JMLR: Workshop and Conference Proceedings 80:1–17, 2017 ACML 2017

ST-GAN: Unsupervised Facial Image Semantic Transformation Using Generative Adversarial Networks

Jichao Zhang ZHANG163220@GMAIL.COM
Fan Zhong ZHONGFAN@SDU.EDU.CN
College of Computer Science and Technology
Shandong university
Gongze Cao ASXAQZ2@GMAIL.COM
School of Mathematics
Zhejiang University
Xueying Qin QXY@SDU.EDU.CN
College of Computer Science and Technology
Shandong university

Editors: Jichao Zhang and Xueying Qin

Abstract

Facial Image semantic transformation aims to convert one image into another image with different semantic features (e.g., face pose, hairstyle). The previous methods, which learn the mapping function from one image domain to the other, require supervised information directly or indirectly. In this paper, we propose an unsupervised facial image semantic transformation method called semantic transformation generative adversarial networks (ST-GAN). We further improve ST-GAN with the Wasserstein distance to generate more realistic images and propose a method called local mutual information maximization to obtain a more explicit semantic transformation. ST-GAN has the ability to map the image semantic features into the latent vector and then perform transformation by controlling the latent vector. After the proposed framework is trained on a benchmark, the original face images can be reconstructed and then translated into various images with different semantic features.

Keywords: Semantic discovery; Unsupervised semantic transformation; Generative adversarial networks

1. Introduction

Image semantic transformation from one domain to another is a very interesting subject that has wide range of applications. Previous works have considered style transfer, black and white to color image, edges to photo Gatys et al. (2016), even image super-resolution Ledig et al. (2016) and image inpainting Pathak et al. (2016) have been investigated.

The essence of one-to-one image transformation is to learn a function that maps pixels to pixels from one domain to another domain. Previous image transformation methods can be categorized as *paired* or *unpaired* according to their requirements for the training datasets. As illustrated in Fig. 1, paired methods require each input training image to have a corresponding ground truth output, and thus, the training dataset are difficult to collect. Unpaired methods alleviate this problem by using only source and target datasets for which

it is not necessary to have a one-to-one correspondences of image instances. However, the required domain transformation still needs to be specified by collecting source and target datasets in specific domains; hence, they actually require one-to-one correspondences in the image domain.

Different from directly learning the mapping function between image domains, another method is based on disentangled representation learning and uses low-dimension latent codes to correspond to salient semantic features of the observation. In general, previous works need to binary label information for supervised training or testing and can be categorized as *unpaired* method. In this paper, we focus on unsupervised methods for facial image semantic transformation. Our proposed method, semantic transformation generative adversarial networks (ST-GAN), can be trained with an unlabeled training dataset that has a mixed image domain and therefore is unsupervised, as illustrated in Fig. 1 (*mixed*). By maximizing the mutual information between the output images and the latent code, the latent code corresponds to the very salient semantic features of the generated image. Hence, the facial semantic transformation of ST-GAN is achieved by first reconstructing the original face image, then changing the latent codes to translate the reconstruction result into more face images with different semantic features.

ST-GAN is based on InfoGAN Chen et al. (2016). After training InfoGAN, the mutual information between the generated instances and the latent code has been maximized, and this latent code will capture the variations of the semantic features. However, in infoGAN, we do not know which the latent code corresponds to which semantic features before we observe the result. For example, if we would like InfoGAN to discover different eye sizes in a face dataset, we might need to train it many times to find suitable parameters. To make semantic discovery more explicit, we propose a new method called *local mutual information maximization* along with a preprocessing binary mask function.

The contributions of our work are as follows:

- We propose an unsupervised facial image semantic transformation architecture called ST-GAN and this architecture can be trained from a mixed training dataset. ST-GAN can encode the salient semantic feature into latent code c which is the low dimension vector and decode c and z into the reconstruction results. Image semantic transformation can be achieved just by changing the value of the latent code.
- ST-GAN leverages the ability of semantic discovery to achieve various facial image semantic transformations, which is difficult for the previous supervised methods.
- We optimize ST-GAN using local mutual information maximization, named LST-GAN to make semantic discovery and transformation more explicit and regionally oriented.

2. Related work

In addition to the variational autoencoder(VAE) Kingma and Welling (2013), GAN Goodfellow et al. (2014) provides a powerful framework for a generation model that generates very sharp, realistic images Radford et al. (2015); Zhang et al. (2016); Nguyen et al. (2016). Many researchers have focused on improving the stability of training and the quality of generated images by applying deep learning skills or optimizing the objective function. Salimans

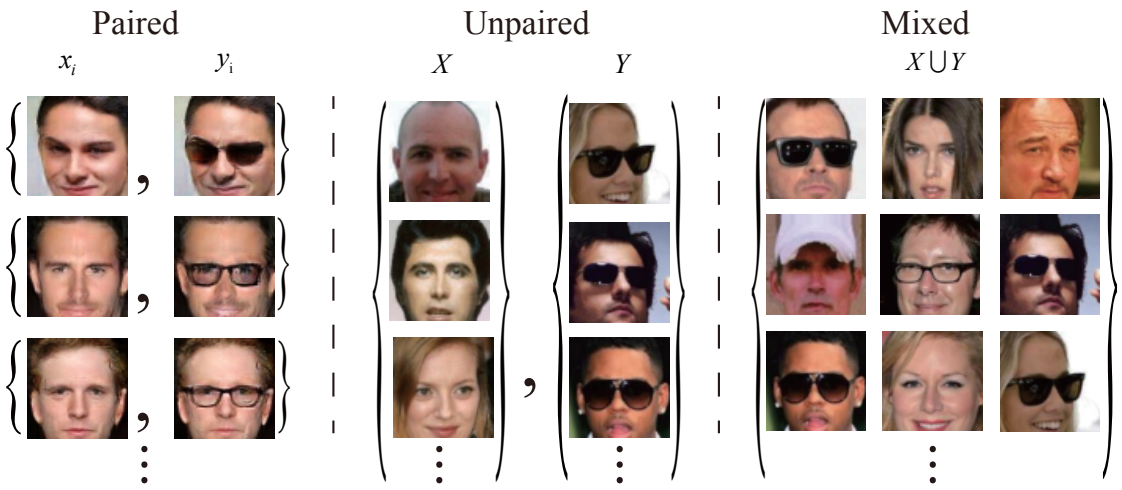


Figure 1: Comparison of previous face image training dataset (paired, unpaired) and our training dataset (mixed). Paired training datasets consist of samples $\{x_i, y_i\}_{i=1}^N$ where y_i need to correspond to each x_i. Unpaired training datasets consist of a source dataset $\{x_i\}_{i=1}^N \in X$ and a target dataset $\{y_i\}_{i=1}^N \in Y$. A mixed training dataset consists of samples $\{x_i\}_{i=1}^N \in X \cup Y$. Most of previous methods become unworkable on the mixed training dataset. ST-GAN can learn the mapping functions from the mixed dataset.

et al. (2016) proposed some techniques to encourage the convergence of GAN, for example, using *virtual batch normalization* to replace batch normalization Ioffe and Szegedy (2015). Some researchers have tried to optimize the objective function of GAN, for example, LSGAN Mao et al. (2016) uses the least squares loss function for the discriminator to solve the vanishing gradient problem. Recently, the proposed Wasserstein GAN (WGAN) Arjovsky et al. (2017) which uses the Wasserstein distance instead of the Jensen-Shannon distance to form a new objective function, has provided a powerful theoretical proof, and experiments have illustrated that WGAN can make the GAN training process more stable. With the development and optimization of GAN, it has been applied to many fields, for example, image in-painting Pathak et al. (2016), image super-resolution Ledig et al. (2016), style transfer Li and Wand (2016), video prediction Mathieu et al. (2015) and object detection Wang et al. (2017).

One-to-one image semantic translation using a convolution neural network is a popular research topic. The most interesting work concerns style transfer, which can be considered as a problem of texture transfer. Recently, image transformation using conditional GAN has made great progress Isola et al. (2016), and the goal of Isola et al. (2016) was to develop a common framework for the image transformation problem. However, it must be based on supervised learning, that is, aligned image pairs are required in the training process. Some new methods have been proposed that use unpaired datasets as training data Liu et al. (2017); Dong et al. (2017); Zhu et al. (2017); Kim et al. (2017). If the training datasets are mixed together in one domain, these methods must divide the data into different domains using supervised labels before transformation, otherwise, these methods will not be effective. Moreover, these methods must require to be trained for each pair of domains.

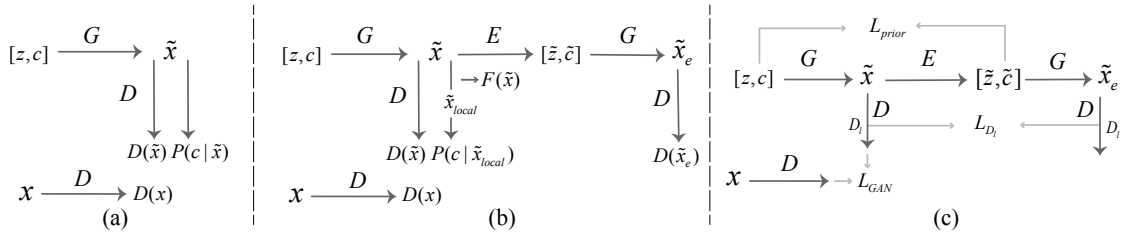


Figure 2: (a) InfoGAN architecture; (b) Overview of the ST-GAN architecture; (c) the objective function for ST-GAN training. In contrast to InfoGAN, ST-GAN consists of three networks: E, G, and D. The original input x is from the mixed dataset $X \cup Y$. We add the E network after G for reconstructing [z,c] and add $G-D$ network after E for performing L_{D_l} loss. Note that posterior probability $P(c|\tilde{x}_{local})$ is used to obtain the mutual information for local region image transformation. Probability $P(c|\tilde{x})$ is obtained if binary mask function $F(x)$ is not used. In (c), L_{prior} and L_{D_l} is for training the E network and L_{GAN} is for training the D and G networks as vanilla GAN does.

Recently, facial image semantic transformation which is based on disentangled representation have made a great progress with high visual quality. Image disentangled representations aims to represent the salient attributes of an image instance. For example, for a dataset of faces, a useful disentangled representation may allocate a separate set of dimensions for each of the following attributes: facial expression, eye color, hairstyle, or the presence or absence of eyeglasses. Most of the previous approachs have an encode-decode network that encodes the image semantics into the latent vectors and decode them into images for reconstruction. Larsen et al. (2015) proposed a new architecture called VAE/GAN that can be used for image semantic transformation and acquire labeled data to compute the visual attribute vectors after training. Another model, Adversarially Learned Inference Dumoulin et al. (2016), also needs supervised information and must be embedded with binary attributes when it is trained for the semantic transformation task. Brock et al. (2016) proposed a new architecture called Neural Photo Editing(NPE) to edit a neural photo using an unsupervised process. However, NPE's editing results are ambiguous and not natural with respect to human perception. Recently, Hadap et al. has been able to obtain very varied semantic editing results for face; it needs use 3D morphable model Blanz and Vetter (1999) for supervised training, and its image transformation looks blurry.

3. Method

3.1. GAN, InfoGAN

GAN: The goal of GAN Goodfellow et al. (2014) is to let the G network learn a distribution $P_G(x)$ that matches the real data distribution $P_{data}(x)$ via an adversarial process. GAN consists of two networks: a generative network G and discriminative network D. In most of GAN works, G and D are deep convolutional networks. The training process can be treated

as a minimax game, and the objective function of GAN is given by the following expression:

$$\min_G \max_D L(D,G) = \mathbb{E}_{x\sim P_{data}(x)}[logD(x)] + \mathbb{E}_{z\sim P_z(z)}[log(1-D(G(z)))] \tag{1}$$

InfoGAN: InfoGAN Chen et al. (2016) is an unsupervised method for disentangling representation learning by adding a mutual information term $I(c;G(z,c))$, placed between latent code c and generated sample $G(z,c)$, to the objective function of GAN. The architecture is shown in Fig. 2 (a). When maximal mutual information is attained, InfoGAN can discover highly semantic and meaningful hidden representations. For a categorical latent code c, it can model discrete variations in data, for example, presence or absence of face glasses. For a continuous latent code, it can capture continuous variations, for example, object rotation. The objective function for InfoGAN's minimax game is:

$$\min_G \max_D L_{InfoGAN}(D,G) = L(D,G) - \lambda I(c;G(z,c)) \tag{2}$$

3.2. ST-GAN

Although InfoGAN could learn the semantic correspondence between generated samples x and latent code c, InfoGAN cannot reconstruct the original images x because of the lack of an encoding network, thus, it does not have the ability to transform x from one domain into another domain. We can add an encoding network E before G to form the encode-decode architecture for reconstruction ability. If the $E-G$ network is trained using the reconstruction loss, similar to VAE/GAN Larsen et al. (2015), it is necessary to know the posterior $P(y|x)$ for label y to restrain $\tilde{c}$, which is unavailable, because we only use unlabeled data. As shown in Fig. 2 (b) we add an E network after G to form the new architecture $G-E$ for reconstructing z and c. The purpose is to make not only it able to perform reconstruction, but also enable latent code $\tilde{c}$ to capture the salient semantic variations in the data. Moreover, we augment a G network after $G-E$ to form a $G-E-G$ network for better reconstruction result.

The objective function for G and D: A recent method called WGAN-GP Gulrajani et al. (2017) minimizes an approximation of the Wasserstein distance between $P_{data}(x)$ and $P_G(x)$ instead of the previous Jensen–Shannon divergence used by the original GAN. WGAN-GP has better theoretical properties and solves most of the problems of the original GAN, for example, training instability and improvements in the quality of generated samples. The objective function of WGAN-GP has been given in Appendix A. We optimize ST-GAN using the Wasserstein distance just like the WGAN-GP model. In detail, we add a mutual information term $I(c;G(z,c))$ to the objective function of WGAN-GP to form the new objective function for the G and D network of ST-GAN. They are expressed as:

$$\min_G L_{GAN} = \mathbb{E}_{x\sim P_{data}(x)}[D(x)] - \mathbb{E}_{z\sim P_z(z),c\sim P_c(c)}[D(G(z,c))] - \lambda_2 I(c;G(z,c)) \tag{3}$$

$$\begin{aligned}\min_D L_{GAN} &= \mathbb{E}_{z\sim P_z(z),c\sim P_c(c)}[D(G(z,c))] - \mathbb{E}_{x\sim P_{data}(x)}[D(x)] \\ &+ \lambda_1\mathbb{E}(t) - \lambda_2 I(c;G(z,c)),\end{aligned} \tag{4}$$

where t is a gradient penalty variable, λ_1 and λ_2 are weighting parameters. More details about t can be found in Appendix A.

The term $I(c; G(z, c))$ requires the posterior $P(c|G(z, c))$, thus, it is hard to maximize directly. ST-GAN uses a technique called Variational Information Maximization Barber and Agakov (2003) by defining an auxiliary distribution $Q(c|x)$ to approximate $P(c|x)$ as InfoGAN Chen et al. (2016) does. More details can be found in Appendix B.

Reconstruction loss for E: Different from the previous reconstruction loss using directly element-wise similarity measure Kingma and Welling (2013) or learned similarity measure Larsen et al. (2015), we try to reconstruct z and c to indirectly reconstruct the original images. In detail, we use $E(G(z, c))$ to reconstruct z and c and call the objective function L_{prior}, as z and c are sampled from the prior distribution. After training using objective function L_{prior}, E can learn the mapping function from input images x to variables $[\tilde{z}, \tilde{c}]$, and then, we can use $G(\tilde{z}, \tilde{c})$ to reconstruct x. To improve the reconstruction ability of the proposed model, we explore a new loss to train the E network, for example, adding $L1$ distance between output images $\tilde{x}_e$ and input $\tilde{x}$ for the objective function. However Larsen et al. (2015) has found that the $L1$ pixel-wise distance is not adequate for image data, as it does not model the properties of human visual perception and causes blurring. We replace this pixel-wise loss with perception loss L_{D_l}, which computes the lth feature difference of the GAN discriminator between $\tilde{x}_e$ and $\tilde{x}$. A similar method has been used in Larsen et al. (2015). Finally, the new objective function with a hyperparameter λ_3, $L_{prior} + \lambda_3 L_{D_l}$, for E is shown as:

$$L_{prior} = \mathbb{E}_{z\sim P_z(z), c\sim P_c(c)}((z - E_z(G(z, c)))^2 + (c - E_c(G(z, c)))^2), \tag{5}$$

where E_z and E_c represent the different parts of the final output for E.

$$L_{D_l} = \mathbb{E}((D_l(\tilde{x}_e) - D_l(\tilde{x}))^2) \tag{6}$$

GST-GAN and LST-GAN: ST-GAN is based on InfoGAN, which adds a mutual information term $I(c; G(z, c))$ for the objective function to enable latent code c to capture the semantic variations in the unsupervised process. Here, we refer to the original ST-GAN architecture as global ST-GAN (GST-GAN). As mentioned before, the semantic capturing of InfoGAN is not explicit. We propose a method called local mutual information maximization for ST-GAN to make the semantic capture of c more regionally oriented and explicit by applying a binary mask function $F(\tilde{x})$ to the generated samples $\tilde{x}$. $F(\tilde{x})$, which is used to get the local semantic region x_{local}, is defined as:

$$\tilde{x}_{local} = F(\tilde{x}) = F(G(z, c)) = M \odot G(z, c), \tag{7}$$

where M denotes the binary mask of the local semantic region(e.g., eyes or the mouth) and $\odot$ denotes the element-wise product operation. Here, we refer to this ST-GAN as local ST-GAN(LST-GAN). The objective function for the G and D of LST-GAN just uses the local mutual information term $I(c; F(G(z, c)))$, replacing the term $I(c; G(z, c))$ of Eq. 3 and 4.

Semantic transformation of LST-GAN: It is essential that the global appearance is well preserved but the difference appears in the local region that we need. However, this is hard to achieve if we directly use the transformation images as the final result, because we would not reconstruct completely the original input, especially for the imbalance and

inadequate training dataset. To address this problem, we define a new mask function to obtain the global context of the original input x. Mask $F_{new}(x)$ is shown as:

$$F_{new}(x) = (1 - M) \odot x, \tag{8}$$

where M denotes the binary mask of the local semantic region that we need. Now, the new transformation result of input image x is now defined as:

$$x_{transfer} = F_{new}(x) + F(x_{origin}) = (1 - M) \odot x + M \odot x_{origin}, \tag{9}$$

where x_{origin} is the original transformation result, which is the generation of $G(E(x))$. To seamlessly blend the images and make them look realistic, Poisson blending Pérez et al. (2003) is used for the final transformation result ($x_{blending}$).

Training ST-GAN: The training process of ST-GAN(GST-GAN, LST-GAN) is as follows.

Algorithm

θ_E, θ_D, $\theta_G \leftarrow$ initialize network parameters. N is the numbers of training D.
repeat
  for $i = 1, ..., N$ do
    Sample $x \leftarrow P_{data}(x)$; $z \leftarrow P_z(z)$; $c \leftarrow P_c(c)$
    D loss $\min_D L_{GAN} \leftarrow$ Eq. 4
    // Update D parameters according to gradients.
    θ_D += $-\nabla_{\theta_D}(\min_D L_{GAN})$
  G loss $\min_G L_{GAN} \leftarrow$ Eq. 3 loss function $L_{prior} \leftarrow$ Eq. 5 perception loss $L_{D_l} \leftarrow$ Eq. 6
  // Update E and G parameters according to gradients.
  θ_E += $-\nabla_{\theta_E}(L_{prior} + \lambda_3 L_{D_l})$; θ_G += $-\nabla_{\theta_G}(\min_G L_{GAN})$
until deadline

4. Experiments

We examine our unsupervised method on the CelebA face images Liu et al. (2015) without using any label information for both training and testing. We first compare ST-GAN with VAE/GAN Larsen et al. (2015), VAE Kingma and Welling (2013), and NPE Brock et al. (2016) to evaluate the reconstruction ability of ST-GAN. Second, we demonstrate how ST-GAN transforms an input facial image into various images with different semantic features and compare them with the baseline. Then, we investigate whether LST-GAN has a more explicit semantic discovery compared with InfoGAN and demonstrate that how LST-GAN transforms input facial images into results with different semantic features of the local region. Our models were trained with Adam using a learning rate of 0.0001, and we used the Conv-Bn-Relu as the basic architecture. Hyper-parameters λ_1 and λ_2 are 10 and 1, respectively.

4.1. CelebA

The CelebFaces Attributes Dataset (CelebA) Liu et al. (2015) contains 202,599 face images with large pose variations and background clutter. We cropped and scaled the images to 64×64 pixels. We randomly selected 1200 images as a test dataset and used the remaining images as a training dataset. In this experiment, the latent vectors c consist of a 10-dimensional categorical latent vector c_1 for capturing discontinuous variations and a one-dimensional continuous vector c_2 for capturing continuous variations.

4.2. Image Reconstruction

The $G(E(x))$ network is used for image reconstruction and is the most critical step for image semantic transformation. In contrast to previous methods, the reconstruction of ST-GAN is an indirect method that reconstructs latent code c and noisy vector z. It cleverly avoids learning the mapping function in a high-dimensional manifold space by learning it in a low-dimension space. Moreover, ST-GAN uses the Wasserstein distance instead of the Jensen-Shannon divergence to optimize the objective function and training process. In our experiments, the Wasserstein distance for ST-GAN improves not only the training stability but also the quality of the generated samples (See Appendix C.).

Evaluation methods for image similarity: Peak signal-to-noise ratio Winkler and Mohandas (2008), often abbreviated as PSNR, is an expression for the ratio between the maximum possible value (power) of a signal and the power of distorting noise that affects the quality of its representation. This ratio is often used as a quality measurement between the original and a reconstructed image. The higher the PSNR, the better the quality of reconstructed image. It is defined as:

$$PSNR = 20log_{10}(\frac{255}{\sqrt{MSE}}), \tag{10}$$

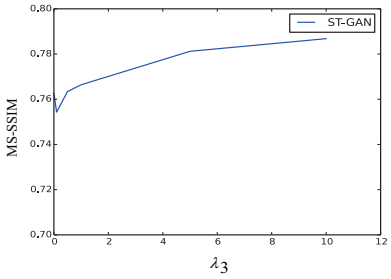
where MSE is the mean squared error of between the original image and the reconstruction image. Multi-scale structural similarity (MS-SSIM) Wang et al. (2004) is an error metric evaluating the perceived quality between the original image and reconstructed image. MS-SSIM values range between 0.0 and 1.0; higher MS-SSIM values correspond to more similar images in human perceptual.

As shown in Fig. 3(a) we use PSNR, MS-SSIM to evaluate images similarity between the real images and reconstructed results, in which ST-GAN outperforms the baseline (VAE/GAN). In Fig. 4, the reconstructed images of ST-GAN are very similar to the original images, and also much sharper than VAE Kingma and Welling (2013). Noted that the input images of NPE method contain background region, different from all other methods. Though it is not very fair comparison, ST-GAN achieves much more realistic reconstruction results for face region comparing with NPE method.

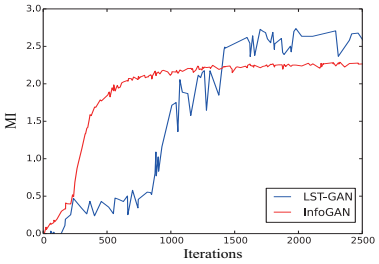
Analysis of the objective function: We also explored the different objective functions for image reconstruction. The $L1$ loss is the pixel-wise distance between reconstruction images and input images and the L_{D_l} loss can be considered as the feature-wise distance. Fig. 3 (a) shows that ST-GAN with the $L1$ loss get the top PSNR scores and higher MS-SSIM scores than when only the L_{prior} loss is used. The third rows shows L_{D_l} achieves the top MS-SSIM scores. Moreover, the MS-SSIM scores increase when hyper-parameter

Model	PSNR	MS-SSIM
VAE/GAN	28.0316	0.6737
ST-GAN	28.1057	0.7627
ST-GAN(L1)	**28.1135**	0.7743
ST-GAN(L_{D_l})	28.1003	**0.7868**

(a)



(b)



(c)

Figure 3: (a) Similarity scores for image reconstruction. The PSNR and MS-SSIM scores were measured on 1,200 test images from the CelebA dataset. The second and third rows show the evaluation scores of ST-GAN with $L_{prior} + \lambda_3 L1$ and $L_{prior} + \lambda_3 L_{D_l}$ loss ($\lambda_3 = 10$). (b) MS-SSIM scores of the images reconstruction, vs. the value of hyper-parameter λ_3. (c)We show the mutual information value of LST-GAN can be maximized just as it is maximized in InfoGAN.

Figure 4: Reconstruction obtained using different methods. ST-GAN's reconstruction samples are sharper and closer to the original images than those of VAE Kingma and Welling (2013), VAE/GAN Larsen et al. (2015), and NPE Brock et al. (2016). Hyperparameter $\lambda_3 = 10$ for ST-GAN(L_{D_l}). ST-GAN has better reconstruction result than all other methods from human perception.

λ_3 is increased (Fig. 3 (b)). We therefore conclude that both terms are useful for image reconstruction, but it is better to select L_{D_l} loss if we want to obtain a better visual effect for human perception.

4.3. Facial Semantic Transformation

By changing the value of a latent code $\tilde{c}$ keeping irrelevant latent variables and noise variables fixed, ST-GAN can transform x into various images with multiple and salient semantic features. To demonstrate this, we tested GST-GAN and LST-GAN on the CelebA 1200 test data. For GST-GAN, we chose to model the latent codes with one categorical code

Figure 5: Facial Image semantic transformation and comparison by changing $\tilde{c}_1$. (a) We show the ability of semantic discovery for GST-GAN. Input images can be transformed into results with different semantic features corresponding to different value of latent code c_1. (b) Semantic transformation comparison with a previous method (NPE). GST-GAN can transform images of faces without glasses into face images with glasses or different hairstyle when changing the value of c_1 to 4 (Left) or 6 (Right). But sometimes, the transformation does not succeed (The 5th column).

$c_1 \sim$Cat $(K = 10, p = 0.1)$ and one continuous code $c_2 \sim$Unif $(-1, 1)$. For LST-GAN, we just use c_2 to capture continuous semantic variations in mixed dataset.

4.3.1. Semantic transformation for GST-GAN

After training GST-GAN, we generated many images corresponding to various latent code values of $\tilde{c}_1$, as shown in Fig. 5 (a). We found that the generated images with the same categorical latent code $\tilde{c}_1$ have some salient semantic features. For example, $\tilde{c}_1 = 4$ and $\tilde{c}_1 = 6$ correspond to face images with glasses and hairstyles with bangs, respectively. After reconstructing input images x, we can translate face images without glasses into those with glasses by changing $\tilde{c}_1$ from another value into $\tilde{c}_1 = 4$. Conversely, we can translate face images with glasses into ones without glasses by changing $\tilde{c}_1$ from 4 to other values. In the process of transformation, the uncorrelated semantic features are basically preserved.

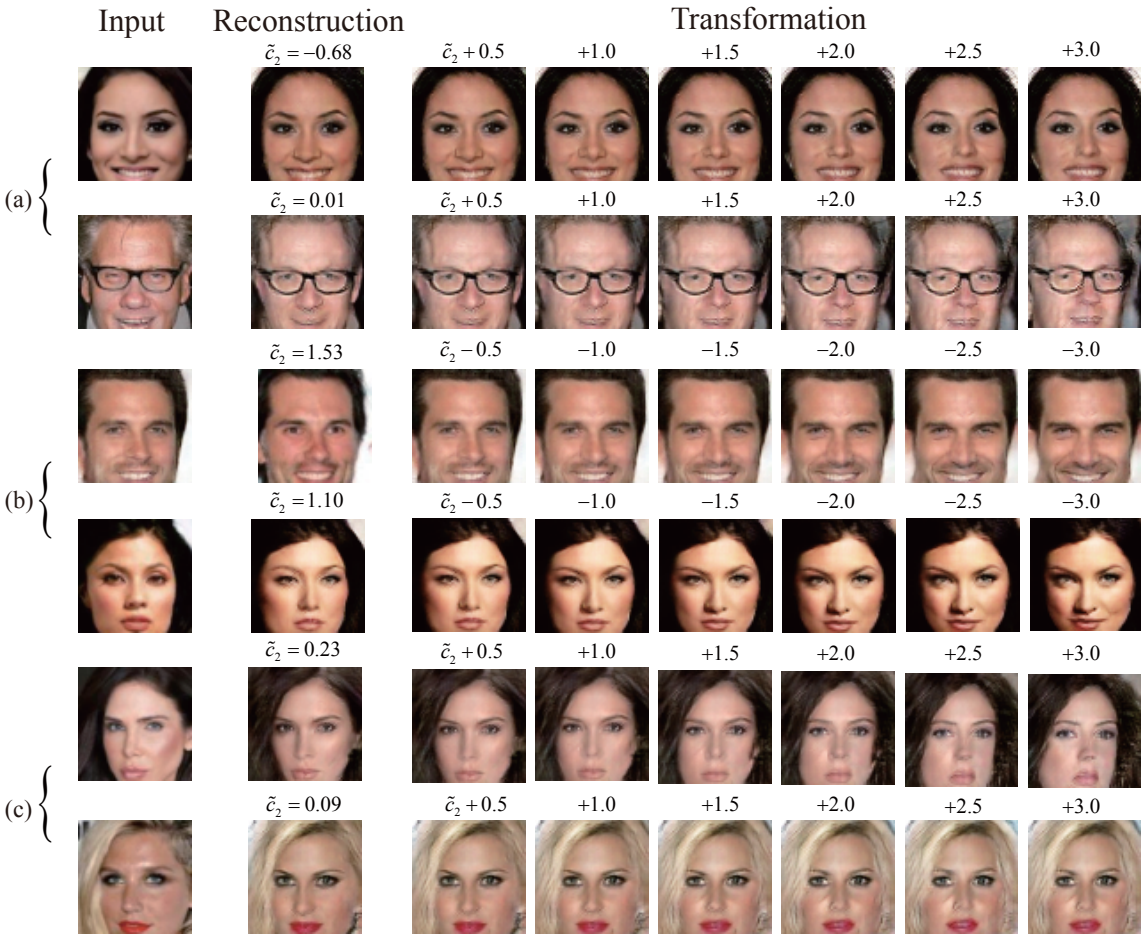


Figure 6: Facial Image semantic transformation by changing $\tilde{c}_2$. In (a), original images with a frontal face are transformed gradually into a right profile face; In (b), original images with a right profile face are transformed gradually into a frontal face; In (c), original images with a left profile face are transformed gradually into a frontal face.

However, we found that thoughout the range of $\tilde{c}_1 \in [1,\ 10]$, $\tilde{c}_1$ does not always correspond to salient semantic variation, for example, consider $\tilde{c}_1 = 1$ in the first row of Fig. 5 (a).

For continuous latent code $\tilde{c}_2$, as shown in Fig. 6, we found that it corresponds to the variations of face pose. We can translate generated samples with frontal faces into ones with a right profile face by continuously increasing its value. We can also transform generated samples with right profile faces into ones with a frontal face by continuously reducing its value.

Baseline comparison: Brock et al. Brock et al. (2016) proposed a new architecture called NPE for image semantic editing using an unsupervised training process. After training, NPE require human interaction for semantic image transformation with a "contextual paintbrush". We used their public code[1] to obtain the transformation results and then

1. Please see https://github.com/ajbrock/Neural-Photo-Editor

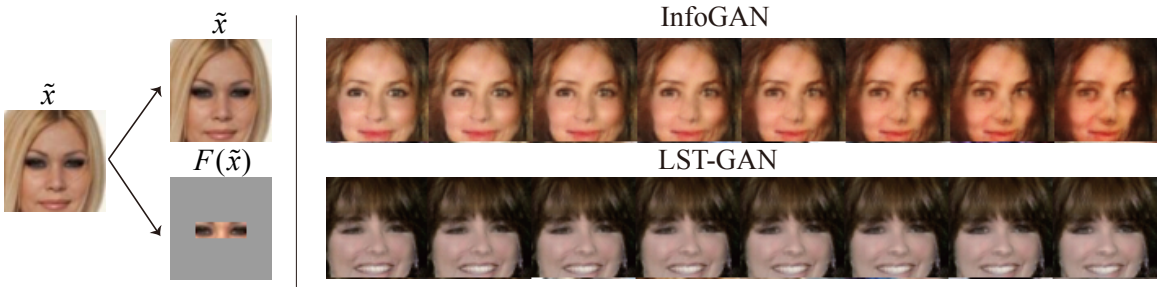


Figure 7: Comparison of semantic capture. Here, $\tilde{x}$ is generated result of $G(z, c)$; $F(\tilde{x})$ is as the input of D network for LST-GAN. After training, the images of the right is sampling from $G(z, c)$, where z is from the prior distribution and value of the continuous latent code c is changed from -1 to 1.

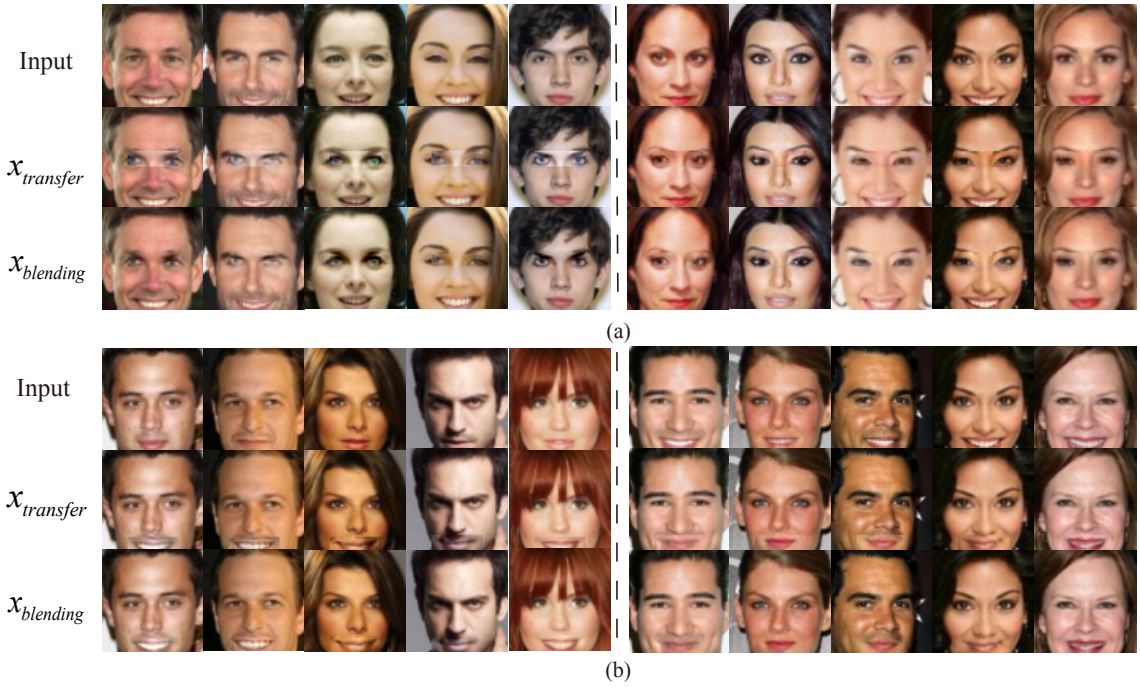


Figure 8: Facial image Semantic Transformation for local regions by changing latent code c_2. (a) We show that LST-GAN can transform the semantic features of the eye region. In the left images, The faces are transformed into faces with larger eyes. In the right images, the faces are transformed into faces with smaller eyes. (b) We show LST-GAN can transform a face between with a closed mouth into one with an open mouth. Poisson blending is used as the final transformation result $x_{blending}$ to preserving the same intensities of the surrounding pixels.

cropped the images to 64×64 for comparison. As shown in Fig. 5 (b), ST-GAN's results are more realistic and natural than those of NPE.

4.3.2. Semantic Transformation for LST-GAN

Comparison with InfoGAN: We have proposed local mutual information maximization for ST-GAN (LST-GAN) to make the semantic capturing of c more explicit and regionally oriented. InfoGAN uses the whole images x as the input to obtain mutual information $I(z; G(z, c))$. However, we do not know which semantic features have been captured before observing the results because of the unsupervised process. Moreover, we found that InfoGAN is sensitive to the most salient variations in the training data and is not no sensitive to local region semantic variations (e.g., the eye region or mouth region). As shown in the first row of Fig. 7, InfoGAN captured the variations in face pose. However, it is hard for InfoGAN to capture the eye region variations. because the variations of face pose are more conspicuous than the variations of face eye for the CelebA dataset. GST-GAN has the same weaknesses, because it is based on InfoGAN model. LST-GAN uses the local mutual information term as the objective function to optimize and constrain latent code c capturing the semantic variations in the local region. The Fig. 3(c) shows that the local mutual information $I(c; F(G(z, c)))$ can also be maximized just as $I(c; G(z, c))$ is maximized in InfoGAN. In the second row of Fig. 7, we show that LST-GAN can successfully capture the semantic variations of the eye region when using the binary mask function $F(\tilde{x})$ to obtain the eye region $\tilde{x}_{local}$ as the input for the D network.

In our experiments, we found the latent code c can capture the variations of eye size and mouth shape, which expresses the powerful semantic discovery ability of LST-GAN. As shown in Fig. 8, we leverage it to achieve various image transformations that are difficult for the previous methods, because the relevant information (eye size or mouth shape) is not labeled in the CelebA dataset. We will explore more semantics discovery tasks for image transformation in future work.

5. Conclusions

We proposed an unsupervised facial image semantic transformation method called ST-GAN. In contrast to the previous approaches which acquire images pair or data labels to divide the original dataset into source domain and target domain, ST-GAN is completely unsupervised and can utilize the semantics in the mixed training dataset, then producing natural and various transformations between the discovered semantics, just controlling the latent code. The experiments on challenging dataset demonstrate the effectiveness of the proposed approach.

Acknowledgments

This work is supported by The National Key Research, Development Program of China (No. 2016YFB1001501) and NSF of China (Nos.61672326, 61572290).

References

Martín Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. *CoRR*, abs/1701.07875, 2017. URL http://arxiv.org/abs/1701.07875.

David Barber and Felix Agakov. The im algorithm: a variational approach to information maximization. In *Proceedings of the 16th International Conference on Neural Information Processing Systems*, pages 201–208. MIT Press, 2003.

Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194. ACM Press/Addison-Wesley Publishing Co., 1999.

Andrew Brock, Theodore Lim, James M. Ritchie, and Nick Weston. Neural photo editing with introspective adversarial networks. *CoRR*, abs/1609.07093, 2016. URL http://arxiv.org/abs/1609.07093.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2172–2180, 2016.

Hao Dong, Paarth Neekhara, Chao Wu, and Yike Guo. Unsupervised image-to-image translation with generative adversarial networks. *arXiv preprint arXiv:1701.02676*, 2017.

Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Alex Lamb, Martín Arjovsky, Olivier Mastropietro, and Aaron C. Courville. Adversarially learned inference. *CoRR*, abs/1606.00704, 2016. URL http://arxiv.org/abs/1606.00704.

Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2414–2423, 2016.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028*, 2017.

Zhixin Shu1 Ersin Yumer2 Sunil Hadap, Kalyan Sunkavalli, Eli Shechtman, and Dimitris Samaras. Neural face editing with intrinsic image disentangling.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint arXiv:1611.07004*, 2016.

Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jungkwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. *arXiv preprint arXiv:1703.05192*, 2017.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2013. URL http://arxiv.org/abs/1312.6114.

Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. *arXiv preprint arXiv:1512.09300*, 2015.

Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. *arXiv preprint arXiv:1609.04802*, 2016.

Chuan Li and Michael Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *European Conference on Computer Vision*, pages 702–716. Springer, 2016.

Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised image-to-image translation networks. *arXiv preprint arXiv:1703.00848*, 2017.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3730–3738, 2015.

Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, and Zhen Wang. Least squares generative adversarial networks. *arXiv preprint ArXiv:1611.04076*, 2016.

Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *arXiv preprint arXiv:1511.05440*, 2015.

Anh Nguyen, Jason Yosinski, Yoshua Bengio, Alexey Dosovitskiy, and Jeff Clune. Plug & play generative networks: Conditional iterative generation of images in latent space. *arXiv preprint arXiv:1612.00005*, 2016.

Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016.

Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. In *ACM Transactions on graphics (TOG)*, volume 22, pages 313–318. ACM, 2003.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, abs/1511.06434, 2015. URL http://arxiv.org/abs/1511.06434.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, pages 2234–2242, 2016.

Xiaolong Wang, Abhinav Shrivastava, and Abhinav Gupta. A-fast-rcnn: Hard positive generation via adversary for object detection. *CoRR*, abs/1704.03414, 2017. URL http://arxiv.org/abs/1704.03414.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Stefan Winkler and Praveen Mohandas. The evolution of video quality measurement: From psnr to hybrid metrics. *IEEE Transactions on Broadcasting*, 54(3):660–668, 2008.

Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaolei Huang, Xiaogang Wang, and Dimitris N. Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. *CoRR*, abs/1612.03242, 2016. URL http://arxiv.org/abs/1612.03242.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. *arXiv preprint arXiv:1703.10593*, 2017.

Appendix A. The objective functions of WGAN-GP

For D network:

$$\begin{aligned} \min_D L_{GAN} &= \mathbb{E}_{z\sim P_z(z), c\sim P_c(c)}[D(G(z,c))] - \mathbb{E}_{x\sim P_{data}(x)}[D(x)] \\ &+ \lambda \mathbb{E}(t), \end{aligned} \tag{11}$$

where $t = (\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2$. Here, $\hat{x} = \epsilon x + (1-\epsilon)\tilde{x}$, where $\epsilon \sim U[0,1], x \sim P_{data}(x), \tilde{x} \sim P_G(x)$.
For G network:

$$\min_G L_{GAN} = \mathbb{E}_{x\sim P_{data}(x)}[D(x)] - \mathbb{E}_{z\sim P_z(z), c\sim P_c(c)}[D(G(z,c))] \tag{12}$$

Appendix B. Mutual information term

The mutual information term $I(c; G(z,c))$ requires the posterior $P(c|G(z,c))$, thus, it is hard to maximize directly. ST-GAN uses a technique called Variational Information Maximization Barber and Agakov (2003) by defining an auxiliary distribution $Q(c|x)$ to approximate $P(c|x)$ as InfoGAN Chen et al. (2016) does. The variational lower bound, $L_I(G,Q)$, of the local mutual information $I(c; G(z,c))$ is defined as:

$$\begin{aligned} L_I(G,Q) &= \mathbb{E}_{c\sim P(c), x\sim G(z,c)}[logQ(c|x)] + H(c) \\ &= \mathbb{E}_{x\sim G(z,c)}[\mathbb{E}_{c'\sim P(c|x)}[logQ(c^{'}|x)]] + H(c) \\ &\leq I(c; G(z,c)) \end{aligned} \tag{13}$$

ST-GAN simply adds some fully connected layers to D and the output of the final layer is regarded as the parameters of conditional distribution $Q(c|x)$. Finally, we replace the

Table 1: Inception-scores for VAE/GAN and ST-GAN, evaluated on 12800 images. the variances of these scores are very small value, thus they has strong credibility.

Method	Inception Score
VAE/GAN Larsen et al. (2015)	2.80±0.057
ST-GAN	**2.86±0.04**
CelebA dataset	3.06±0.08

term $I(c; G(z, c))$ with $L_I(G, Q)$ for the objective function of ST-GAN, thus, the practical objective functions for G and D of ST-GAN is:

$$\begin{aligned} \min_G L_{GAN} &= \mathbb{E}_{x\sim P_{data}(x)}[D(x)] - \mathbb{E}_{z\sim P_z(z), c\sim P_c(c)}[D(G(z,c))] \\ &- \lambda_2 L_I(G, Q) \end{aligned} \tag{14}$$

$$\begin{aligned} \min_D L_{GAN} &= \mathbb{E}_{z\sim P_z(z), c\sim P_c(c)}[D(G(z,c))] - \mathbb{E}_{x\sim P_{data}(x)}[D(x)] \\ &+ \lambda_1 \mathbb{E}(t) - \lambda_2 L_I(G, Q) \end{aligned} \tag{15}$$

The practical objective functions for LST-GAN are the same as ST-GAN, except replacing $Q(c|x)$ of $L_I(G, Q)$ with $Q(c|\tilde{x}_{local})$, where $\tilde{x}_{local} = F(G(z, c))$.

Appendix C. Assessment of image quality

In this experiment, we compared generated samples quality of ST-GAN with VAE/GAN Larsen et al. (2015). We trained separately every method on the CelebA training dataset and used Inception score Salimans et al. (2016) to evaluate the sample quality in 12800 images. The comparison results are shown in Table 1 and the Inception score of the CelebA dataset is performed in the last row of Table 1. Comparisons show ST-GAN get better generated results than VAE/GAN.