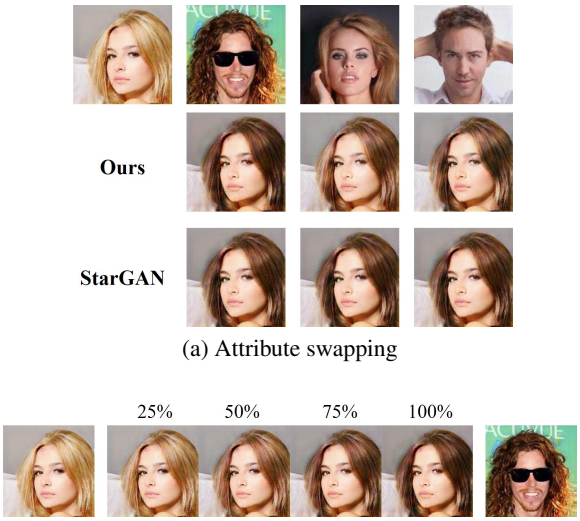
Abstract

Recent studies show significant progress in image-to-image translation task, especially facilitated by Generative Adversarial Networks. They can synthesize highly realistic images and alter the attribute labels for the images. However, these works employ attribute vectors to specify the target domain which diminishes image-level attribute diversity. In this paper, we propose a novel model formulating disentangled representations by projecting images to latent units, grouped feature channels of Convolutional Neural Network, to disassemble the information between different attributes. Thanks to disentangled representation, we can transfer attributes according to the attribute labels and moreover retain the diversity beyond the labels, namely, the styles inside each image. This is achieved by specifying some attributes and swapping the corresponding latent units to "swap" the attributes appearance, or applying channel-wise interpolation to blend different attributes. To verify the motivation of our proposed model, we train and evaluate our model on face dataset CelebA. Furthermore, the evaluation of another facial expression dataset RaFD demonstrates the generalizability of our proposed model.

Introduction

Image-to-image translation gains significant popularity in recent years, benefiting from the progress of generative models like Generative Adversarial Network (Goodfellow et al. 2014) and Variational Auto-Encoder (Kingma and Welling 2013). Generally speaking, image-to-image translation is defined as a task to translate an image from one domain to another, with representative applications such as colorization(Isola et al. 2017; Zhang, Isola, and Efros 2016), super-resolution(Ledig et al. 2017; Dong et al. 2016) and facial attribute transfer(Choi et al. 2018; Bouchacourt, Tomioka, and Nowozin 2018), etc.

We define a *domain* as a set of images sharing the same attribute labels, where the term *attribute* is denoted as a meaningful instance-level feature inherent in an image such as hair color or style, gender or age, and smiles. For each attribute, there can be different attribute labels such as black/blond/brown for hair color, or male/female for gender.



(a) Attribute swapping

(b) Attribute blending

Figure 1: An illustration of the motivation for our proposed model. We want to change the source image on the upper left, a lady with blond hair, to target images with brown hairs in the upper row. Give the target label, brown hair, previous methods like StarGAN (Choi et al. 2018) can only change the hair color to one style. However, for our proposed method, the target attribute is specified with another image, i.e., the hair color is transferred from the topmost images, and the color shade is retained, as shown in the second row. Besides, we can blend different attribute values by interpolating hidden units, changing hair color from golden to brown gradually.

Furthermore, beyond the attribute label, there can be image-level diversity in the style of different image instances with the same attribute label which can not simply be depicted by labels. For example, bang hairs may vary in directions, and smiles can have different attitudes. We call this diversity as concept, which can be formulated as a distribution of certain attribute appearance.

Previous works have explored a lot in image-to-image translation topic and achieved significant progress (Zhu et al. 2017; Isola et al. 2017; Choi et al. 2018; Xiao, Hong, and

*correspondence author

Ma 2017). However, these works either are only capable of mapping between a paired domains with poor flexibility for multi-domain translation or employing attribute vectors to specify the target domain which diminishes image-level attribute diversity.

In this paper, we propose a method which conducts attributes translation by transferring attributes from one image to another and keep the styles for each image at the same time. We implement our model using an auto-encoder, training on labeled input data pairs by swapping designated parts of embeddings. The transfer process is implemented via embedding images in disentangled representations and substituting the corresponding part of the embeddings. *A disentangled representation can be defined as one where single latent units are sensitive to changes in single generative factors while being relatively invariant to changes in other factors* (Higgins et al. 2017; Bengio, Courville, and Vincent 2013). The disentangling process aims at learning dimension-wise interpretable representations from data, and the attributes which are independent of each other can be disentangled into a well-defined feature space. For example, given an image dataset of human faces, the generative factors in the aforesaid definition is equivalent to the interpretable attributes like facial expression, and disentangling should produce representations or embeddings for each part corresponding to the attribute. Thanks to disentangled representations, we formulate our model with a manipulable and interpretable representation of images, transferring attributes from one to another and keep the image-level diversity in the style of attributes.

Specifically, our model differs from the previous image-to-image translation and disentangling models in the following aspects, which are also shown in Figure 1:

- We propose a model that learns disentangled representations for multiple different attributes, implementing a function that user can select some specific attributes and swap the corresponding parts of embeddings to "swap" the attributes, or apply embedding interpolation to manipulate different attributes;
- By disentangling different attributes, we synchronously transfer the attributes and retain the intra-attribute diversity, preserving the styles of each image;
- Experiments show that we provide qualitative results on selected datasets with high quality synthesized images and disentangled representation.

Related Works

In this section, we review several previous works for image-to-image translation applications which are closely related to our proposed method, especially for those which are based on Generative Adversarial Network and Disentangled Representation.

Image-to-image transformation applications (Gong et al. 2018; Chang et al. 2018; Isola et al. 2017; Zhu et al. 2017; Choi et al. 2018; Yan et al. 2016) progress rapidly with the help of Generative Adversarial Network (GAN)(Goodfellow et al. 2014) and Variational Auto-Encoder (VAE) (Kingma and Welling 2013) frameworks. Thanks to the recent progress (Arjovsky, Chintala, and Bottou 2017; Gulrajani et al. 2017; Miyato et al. 2018), GAN has now achieved improved stability and better performance. Pix2Pix (Isola et al. 2017) converts images from source style to destination style, using conditional GAN trained with pair-wised examples. Cycle-GAN (Zhu et al. 2017) breaks the limitation of pair-wised data by training mapping and reverse mapping networks cross two domains, but still suffering from limited flexibility for multi-domain transformation since one model can only deal with a single pair of domains. StarGAN (Choi et al. 2018) framework trains a multi-domain model using only one single model which further improve the scalability and efficiency of image-to-image translation tasks. However, these aforementioned methods translate images according to the specified target attribute labels, while image-level variances for attributes are dropped for synthesized images.

The goal of Disentangled Representation (Bengio, Courville, and Vincent 2013) is to extract explanatory factors of the data in the input distribution and generate a more meaningful representation. Recently, there are many exciting works based on this topic (Jha et al. 2018; Mathieu et al. 2016; Hadad, Wolf, and Shahar 2018; Donahue et al. 2018; Huang et al. 2018), here we will discuss some classic and related methods. InfoGAN (Chen et al. 2016) utilizes GAN framework and maximizes the mutual information between a subset of the latent variables to learn disentangled representations in an unsupervised manner. β-VAE (Higgins et al. 2017) . Although these unsupervised methods are unable to learn specific meaningful attributes, they provided elegant baselines for the later works to learn to disentangle representations for annotated attributes. ML-VAE (Bouchacourt, Tomioka, and Nowozin 2018) separates the latent representation into semantically meaningful parts, i.e., style and content. The content is common for a group, while the style can differ within the group. DC-IGN (Kulkarni et al. 2015) was proposed to disentangle factors by changing a single attribute and constraining other factors by feeding other attributes with average attribute values.

Some previous works share similar inspiration with our proposed model. DSD (Feng et al. 2018) constructs an "encoding-swap-decoding" process where shared attributes are swapped to disentangle different attributes with paired labels, which is similar to our model while DSD can only swap the attributes that have the same attribute values. DNA-GAN (Xiao, Hong, and Ma 2017) also formulates a swapping strategy to disentangle different attributes via GAN framework. However, DNA-GAN is not capable of reserving image-level feature details or handling the attributes with more than two label values like hair color (e.g., black, blond, and brown). In Gonzalez-Garcia, van de Weijer, and Bengio (2018), the authors combine the disentanglement objective with image-to-image translation between a pair of domains by disentangling the embeddings into three parts: one domain-invariant part and two domain-specific parts corresponding to two domains. Different from the above works, our framework can disentangle multiple attributes simultaneously within only one model and generate images with high quality.

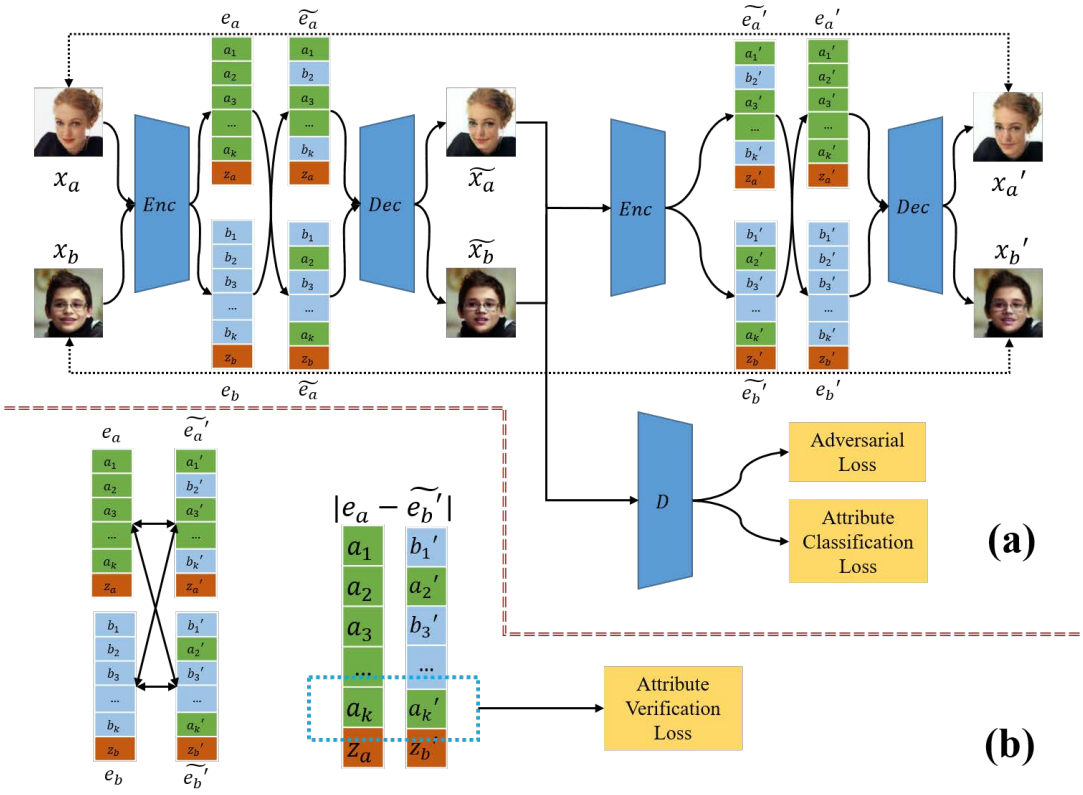


Figure 2: (a). An illustration of the training pipeline of our work. Given two input images (x_a, x_b), we encode each of them into embeddings (e_a, e_b), and divide each embedding into latent units as the filled rectangles in the diagram. Each unit in green indicates the information of a specific attribute from x_a, blue for x_b and orange for attribute-irrelevant parts. Subsequently, we select and swap some of the units and decode the embeddings into synthesized images. The whole process will be repeated to reconstruct the input images with the same encoder and decoder. (b). An illustration of the structure of Attribute Verification Loss. We compute and optimize attribute verification loss for four pairs of embeddings. For each pair, say e_a and $\tilde{e'_b}$, we calculate the absolute difference between the two selected embeddings. Then a binary classification loss will be applied to each latent unit pair to recognize whether the two units are representing attribute information from the same input image.

Method

The goal of our model is to exchange some of the attributes between a pair of images. To this end, an auto-encoder architecture $G = (Enc, Swap, Dec)$ first project images to embeddings which comprise different latent units, such that attribute exchanging can be accomplished by swapping the corresponding latent units, and then decodes the embeddings to synthesize images. To synthesize plausible images, we employ a discriminator D to formulate a cycle-consistent GAN framework for our model. The overall pipeline of our proposed model is shown in Figure 2.

Define a dataset of multi-label images $\mathcal{X}$ with a set of labels $\mathcal{Y}$. k denotes the number of different attributes in $\mathcal{Y}$ that for any $x \in \mathcal{X}$ there is a label $y = (y_1, \ldots, y_k) \in \mathcal{Y}$ with k different attribute annotations. Given two images $(x_a, x_b) \in \mathcal{X}^2$ with label $(y_a, y_b) \in \mathcal{Y}^2$, Enc encodes (x_a, x_b) into *attribute embeddings* (e_a, e_b), where e_a (and the same for e_b) can be further divided into $k+1$ different *latent units* $e_a = (a_1, a_2, \ldots, a_k, z_a)$. Each of the unit, which consists of a specific number of channels of feature map, represents a single attribute in all k attributes $(a_1, \ldots, a_k)$ or attribute-irrelevant part z_a such as image background. After encoding the images into embeddings, $Swap$ is applied. We select some of the attributes following specific strategies and swap the corresponding latent units, transforming (e_a, e_b) to $(\tilde{e}_a, \tilde{e}_b)$. Afterwards, a decoder Dec is adopted to generate synthetic images $(\tilde{x}_a, \tilde{x}_b)$, taking $(\tilde{e}_a, \tilde{e}_b)$ as input. For discriminator network $D : x \mapsto \{D_{adv}(x), D_{cls}(x)\}$, D_{adv} is introduced to enhance the quality of the synthetic images while D_{cls} ensures the swapped units do affect their corresponding attributes properly. D_{adv} and D_{cls} share the same network structure and parameters except for the last output layer. Moreover, our model is trained under the cycle-consistency constraint, which means we repeat the foregoing encode-swap-decode process with the same encoder and decoder on image pair (x'_a, x'_b) to swap back the selected attributes and reconstruct the original image pair (x_a, x_b). Specifically, Enc, for this time, encodes the synthetic images $(\tilde{x}_a, \tilde{x}_b)$ to new embeddings denoted as $(\tilde{e'_a}, \tilde{e'_b})$, then the pre-selected latent units are re-swapped back to (e'_a, e'_b).

Finally we apply Dec to (e'_a, e'_b) and generate two reconstructed images (x'_a, x'_b). To attain disentanglement of the mutual information between different attributes and constrain the generated images with given attribute information, we design four different losses: adversarial loss, attribute classification loss, attribute verification loss and reconstruction loss.

Attributes Swapping Strategy

Before we describe the losses we used to train our model, let's see how we swap the attributes for training. For two input images x_a and x_b, labeled as $y_a = (y_a^1, \ldots, y_a^k)$ and $y_b = (y_b^1, \ldots, y_b^k)$ and embedded as $e_a = (a_1, \ldots, a_k, z_a)$ and $e_b = (b_1, \ldots, b_k, z_b)$, we have two different strategies to choose an attribute set $\mathcal{S}$ and swap the units corresponding to all attributes in $\mathcal{S}$, changing the attribute embeddings to $(\tilde{e}_a, \tilde{e}_b)$ and the labels to $(\tilde{y}_a, \tilde{y}_b)$. We divide all the attributes into two sets $\mathcal{P} = \{i | y_a^i = y_b^i\}$ and $\mathcal{N} = \{j | y_a^i \neq y_b^i\}$. Then randomly choose one swapping strategy from the followings:

- Randomly choose an subset $\mathcal{A} \subseteq \mathcal{P}$ and let $\mathcal{S} = \mathcal{A} \cup \mathcal{N}$, then $\tilde{y}_a = y_b$ and $\tilde{y}_b = y_a$.
- Randomly choose an subset $\mathcal{A} \subseteq \mathcal{P}$ and let $\mathcal{S} = \mathcal{A}$, then $\tilde{y}_a = y_a$ and $\tilde{y}_b = y_b$;

These two choices ensure that after $Swap$, the new embeddings labels $\tilde{y}_a$ and $\tilde{y}_b$ satisfy $\tilde{y}_a \in \mathcal{Y}$ and $\tilde{y}_b \in \mathcal{Y}$. Under this circumstance, the generator G can only synthesize images with observed attribute label combinations, which means that the real and the synthesized image will have the same attribute label distribution, making the whole GAN framework more robust. Note that the background embeddings z will never be swapped and the two strategies are chosen with equal probability during training.

Adversarial Loss

GAN(Goodfellow et al. 2014) loss is involved in our framework to improve the quality of images generated by the generator G. We leverage WGAN (Arjovsky, Chintala, and Bottou 2017) with gradient panelty (Gulrajani et al. 2017) which is a variation of GAN framework and also PatchGAN (Isola et al. 2017) which discriminate whether local patches are real or fake. The loss function is defined as

$$\begin{aligned} \mathcal{L}_{adv} &= \mathcal{L}_D + \mathcal{L}_G + \lambda_{gp}\, \mathbb{E}_{\hat{x}}[(||\nabla_{\hat{x}} D(\hat{x})||_2 - 1)^2] \\ \mathcal{L}_D &= \mathbb{E}_x[ReLU(1 - D_{adv}(x))] + \mathbb{E}_{\tilde{x}}[ReLU(1 + D_{adv}(\tilde{x}))] \\ \mathcal{L}_G &= -\mathbb{E}_{\tilde{x}}[D_{adv}(\tilde{x})], \end{aligned} \tag{1}$$

where x and $\tilde{x}$ represent real and synthesized images and $\hat{x} = \epsilon x + (1 - \epsilon)\tilde{x}$, the random number $\epsilon \sim U[0, 1]$. Hinge loss is adopted to stabilize the training process. For technical details of GAN frameworks please refer to Gulrajani et al. (2017); Isola et al. (2017); Arjovsky, Chintala, and Bottou (2017). Note that $\mathcal{L}_D$ only optimizes D while $\mathcal{L}_G$ only optimizes G.

Attribute Classification Loss

Attribute classification loss (Odena, Olah, and Shlens 2017) is introduced to help G generate images with corresponding attribute labels properly which will ensure that our model swaps the attributes correctly, also preserve and disentangle the information for different latent units. To achieve this goal we first train a standard classifier by applying binary cross-entropy losses to optimize D with the real images (x_a, x_b) and their attribute labels (y_a, y_b)

$$\mathcal{L}_{cls}^D = \mathbb{E}_{x,y}[-logD_{cls}(y|x)] \tag{2}$$

then optimize G with synthesized images $(\tilde{x}_a, \tilde{x}_b)$ and swapped attribute labels $(\tilde{y}_a, \tilde{y}_b)$

$$\mathcal{L}_{cls}^G = \mathbb{E}_{\tilde{x},\tilde{y}}[-logD_{cls}(\tilde{y}|\tilde{x})]. \tag{3}$$

Note that we learn a classifier and a discriminator with a shared weights network. It has been proved in Odena, Olah, and Shlens (2017); Choi et al. (2018) that it is stable to train and synthesize high-quality images using shared weights for the classifier and discriminator.

Claim of Disentanglement. The attribute classification loss guarantees that attribute labels keep unchanged for unswapped attributes. Hence we can say the swapped units will not affect the attribute appearance that corresponds to un-swapped ones.

Cycle-Consistency Loss

The aforementioned adversarial loss and attribute losses are merely beneficial to the reality and attribute exactitude of synthesized images, whereas they are unable to preserve detailed information in the original images including details of attributes and attribute-irrelevant part. Inspired by the previous work (Zhu et al. 2017), we apply a cycle consistency loss to incentivize the generator to retain pixel-wise information. Suppose input images (x_a, x_b) are translated to $(\tilde{x}_a, \tilde{x}_b)$ after a Enc-$Swap$-Dec process with a specific attribute swapping choice s, we re-apply the same process with the same attribute swapping choice s to $(\tilde{x}_a, \tilde{x}_b)$, thus synthesizing reconstructed images (x'_a, x'_b). Then the cycle-consistency loss is defined as

$$\mathcal{L}_{rec} = ||x_a - x'_a||_1 + ||x_b - x'_b||_1. \tag{4}$$

where L1 loss is adopted as the reconstruction loss. Note that (x_a, x_b) and (x'_a, x'_b) share the same attribute labels respectively since the same latent units are swapped during the two $Swap$ steps.

Attribute Verification Loss

To further disentangle information among different attributes and preserve image-level style diversity within a domain, We introduce the attribute verification loss to verify whether a pair of two latent units corresponding to the same attribute (e.g., a_1 and a'_1) are extracted from the same image . An illustrative structure of attribute verification loss is shown in Figure 2(b). This attribute verification loss imposes a style aspect constraint to our framework, and it focuses on the style details instead of semantic labels. A unit pair that has the same label and comes from different images will be pushed far away from the unit pairs and pulled close to the ones from the same images. The motivation is that embeddings should reflect the difference of styles between images,

like the difference of labels, to avoid a trivial embedding which only contains the information of attribute labels and lose the intra-attribute diversity.

We compute and optimize attribute verification loss on four pairs of attribute embeddings: $(e_a, \tilde{e'_a})$, $(e_a, \tilde{e'_b})$, $(e_b, \tilde{e'_a})$ and $(e_b, \tilde{e'_b})$. For each pair, channel-wise absolute difference is calculated between the paired latent units. Then we apply a fully-connected layer within each unit pair outputting the predicted verification label, and optimize a binary classification loss with logistic regression.

The verification label l_i is defined as 1 for latent unit pairs $(a_i, \tilde{a'_i})$ and $(b_i, \tilde{b'_i})$, and 0 for latent unit pairs $(a_i, \tilde{b'_i})$ and $(b_i, \tilde{a'_i})$, where $i \in \{1, ..., k\}$. Note that for those unit pairs with same attribute labels but from different images, namely $(a_i, \tilde{b'_i})$ with $y_a^i = y_b^i$, the attribute verification process is supposed to classify them as the unit pairs with different attribute labels. The attribute verification loss between e_a and $\tilde{e'_b}$ can be defined as:

$$V(e_a, \tilde{e'_b}) = \frac{1}{k}\sum_{i=1}^{k}[-l_i log(p_i) - (1 - l_i) log(1 - p_i)]. \quad (5)$$

Where p_i is the predicted label corresponding to the groundtruth label l_i. Then the equation for the attribute verification loss is the sum of the verification loss for four different pairs of attribute embeddings:

$$\mathcal{L}_{ver} = V(e_a, \tilde{e'_a}) + V(e_a, \tilde{e'_b}) + V(e_b, \tilde{e'_a}) + V(e_b, \tilde{e'_b}). \quad (6)$$

Since the latent unit pair from the same image, like $(a_i, \tilde{a'_i})$, is expected to be identical after a *Enc-Dec* process (in contrast to the unit pairs from different images). The attribute verification loss can also be partially viewed as an embedding reconstruction loss, which promotes robustness of the auto-encoder.

Claim of Disentanglement. With attribute verification loss, the feature embeddings extracted from synthesized images (i.e., $\tilde{e'}$) can reflect the attribute changes in corresponding latent units. Hence the unit pairs containing these units come from different images will be pushed away with different verification labels, even if two latent units in a pair have the same attribute labels. The latent units corresponding to those unchanged attributes tend to be relatively invariant with the same verification labels, which is reminiscent of the aforementioned definition of disentangled representation. This loss ensures that the changes in attributes indeed affect and only affect the corresponding latent units, which enhances the disentanglement of latent units.

Full Objective and Implementation Details

Finally, we combine all these losses as the objective function to optimize our model,

$$\mathcal{L} = \mathcal{L}_{adv} + \lambda_{cls}\mathcal{L}_{cls} + \lambda_{ver}\mathcal{L}_{ver} + \lambda_{rec}\mathcal{L}_{rec}. \quad (7)$$

We use $\lambda_{cls} = \lambda_{ver} = 0.1$ and $\lambda_{rec} = 10$ in our experiments. We followed the network structure proposed by Zhu et al. (2017); Choi et al. (2018), which contains two downsampling blocks in *Enc* an six residue blocks similar to the model proposed in He et al. (2016). We use Adam optimizer (Kingma and Ba 2014) with $\beta 1 = 0.5$ and $\beta 2 = 0.999$. Random horizontal flip are applied for data augmentation. We set batchsize to 16 and train our model with 200000 iterations with learning rate 0.0001. Spectral Normalization Layers (Miyato et al. 2018) and Batch Normalization Layers (Ioffe and Szegedy 2015) are applied to boost and stabilize the convergence of the network. We perform the update to G and D with the same frequency. We assign 16 channels of feature map for each latent unit and 160 channels for attribute-irrelevant parts. The whole traning process takes about one and a half days on a single NVIDIA Tesla P100 GPU.

Experiments

In this section, we show the experimental results of our proposed model. To demonstrate the effectiveness and generalizability of our proposed model, we train the model on a large public face dataset CelebA and test on CelebA and another facial expression dataset RaFD.

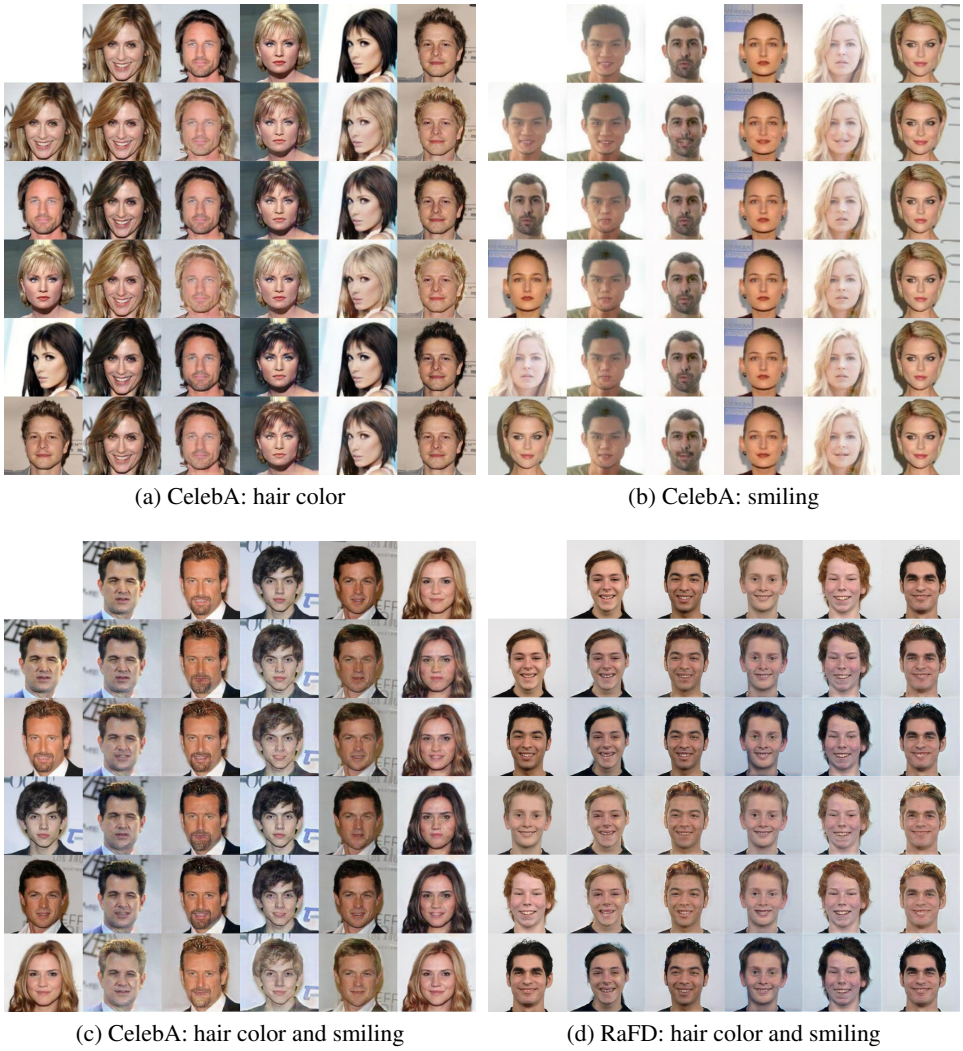
CelebA Dataset The CelebFaces Attributes (CelebA) (Liu et al. 2015) contains 202599 face images of celebrities around the world. Each of the images is annotated with 40 binary attributes. The original image size is 178×218, we apply center crop with the size of 178×178 and resize the image to 128×128. Following the setting of (Choi et al. (2018), we randomly select 2000 images from the whole dataset as the test set and the rest images are used for training. We select two attributes to transfer: hair color and smiling, which are clearly two attributes that can be represented in a disentangled fashion.

RaFD Dataset The Radboud Faces Database (RaFD) (Langner et al. 2010) contains 4824 images collected from 67 different persons. There are eight different facial expressions from 3 different gaze directions and camera angles captured from each person. We affine all the images according to the midpoint of the eyes of each image and crop the image with the size of 512×512 then resize the image to 128×128

Qualitative Results for CelebA Dataset

Note that all the results are generated by a single model, showing that we can disentangle different attributes and deal with multi-domain transfer with one single model. We only swap the specific latent units in the image embeddings and keep the other units (e.g., attribute-irrelevant part) unchanged to show the experimental results of attribute translation. Figure 3(a), Figure 3(b) and Figure 3(c) show the synthetic images after transferring a single facial attribute or a combination of two attributes. The leftmost column and the topmost row of each figure are real input images, and the 5x5 grid in the center of each figure exhibits synthesized images after transferring the specified attributes in the leftmost real images to the topmost ones.

Figure 3(a) shows the results of transferring hair color. As is shown in Figure 3(a), the proposed method achieves hair



(a) CelebA: hair color (b) CelebA: smiling

(c) CelebA: hair color and smiling (d) RaFD: hair color and smiling

Figure 3: Qualitative results for our model transferring the attributes from the leftmost real images to the topmost ones.

color transfer and generate sharp images with remarkable quality. Notably, all the other attributes remain intact, unaffected by the hair color transfer. In particular, the details of ears, skin color and background are perfectly preserved without any brightness variation. The experimental results validate that our multi-attribute transfer framework indeed facilitates the disentangling of different attributes. Although different persons have different hairstyles, our model only change the hair color in the image, indicating that the hair color attribute is well disentangled from other attributes.

For Figure 3(b), the attribute smiling is transferred at this time. As seen in Figure 3(b), synthesized images retain more detailed information beyond the attribute label, smiling or not smiling. We can also observe the difference between the smile, beam and grin in the figure. The reason is that our model clearly retains image-level details by use of attribute verification loss, preserving the style diversity within the same domain.

Figure 3(c) demonstrates the synthesized results of transferring two different attributes, smiling and hair color. Compared with Figure 3(b), the corresponding synthesized images are identical in every detail except for hair color. The style diversities of each attribute are preserved respectively, and we can say that the attributes of generated faces are swapped independently. Furthermore, the transfer of multi-attribute does not incur any background variation in the synthesized images, consistent with the previous experimental results. These results confirm the disentanglement of our explanatory feature representation.

Interpolation of the Latent Units

To demonstrate that we can manipulate different attributes independently with the help of the proposed framework, we conduct an interpolation experiment. In the experiments of interpolation, we first swap a specific latent unit and keep other units fixed to generate a synthetic image with only one

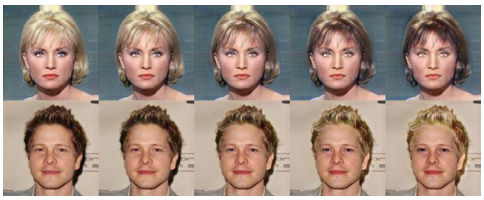


Figure 4: Example results for interpolation of latent units for the hair color attribute. The real images are put on the left side.

attribute label changed. Then we apply channel-wise linear interpolation to this latent unit between the real and synthesized image, which turn the discrete label value into a continuous one. Three more images are generated with different attribute styles from each interpolation process. In Figure 4, we display two results on the interpolation of latent units. The leftmost column is the input real images and the rightmost column shows the synthesized images with different hair color label. In the first row, we transfer blond hair to brown hair and change brown to blond in the second row. Three columns in the middle show the result of linear interpolation of the latent units learned by our model. All the other attributes and attribute-irrelevant parts stay unaltered during the linear interpolation process, while the hair color changes smoothly between blond and brown. This interpolation experiment provides one more evidence to confirm that our framework embeds the image to an explanatory disentangled representation.

Qualitative Results for RaFD Datset

To further prove the generalizability of our proposed model, we evaluate the model on the facial expression dataset RaFD. The model used for evaluation is trained on face dataset CelebA without any fine-tuning process.

The results of generated images are shown in Figure 3(d). The quality of images in Figure 3(d) is comparable to the experimental results of dataset CelebA. The appearance of unchanged attributes and background is consistent in each column and swapped latent unit leads to a proper attribute transfer. We also employ the squared image matrix as experiments on CelebA to illustrate that the proposed method can achieve attribution transfer between every image pair without any constraints. The synthesized images show the impressive generalizability of our model. Note that the results are not as good as CelebA. There are domain shifts issues between CelebA (i.e., the source domain) and RaFD (i.e., the target domain). In particular, backgrounds are complex while in RaFD backgrounds are pure white. It is well-known that the model trained on the source domain may perform much worse on the target domain when the data distributions between the two domains are considerably different (Torralba and Efros 2011).

Quantitative Results

Here we put quantitative results to show the performance comparisons between our model and the others. We only choose Star-GAN (Choi et al. 2018) and DNA-GAN (Xiao, Hong, and Ma 2017) which are designed for facial expression transfer task. We split the test set of CelebA into two parts A and B with equal quantity and construct pairs based on one-to-one matching between part A and part B. For StarGAN: We swap the attribute values for the images in A with the corresponding attribute values for the images in B, and vice versa. We use Frechet Inception Distance (FID), which is a sound measurement to evaluate the image qualities for GANs and introduced in (Heusel et al. 2017), to evaluate the models. For more details, please refer to the paper. Note that for smaller FID indicates better performance. For DNA-GAN: For all the images in A we swap the attributes which have different attribute values compared with the corresponding images in B, and vice versa. For ours: we swap the attributes with the same rule as for DNA-GAN. Since DNA-GAN is not able to transfer hair color attribute, we choose three other independent attributes to train: bangs, smiling and eyeglasses. The results are shown in Table 1, we can see that we significantly improved the quality of synthesized images compared with StarGAN and DNA-GAN.

Methods	StarGAN	DNA-GAN	Ours
FID	78.90	76.08	**45.23**

Table 1: Comparison among our method and other methods.

λ_{ver}	0	0.01	0.1	1	10
FID	55.15	52.39	**45.23**	60.98	70.72

Table 2: Ablative studies for attribute verification loss.

To show the effect of the attribute verification loss, we also run experiments to conduct ablative studies by changing λ_{ver}. The results are shown in Table 2, we can see that verification loss does help our model improve the quality of synthesized images because our model is not able to fully preserve the intra-attribute diversity without verification loss. And $\lambda_{ver} = 0.1$ is about the optimal hyper-parameter.

Conclusions

In this work, we proposed a novel model for image-to-image attributes transfer via disentangling image representations. Instead of specifying the target domain by attribute label, we project images to hidden units, i.e., grouped feature channels of Convolutional Neural Network, such that the specified attributes can be transferred by substituting corresponding latent units. On the one hand, the substituting process allows us to retain the diversity beyond the attribute labels, namely, the style inside each image. On the other hand, the disentanglement of hidden units assures that only the specified attributes are changed while the others remain intact. Experiments on face datasets CelebA and RaFD demonstrate that our proposed model is able to transfer attributes from one image to another with intact attribute patterns and synthesize highly realistic images.

Acknowledgement

The work was supported by the National Basic Research Program of China (Grant No. 2015CB856004), the Key Basic Research Program of Shanghai Municipality, China (15JC1400103,16JC1402800).

References

Arjovsky, M.; Chintala, S.; and Bottou, L. 2017. Wasserstein gan. In *ICML*.

Bengio, Y.; Courville, A.; and Vincent, P. 2013. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*.

Bouchacourt, D.; Tomioka, R.; and Nowozin, S. 2018. Multi-level variational autoencoder: Learning disentangled representations from grouped observations. In *AAAI*.

Chang, H.; Lu, J.; Yu, F.; and Finkelstein, A. 2018. Pairedcyclegan: Asymmetric style transfer for applying and removing makeup. In *CVPR*.

Chen, X.; Duan, Y.; Houthooft, R.; Schulman, J.; Sutskever, I.; and Abbeel, P. 2016. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *NIPS*.

Choi, Y.; Choi, M.; Kim, M.; Ha, J.-W.; Kim, S.; and Choo, J. 2018. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*.

Donahue, C.; Balsubramani, A.; McAuley, J.; and Lipton, Z. C. 2018. Semantically decomposing the latent spaces of generative adversarial networks. In *ICLR*.

Dong, C.; Loy, C. C.; He, K.; and Tang, X. 2016. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*.

Feng, Z.; Wang, X.; Ke, C.; Zeng, A.; Tao, D.; and Song, M. 2018. Dual swap disentangling. In *NIPS*.

Gong, Y.; Karanam, S.; Wu, Z.; Peng, K.-C.; Ernst, J.; and Doerschuk, P. C. 2018. Learning compositional visual concepts with mutual consistency. In *CVPR*.

Gonzalez-Garcia, A.; van de Weijer, J.; and Bengio, Y. 2018. Image-to-image translation for cross-domain disentanglement. In *NIPS*.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *NIPS*.

Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; and Courville, A. C. 2017. Improved training of wasserstein gans. In *NIPS*.

Hadad, N.; Wolf, L.; and Shahar, M. 2018. A two-step disentanglement method. In *CVPR*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*.

Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; and Lerchner, A. 2017. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*.

Huang, X.; Liu, M.-Y.; Belongie, S.; and Kautz, J. 2018. Multimodal unsupervised image-to-image translation. In *ECCV*.

Ioffe, S., and Szegedy, C. 2015. Batch normalization: accelerating deep network training by reducing internal covariate shift. In *ICML*.

Isola, P.; Zhu, J.-Y.; Zhou, T.; and Efros, A. A. 2017. Image-to-image translation with conditional adversarial networks. In *CVPR*.

Jha, A. H.; Anand, S.; Singh, M.; and Veeravasarapu, V. 2018. Disentangling factors of variation with cycle-consistent variational auto-encoders. In *ECCV*.

Kingma, D. P., and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P., and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kulkarni, T. D.; Whitney, W. F.; Kohli, P.; and Tenenbaum, J. 2015. Deep convolutional inverse graphics network. In *NIPS*.

Langner, O.; Dotsch, R.; Bijlstra, G.; Wigboldus, D. H.; Hawk, S. T.; and Van Knippenberg, A. 2010. Presentation and validation of the radboud faces database. *Cognition and emotion*.

Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A. P.; Tejani, A.; Totz, J.; Wang, Z.; et al. 2017. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep learning face attributes in the wild. In *ICCV*.

Mathieu, M. F.; Zhao, J. J.; Zhao, J.; Ramesh, A.; Sprechmann, P.; and LeCun, Y. 2016. Disentangling factors of variation in deep representation using adversarial training. In *NIPS*.

Miyato, T.; Kataoka, T.; Koyama, M.; and Yoshida, Y. 2018. Spectral normalization for generative adversarial networks. In *ICLR*.

Odena, A.; Olah, C.; and Shlens, J. 2017. Conditional image synthesis with auxiliary classifier gans. In *ICML*.

Torralba, A., and Efros, A. A. 2011. Unbiased look at dataset bias. In *CVPR*.

Xiao, T.; Hong, J.; and Ma, J. 2017. Dna-gan: Learning disentangled representations from multi-attribute images. *arXiv preprint arXiv:1711.05415*.

Yan, X.; Yang, J.; Sohn, K.; and Lee, H. 2016. Attribute2image: Conditional image generation from visual attributes. In *ECCV*.

Zhang, R.; Isola, P.; and Efros, A. A. 2016. Colorful image colorization. In *ECCV*.

Zhu, J.-Y.; Park, T.; Isola, P.; and Efros, A. A. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*.