PSGAN: Pose and Expression Robust Spatial-Aware GAN for Customizable Makeup Transfer

Wentao Jiang [1] Si Liu [1*] Chen Gao [2,4] Jie Cao [3,4] Ran He [3,4] Jiashi Feng [5] Shuicheng Yan [6]
[1] School of Computer Science and Engineering, Beihang University
[2] Institute of Information Engineering, Chinese Academy of Sciences
[3] Institute of Automation, Chinese Academy of Sciences
[4] University of Chinese Academy of Sciences [5] National University of Singapore [6] YITU Tech
{jiangwentao, liusi}@buaa.edu.cn, gaochen@iie.ac.cn

Figure 1. Our model allows users to control both the shade of makeup and facial parts to transfer. The first row on the left shows the results of only transferring partial makeup style from the reference. The second row shows the results with controllable shades. Moreover, our method can perform makeup transfer between images that have different poses and expressions, as shown on the right part of the figure. Best viewed in color.

Abstract

In this paper, we address the makeup transfer task, which aims to transfer the makeup from a reference image to a source image. Existing methods have achieved promising progress in constrained scenarios, but transferring between images with large pose and expression differences is still challenging. Besides, they cannot realize customizable transfer that allows a controllable shade of makeup or specifies the part to transfer, which limits their applications. To address these issues, we propose Pose and expression robust Spatial-aware GAN (PSGAN). It first utilizes Makeup Distill Network to disentangle the makeup of the reference image as two spatial-aware makeup matrices. Then, Attentive Makeup Morphing module is introduced to specify how the makeup of a pixel in the source image is morphed from the reference image. With the makeup matrices and the source image, Makeup Apply Network is used to perform makeup transfer. Our PSGAN not only achieves state-of-the-art results even when large pose and expression differences exist but also is able to perform partial and shade-controllable makeup transfer. Both the code and a newly collected dataset containing facial images with various poses and expressions will be available at https://github.com/wtjiang98/PSGAN.

*Corresponding author

1. Introduction

We explore the makeup transfer task, which aims to transfer the makeup from an arbitrary reference image to a source image. It is widely demanded in many popular portrait beautifying applications. Most existing makeup transfer methods [18, 3, 2, 11] are based on Generative Adversarial Networks (GANs) [10]. They generally use face parsing maps and/or facial landmarks as a preprocessing step to facilitate the subsequent processing and adopt the framework of CycleGAN [33] which is trained on unpaired sets of images, i.e., non-makeup images and with-makeup images.

However, existing methods mainly have two limitations. Firstly, they only work well on frontal facial images with neutral expression since they lack a specially designed module to handle the misalignment of images and overfit on

frontal images. While in practical applications, an ideal makeup transfer method should be *pose and expression robust*, which is able to generate high-quality results even if source images and reference images show different poses and expressions. Secondly, the existing methods cannot perform customizable makeup transfer since they encode makeup styles into low dimension vectors which lose the spatial information. An ideal makeup transfer method need to realize *partial* and *shade-controllable* makeup transfer. Partial transfer indicates transferring the makeup of specified facial regions separately, e.g., eye shadows or lipstick. Shade-controllable transfer means the shade of the transferred makeup can be controllable from light to heavy.

To solve these challenges, we propose a novel Pose and expression robust Spatial-aware GAN (PSGAN), which consists of a Makeup Distill Network (MDNet), an Attentive Makeup Morphing (AMM) module and a Makeup Apply Network (MANet). Different from the previous approaches that simply input two images into the network or recombine makeup latent code and identity latent code to perform transfer, we design PSGAN to transfer makeup through scaling and shifting the feature map for only once, inspired by style transfer methods [14, 5]. Comparing with general style transfer, makeup transfer is more difficult since the human perception system is very sensitive to the artifacts on faces. Also, makeup styles contain subtle details in each facial region instead of general styles. To this end, we propose MDNet to disentangle the makeup from the reference image into two makeup matrices, i.e., the coefficient matrix γ and bias matrix β which both have the same spatial dimensions with visual features. These matrices embed the makeup information and serve as the shifting and scaling parameters. Then, γ and β are morphed and adapted to the source image by the AMM module which calculates an attentive matrix A to produce adapted makeup matrices γ' and β'. The AMM module utilizes the face parsing maps and facial landmarks to build the pixel-wise correspondences between source images and reference images, which solves the misalignment of faces. Finally, the proposed MANet conducts makeup transfer through applying pixel-wise multiplication and addition on visual features using γ' and β'.

Since the makeup style has been distilled in a spatial-aware way, *partial transfer* can be realized by applying masks pixel-wisely according to the face parsing results. For example, in the top left panel of Figure 1, the lip gloss, skin and eye shadow can be individually transferred from the reference image to the source image. *Shade-controllable transfer* can be realized through multiplying the weights of makeup matrices by coefficient within $[0, 1]$. As shown in the bottom left panel of Figure 1, where the makeup shade is increasingly heavier. Moreover, the novel AMM module effectively assists the generating of *pose and expression robust* results, as shown in the right part of Figure 1. We also directly apply transfer to every frame of facial videos and still get nice and consistent results. With the three novel components, PSGAN satisfies the requirements we pose for an ideal customizable makeup transfer method.

We make the following contributions in this paper:

- To our best knowledge, PSGAN is the first to simultaneously realize partial, shade-controllable, and pose/expression robust makeup transfer, which facilitates the applications in the real-world environment.
- A MDNet is introduced to disentangle the makeup from the reference image as two makeup matrices. The spatial-aware makeup matrices enable the flexible partial and shade-controllable transfer.
- An AMM module that adaptively morphs the makeup matrices to source images is proposed, which enables pose and expression robust transfer.
- A new Makeup-Wild dataset containing images with diverse poses and expressions is collected for better evaluations.

2. Related Work

2.1. Makeup Transfer

Makeup transfer has been studied a lot these years [27, 12, 17, 21, 20, 1]. BeautyGAN [18] first proposed a GAN framework with dual input and output for makeup transfer and removal simultaneously. They also introduced a makeup loss that matches the color histogram in different parts of faces for instance-level makeup transfer. BeautyGlow [3] proposed a similar idea on the Glow framework and decomposed makeup component and non-makeup component. PairedCycleGAN [2] employed an additional discriminator to guide makeup transfer using pseudo transferred images generated by warping the reference face to the source face. LADN [11] leveraged additional multiple overlapping local discriminators for dramatic makeup transfer. However, the above approaches often fail on transferring in-the-wild images and cannot adjust transfer precisely and partially, which limits their applications, such as the makeup transfer in videos.

2.2. Style Transfer

Style transfer has been investigated extensively [8, 7, 15, 22, 26]. [9] proposed to derive image representations from CNN, which can be separated and recombined to synthesize images. Some methods are developed to solve the fast style transfer problem. [5] found the vital role of normalization in style transfer networks and achieved fast style transfer by the conditional instance normalization. While their methods can only transfer a fixed set of styles and cannot adapt to arbitrary new styles. Then, [14] proposed adaptive instance

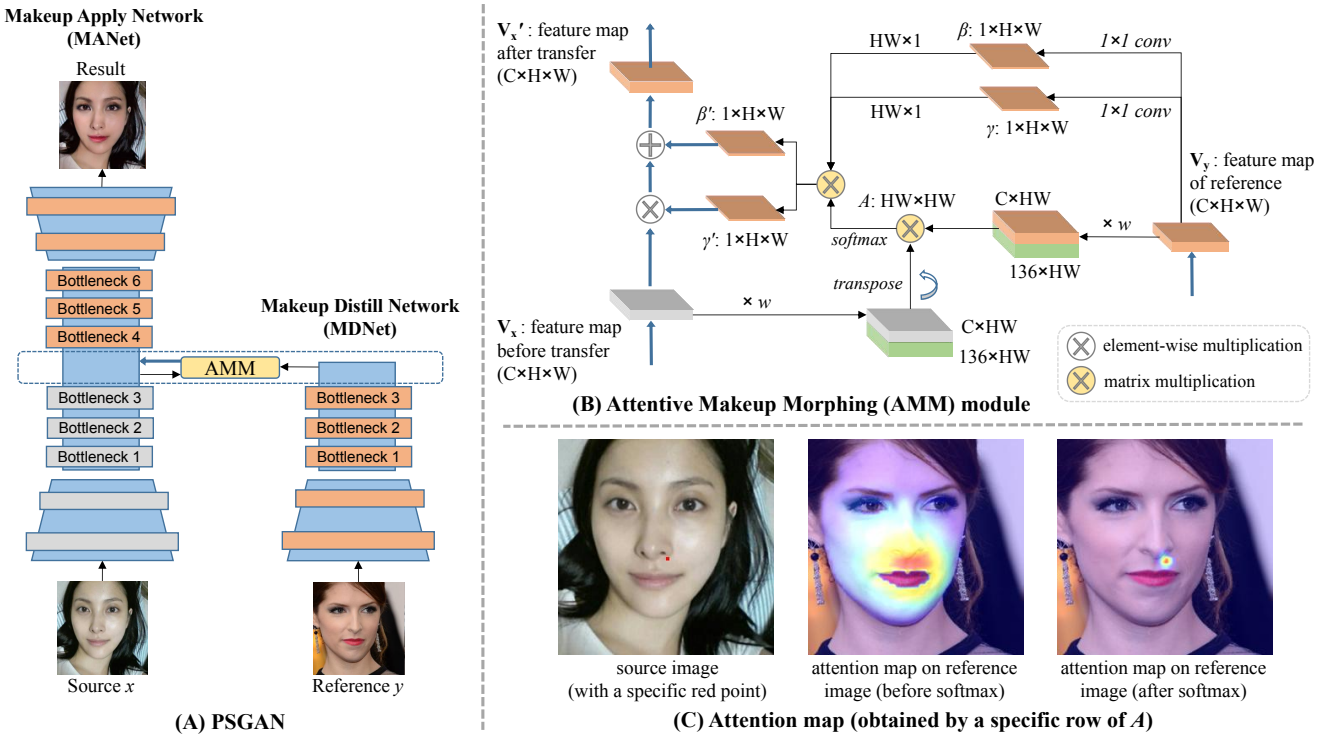


Figure 2. **(A)** Illustration of PSGAN framework. MDNet distills makeup matrices from the reference image. AMM module applies the adapted makeup matrices to the output feature map of the third bottleneck of MANet to achieve makeup transfer. **(B)** Illustration of AMM module. Green blocks with 136 (68×2) channels indicate relative position features of the pixels, which are then concatenated with C-channel visual features. Thus, the attention map is computed for each pixel in the source image through the similarity of relative positions and visual appearances. The adapted makeup matrices γ' and β' are produced by the AMM module, which are then multiplied and added to feature maps of MANet element-wisely. The orange and the gray blocks in the figure indicate visual features with makeup and without makeup. **(C)** Attention maps for a specific red point in the source image. Note that we only calculate attentive values for pixels that belong to the same facial region. Thus, there are no response values on the lip and eye of the reference image.

normalization (AdaIN) that aligns the mean and variance of the content features with those of the style features and achieved arbitrary style transfer. Here, we propose spatial-aware makeup transfer for each pixel rather than transferring a general style from the reference inspired by [24].

2.3. Attention Mechanism

Attention mechanism has been utilized in many areas [30, 23, 13, 25, 6]. [28] proposed the attention mechanism in the natural language processing area by leveraging a self-attention module to compute the response at a position in a sequence by attending to all positions and taking their weighted average in an embedding space. [29] proposed the non-local network, which is to compute the response at a position as a weighted sum of the features at all positions. Inspired by these works, we explore the application of attention module by calculating the attention between two feature maps. Unlike the non-local network that only considers visual appearance similarities, our proposed AMM module computes the weighted sum of another feature map by considering both visual appearances and locations.

3. PSGAN

3.1. Formulation

Let X and Y be the source image domain and the reference image domain. Also, we utilize $\{x^n\}_{n=1,\dots,N}$, $x^n \in X$ and $\{y^m\}_{m=1,\dots,M}$, $y^m \in Y$ to represent the examples of two domains respectively. Note that paired datasets are not required. That is, the source and reference images have different identities. We assume x is sampled from X according to the distribution $\mathcal{P}_X$ and y is sampled from Y according to the distribution $\mathcal{P}_Y$. Our proposed PSGAN learns a transfer function $G : \{x, y\} \rightarrow \tilde{x}$, where the transferred image $\tilde{x}$ has the makeup style of the reference image y and preserves the identity of the source image x.

3.2. Framework

Overall. The framework of PSGAN is shown in Figure 2 (A). Mathematically, it is formulated as $\tilde{x} = G(x, y)$. It can be divided into three parts. 1) *Makeup distill network.* MDNet extracts the makeup style from the reference image y and represents it as two makeup matrices γ and β, which have the same height and width as the feature map. 2) *At-*

tentive makeup morphing module. Since source images and reference images may have large discrepancies in expressions and poses, the extracted makeup matrices cannot be directly applied to the source image x. We then propose an AMM module to morph the two makeup matrices to two new matrices γ' and β' which are adaptive to the source image by considering the similarities between pixels of the source and reference. 3) *Makeup apply network.* The adaptive makeup matrices γ' and β' are applied to the bottleneck of the MANet to perform makeup transfer with pixel-level guidance by element-wise multiplication and addition.

Makeup distill network. The MDNet utilizes the encoder-bottleneck architecture used in [4] without the decoder part. It disentangles the makeup related features, e.g., lip gloss, eye shadows, from the intrinsic facial features, e.g., facial shape, the size of eyes. The makeup related features are represented as two makeup matrices γ and β, which are used to transfer the makeup by pixel-level operations. As shown in Figure 2 (B), the output feature map of MDNet $\mathbf{V_y} \in \mathbb{R}^{C\times H\times W}$ is fed into two 1×1 convolution layers to produce $\gamma \in \mathbb{R}^{1\times H\times W}$ and $\beta \in \mathbb{R}^{1\times H\times W}$, where C, H and W are the number of channels, height and width of the feature map.

Attentive makeup morphing module. Since the source and reference images may have different poses and expressions, the obtained spatial-aware γ and β cannot be applied directly to the source image. The proposed AMM module calculates an attentive matrix $A \in \mathbb{R}^{HW\times HW}$ to specify how a pixel in the source image x is morphed from the pixels in the reference image y, where $A_{i,j}$ indicates the attentive value between the i-th pixel x_i in image x and the j-th pixel y_j in image y.

Intuitively, makeup should be transferred between the pixels with similar relative positions on the face, and the attentive values between these pixels should be high. For example, the lip gloss region of the transferred result $\tilde{x}$ should be sampled from the corresponding lip gloss region of the reference image y. To describe the relative positions, we take the facial landmarks as anchor points. The relative position feature of pixel x_i is represented by $\mathbf{p}_i \in \mathbb{R}^{136}$, which is reflected in the differences of coordinates between pixel x_i and 68 facial landmarks, calculated by

$$\begin{aligned}\mathbf{p}_i = [&f(x_i)-f(l_1), f(x_i)-f(l_2), \ldots, f(x_i)-f(l_{68}),\\ &g(x_i)-g(l_1), g(x_i)-g(l_2), \ldots, g(x_i)-g(l_{68})],\end{aligned} \tag{1}$$

where $f(\cdot)$ and $g(\cdot)$ indicate the coordinates on x and y axes, l_i indicates the i-th facial landmark obtained by the 2D facial landmark detector [31], which serves as the anchor point when calculating $\mathbf{p}_i$. In order to handle faces that occupy different sizes in images, we divide $\mathbf{p}$ by its two-norm (i.e., $\frac{\mathbf{p}}{\|\mathbf{p}\|}$) when calculating the attentive matrix.

Moreover, to avoid unreasonable sampling pixels with similar relative positions but different semantics, we also consider the visual similarities between pixels (e.g., x_i and y_j), which are denoted as the similarities between $\mathbf{v}_i$ and $\mathbf{v}_j$ that extracted from the third bottleneck of MANet and MDNet respectively. To make the relative position to be the primary concern, we multiply the visual features by a weight when calculating A. Then, the relative position features are resized and concatenated with the visual features along the channel dimension. As Figure 2 (B) shows, the attentive value $A_{i,j}$ is computed by considering the similarities of both visual appearances and relative positions via

$$A_{i,j} = \frac{\exp\left([w\mathbf{v}_i, \frac{\mathbf{p}_i}{\|\mathbf{p}_i\|}]^T [w\mathbf{v}_j, \frac{\mathbf{p}_j}{\|\mathbf{p}_j\|}]\right) \mathbb{I}(m_x^i = m_y^j)}{\sum_j \exp\left([w\mathbf{v}_i, \frac{\mathbf{p}_i}{\|\mathbf{p}_i\|}]^T [w\mathbf{v}_j, \frac{\mathbf{p}_j}{\|\mathbf{p}_j\|}]\right) \mathbb{I}(m_x^i = m_y^j)}, \tag{2}$$

where $[\cdot,\cdot]$ denotes the concatenation operation, $\mathbf{v} \in \mathbb{R}^C$ and $\mathbf{p} \in \mathbb{R}^{136}$ indicate the visual features and relative position features, w is the weight for visual features. $\mathbb{I}(\cdot)$ is an indicator function whose value is 1 if the inside formula is true, $m_x, m_y \in \{0, 1, \ldots, N-1\}^{H\times W}$ are the face parsing map of source image x and reference image y, where N stands for the number of facial regions (N is 3 in our experiments including eyes, lip and skin), m_x^i and m_y^j indicate the facial regions that x_i and x_j belong to. Note that we only consider the pixels belonging to same facial region, i.e., $m_x^i = m_y^j$, by applying indicator function $\mathbb{I}(\cdot)$.

Given a specific point that marked in red in the lower-left corner of the nose in the source image, the middle image of Figure 2 (C) shows its attention map by reshaping a specific row of the attentive matrix $A_{i,:} \in \mathbb{R}^{1\times HW}$ to $H \times W$. We can see that only the pixels around the left corner of the nose have large values. After applying softmax, attentive values become more gathered. This verifies that our proposed AMM module is able to locate semantically similar pixels to attend.

We multiply attentive matrix A by the γ and β, and get the morphed makeup matrices γ' and β'. More specifically, the matrices γ' and β' are computed by

$$\begin{aligned}\gamma'_i &= \sum_j A_{i,j}\gamma_j;\\ \beta'_i &= \sum_j A_{i,j}\beta_j,\end{aligned} \tag{3}$$

where i and j are the pixel index of x and y. After that, the matrix $\gamma' \in \mathbb{R}^{1\times H\times W}$ and $\beta' \in \mathbb{R}^{1\times H\times W}$ are duplicated and expanded along the channel dimension to produce the makeup tensors $\Gamma' \in \mathbb{R}^{C\times H\times W}$ and $B' \in \mathbb{R}^{C\times H\times W}$, which will be the input of MANet.

Makeup apply network. MANet utilizes a similar encoder-bottleneck-decoder architecture as [4]. As shown in Figure 2 (A), the encoder part of MANet shares the

same architecture with MDNet, but they do not share parameters. In the encoder part, we use instance normalizations that have no affine parameters that make the feature map to be a normal distribution. In the bottleneck part, the morphed makeup tensors Γ' and B' obtained by the AMM module are applied to the source image feature map $\mathbf{V_x} \in \mathbb{R}^{C\times H\times W}$. The activation values of the transferred feature map $\mathbf{V_x}'$ are calculated by

$$\mathbf{V_x}' = \Gamma' \mathbf{V_x} + B'. \tag{4}$$

Eq. (4) gives the function of makeup transfer. The updated feature map $\mathbf{V_x}'$ is then fed to the subsequent decoder part of MANet to produce the transferred result.

3.3. Objective Function

Adversarial loss. We utilize two discriminators D_X and D_Y for the source image domain X and the reference image domain Y, which try to discriminate between generated images and real images and thus help the generators synthesize realistic outputs. Therefore, the adversarial loss L_D^{adv}, L_G^{adv} for discriminator and generator are computed by

$$\begin{aligned} L_D^{adv} = &-\mathbb{E}_{x\sim\mathcal{P}_X}\left[\log D_X(x)\right] - \mathbb{E}_{y\sim\mathcal{P}_Y}\left[\log D_Y(y)\right] \\ &- \mathbb{E}_{x\sim\mathcal{P}_X, y\sim\mathcal{P}_Y}\left[\log\left(1 - D_X(G(y,x))\right)\right] \\ &- \mathbb{E}_{x\sim\mathcal{P}_X, y\sim\mathcal{P}_Y}\left[\log\left(1 - D_Y(G(x,y))\right)\right] \\ L_G^{adv} = &- \mathbb{E}_{x\sim\mathcal{P}_X, y\sim\mathcal{P}_Y}\left[\log\left(D_X(G(y,x))\right)\right] \\ &- \mathbb{E}_{x\sim\mathcal{P}_X, y\sim\mathcal{P}_Y}\left[\log\left(D_Y(G(x,y))\right)\right] \end{aligned} \tag{5}$$

Cycle consistency loss. Due to the lack of triplets data (source image, reference image, and transferred image), we train the network in an unsupervised way. Here, we introduce the cycle consistency loss proposed by [33]. We use the L1 loss to constrain the reconstructed images and define the cycle consistency loss L_G^{cyc} as

$$\begin{aligned} L_G^{cyc} = &\ \mathbb{E}_{x\sim\mathcal{P}_X, y\sim\mathcal{P}_Y}\left[\|G(G(x,y),x) - x\|_1\right] \\ &+ \mathbb{E}_{x\sim\mathcal{P}_X, y\sim\mathcal{P}_Y}\left[\|G(G(y,x),y) - y\|_1\right]. \end{aligned} \tag{6}$$

Perceptual loss. When transferring the makeup style, the transferred image is required to preserve personal identity. Instead of directly measuring differences at pixel-level, we utilize a VGG-16 model pre-trained on ImageNet to compare the activations of source images and generated images in the hidden layer. Let $F_l(\cdot)$ denote the output of the l-th layer of VGG-16 model. We introduce the perceptual loss L_G^{per} to measure their differences using L2 loss:

$$\begin{aligned} L_G^{per} = &\ \mathbb{E}_{x\sim\mathcal{P}_X, y\sim\mathcal{P}_Y}\left[\|F_l(G(x,y)) - F_l(x)\|_2\right] \\ &+ \mathbb{E}_{x\sim\mathcal{P}_X, y\sim\mathcal{P}_Y}\left[\|F_l(G(y,x)) - F_l(y)\|_2\right]. \end{aligned} \tag{7}$$

Makeup loss. To provide coarse guidance for makeup transfer, we utilize the makeup loss proposed by [18]. Specifically, we perform histogram matching on the same facial regions of x and y separately and then recombine the results, denoted as $HM(x,y)$. As a kind of pseudo ground truth, $HM(x,y)$ preserves the identity of x and has a similar color distribution with y. Then we calculate the makeup loss L_G^{make} as coarse guidance by

$$\begin{aligned} L_G^{make} = &\ \mathbb{E}_{x\sim\mathcal{P}_X, y\sim\mathcal{P}_Y}\left[\|G(x,y) - HW(x,y)\|_2\right] \\ &+ \mathbb{E}_{x\sim\mathcal{P}_X, y\sim\mathcal{P}_Y}\left[\|G(y,x) - HW(y,x)\|_2\right]. \end{aligned} \tag{8}$$

Total loss. The loss L_D and L_G for discriminator and generator of our approach can be expressed as

$$\begin{aligned} L_D &= \lambda_{adv} L_D^{adv} \\ L_G &= \lambda_{adv} L_G^{adv} + \lambda_{cyc} L_G^{cyc} + \lambda_{per} L_G^{per} + \lambda_{make} L_G^{make}, \end{aligned} \tag{9}$$

where λ_{adv}, λ_{cyc}, λ_{per}, λ_{make} are the weights to balance the multiple objectives.

4. Experiments

4.1. Data Collection

Since the existing makeup datasets only consist of frontal facial images with neutral expressions, we collect a new Makeup-Wild dataset that contains facial images with various poses and expressions as well as complex backgrounds to test methods in the real-world environment. We collect data from the Internet and then manually remove images with frontal face or neutral expression. After that, we crop and resize the images to be 256×256 resolution. Finally, 403 with-makeup images and 369 non-makeup images are collected to form the Makeup-Wild dataset.

4.2. Experimental Setting and Details

We train our network using the training part of the MT (Makeup Transfer) dataset [18, 3] and test it on the testing part of MT dataset and the Makeup-Wild dataset. MT dataset contains $1,115$ non-makeup images and $2,719$ with-makeup images which are mostly well-aligned, with the resolution of 361×361 and the corresponding face parsing results. We follow the splitting strategy of [18] to form the train/test set and conduct frontal face experiments in the test set of MT dataset since the examples in the test set are well-aligned frontal facial images. To further prove the effectiveness of PSGAN for handling pose and expression differences, we use the Makeup-Wild dataset as an extra test set. Note that we only train our network using the training part of the MT dataset for a fair comparison.

For all experiments, we resize the images to 256×256, and utilize the $relu\_4\_1$ feature layer of VGG-16 for calculating perceptual loss. The weights of different loss functions are set as $\lambda_{adv} = 1$, $\lambda_{cyc} = 10$, $\lambda_{per} = 0.005$, $\lambda_{make} = 1$, and the weight for visual feature in AMM is

Source Reference w.o. AMM PSGAN

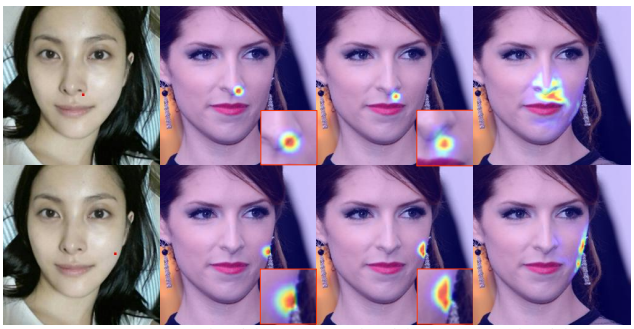
Figure 3. Without AMM module, the makeup transfer results (the 3rd column) are bad due to pose and expression differences between source and reference images.

set to 0.01. We train the model for 50 epochs optimized by Adam [16] with learning rate of 0.0002 and batch size of 1.

4.3. Ablation Studies

Attentive makeup morphing module. In PSGAN, AMM module morphs the distilled makeup matrices γ and β to γ', β'. It alleviates the pose and expression differences between source and reference images. The effectiveness of the AMM module is shown in Figure 3. In the first row, the pose of source and reference images are very different. The bangs of the reference image are transferred to the skin of the source image without AMM. By applying AMM, the pose misalignment is well solved. A similar observation can be found in the second row: the expressions of source and reference images are smiling and neutral respectively, while the lip gloss is applied to the teeth region without the AMM module shown in the third column. After integrating AMM, lip gloss is applied to the lip region, bypassing the teeth area. The experiments demonstrate that the AMM module can specify how a pixel in the source image is morphed from pixels of the reference instead of mapping the makeup from the same location directly.

The weight of visual feature in calculating A. In the AMM module, we calculate the attentive matrix A by considering both the visual features $\mathbf{v}$ and relative positions $\mathbf{p}$ using Eq. (2). Figure 4 demonstrates that if only relative positions are considered by setting the weight to zero, the attentive maps in the second column are similar to a 2D Gaussian distribution. In the first column of Figure 4, the red point on the skin of the source may wrongly receive makeup from the nostrils area in the reference image (1st row). The attention map also crosses the face boundary and covers the earrings (2nd row) which is unreasonable. Besides, larger weights will lead to scattered and unreasonable attention maps, as shown in the last column. After considering the appearance feature appropriately by setting the weight to 0.01, the attention maps focus more on the skin and also bypass the nostrils as well as background.



Source (with a red point) w.o. visual features ($w = 0$) w. visual features ($w = 0.01$) w. visual features ($w = 0.05$)

Figure 4. Given a red point on the skin, the corresponding attention maps with different weights on visual features are shown. Without using visual features, attention maps fail to avoid nostrils (1st row, 2nd column) and wrongly crosses the facial boundary (2nd row, 2nd column). While a larger weight leads to scattered and unreasonable attention maps.

Reference 1 Source lip from R1 other from R2 Reference 2

Figure 5. Given the source image (2nd column), the transferred images (3rd column) are generated by transferring the lipstick from reference 1 and other makeup from reference 2.

4.4. Partial and Interpolated Makeup Transfer

Since the makeup matrices γ and β are spatial-aware, the partial and interpolated transfer can be realized during testing. To achieve partial makeup generation, we compute the new makeup matrices by weighting the matrices using the face parsing results. Let x, y_1, and y_2 denote a source image and two reference images. We can obtain Γ'_x, B'_x and Γ'_{y_1}, B'_{y_1} as well as Γ'_{y_2}, B'_{y_2} by feeding the images to MDNet. In addition, we can obtain the face parsing mask m_x of x through the existing deep learning method [32]. Suppose we want to mix the lipstick from y_1 and other makeup from y_2, we can first obtain the binary mask of the lip, denoted as $m^l_x \in \{0,1\}^{H\times W}$. Then, PSGAN can realize partial makeup transfer by assigning different makeup parameters on different pixels. By modifying Eq. (4), the partial transferred feature map $\mathbf{V_x}'$ can be calculated by

$$\mathbf{V_x}' = (m^l_x\Gamma'_{y_1}+(1-m^l_x)\Gamma'_{y_2})\mathbf{V_x}+(m^l_xB'_{y_1}+(1-m^l_x)B'_{y_2}). \quad (10)$$

Figure 5 shows the results by mixing the makeup styles from two references partially. The results on the third column recombine the makeup of lip from reference 1 and other part of makeup from reference 2, which are natural and realistic. Also, only transferring the lipstick from reference 1 and remain other parts unchanged can be achieved

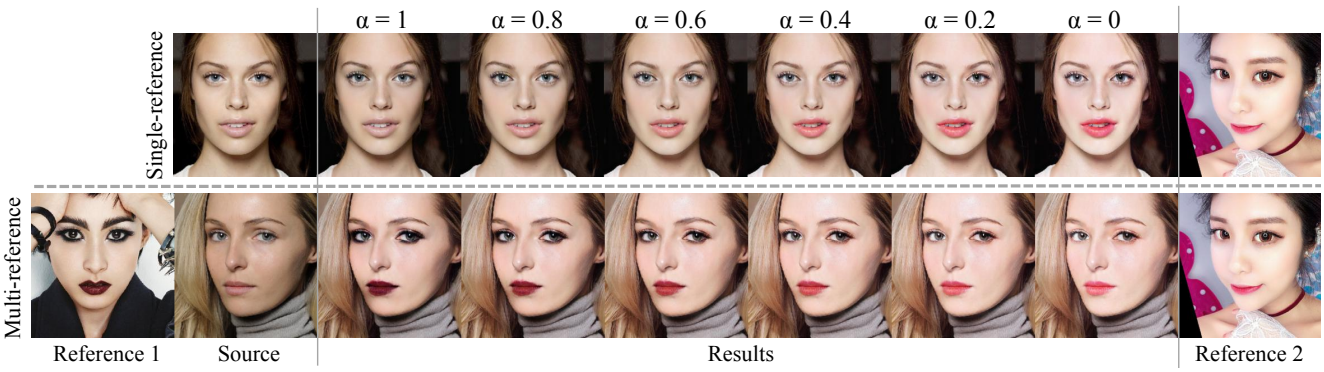


Figure 6. Results of interpolated makeup styles. If only one reference is used, adjusting the shade of makeup can be realized (1st row). If two references are used (1st column and last column), the makeup of the transferred images is gradually changing from reference 1 towards reference 2 from left to right (2nd rows).

by assigning $x = y_2$. The new feature of partial makeup makes PSGAN realize the flexible partial makeup transfer.

Moreover, we can interpolate the makeup with two reference images by a coefficient $\alpha \in [0, 1]$. We first get the makeup tensors of two references y_1 and y_2, and then compute the new parameters by weighting them with the coefficient α. The resulted feature map $\mathbf{V_x}'$ is calculated by

$$\mathbf{V_x}' = (\alpha\Gamma'_{y_1} + (1-\alpha)\Gamma'_{y_2})\mathbf{V_x} + (\alpha B'_{y_1} + (1-\alpha)B'_{y_2}). \quad (11)$$

Figure 6 shows the interpolated makeup transfer results with one and two reference images. By feeding the new makeup tensors into MANet, we yield a smooth transition between two reference makeup styles. Similarly, we can adjust the shade of transfer using only one reference image by assigning $x = y_1$. The generated results demonstrate that our PSGAN can not only control the shade of makeup transfer but also generate a new style of makeup by mixing the makeup tensors of two makeup styles.

We can also perform partial and interpolated transfer simultaneously by leveraging both the face parsing maps and coefficient thanks to the design of spatial-aware makeup matrices. The above experiments have demonstrated that PSGAN broadens the application range of makeup transfer significantly.

4.5. Comparison

We conduct comparison with general image-to-image translation methos DIA [19] and CycleGAN [33] as well as state-of-the-art makeup transfer methods BeautyGAN (BGAN) [18], PairedCycleGAN (PGAN) [2], BeautyGlow (BGlow) [3] and LADN [11]. Current makeup transfer methods leverage face parsing maps [2, 3, 18] and facial landmarks [11] for training and realize different functions as shown in Table 1.

Quantitative Comparison. We conduct a user study for quantitative evaluation on Amazon Mechanical Turk that use BGAN, CGAN, DIA, and LADN as baselines. For a fair comparison, we only compare with methods whose code and pre-train model are released since we cannot guarantee a perfect re-implementation. We randomly select 20 source images and 20 reference images from both the MT test set and Makeup-Wild (M-Wild) dataset. After using the above methods to perform makeup transfer between these images, we obtain 800 images for each method. Workers are asked to choose the best images generated by five methods by considering image realism and the similarity with reference makeup styles. Table 2 shows the human evaluation results. Our PSGAN outperforms other methods by a large margin, especially on the M-Wild test set.

Method	shade-controllable	partial	robust
BGAN [18]			
PGAN [2]			
BGlow [3]	✓		
LADN [11]	✓		
PSGAN	✓	✓	✓

Table 1. Analysises of existing methods. "robust" indicates pose and expression robust transfer.

Test set	PSGAN	BGAN	DIA	CGAN	LADN
MT	**61.5**	32.5	3.25	2.5	0.25
M-Wild	**83.5**	13.5	1.75	1.25	0.0

Table 2. Ratio selected as best (%).

Qualitative Comparison. Figure 7 shows the qualitative comparison of PSGAN with other state-of-the-art methods on frontal faces in neutral expressions. Since the code of BeautyGlow and PairedCycleGAN is not released, we follow the strategy of BeautyGlow which cropped the results from corresponding papers. The result produced by DIA has an unnatural color on hair and background since it performs transfer in the whole image. CycleGAN can only synthesize general makeup style with is not similar to the

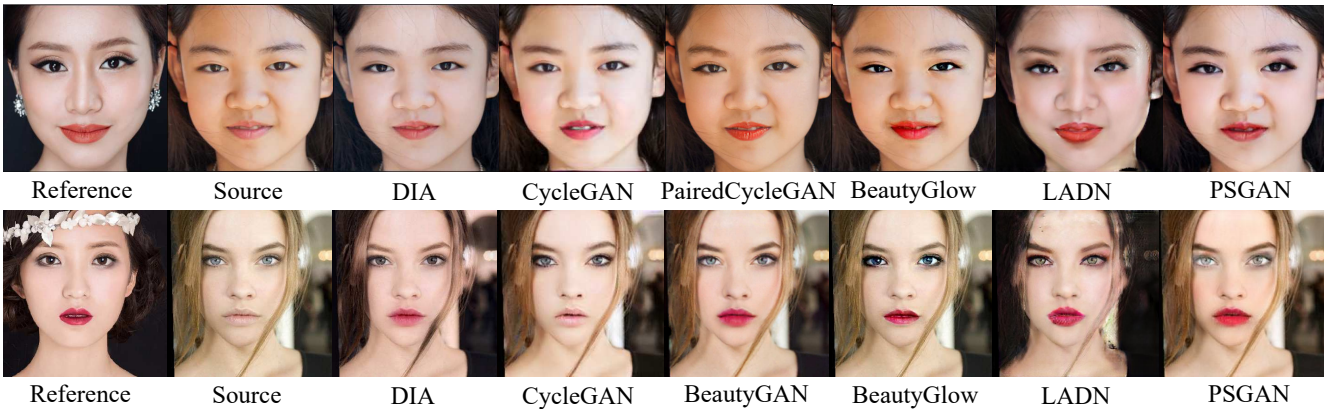


Figure 7. Qualitative comparison. PSGAN is able to generate realistic images with the same makeup styles as the reference.

Figure 8. Qualitative comparison on M-Wild test set.

reference. Besides, BeautyGlow fails to preserve the color of pupils and does not have the same foundation makeup as reference. We also use the pre-trained model released by the author of LADN, which produces blurry transfer results and unnatural background. Compared to the baselines, our method is able to generate vivid images with the same makeup styles as reference.

We also conduct a comparison on the M-Wild test set with the state-of-the-art method (BeautyGAN and LADN) that provide code and pre-trained model, as shown in Figure 8. Since the current methods lack an explicit mechanism to guide the direction of transfer at the pixel-level and also overfit on frontal images, the makeup is applied in the wrong region of the face when dealing with images with different poses and expressions. For example, the lip gloss is transferred to the skin on the first row of Figure 8. In the second row, other methods fail to perform transfer on faces with different sizes. However, our AMM module can accurately assign the makeup for every pixel through calculating the similarities, which makes our results look better.

4.6. Video Makeup Transfer

To transfer makeup for a person in the video is a challenging and meaningful task, which has wide prospects in the applications. However, the pose and expression of a face in the video are continuously changing which brings extra difficulties. To examine the effectiveness of our method,

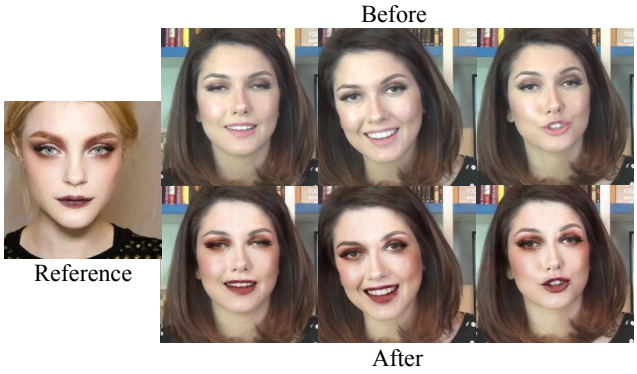


Figure 9. Video makeup transfer results of PSGAN.

we simply perform makeup transfer on every frame of the video, as shown in Figure 9. By incorporating the design of PSGAN, we receive nice and stable transferred results.

5. Conclusion

In order to bring makeup transfer to real-world applications, we propose PSGAN that first distills the makeup style into two makeup matrices from the reference and then leverages an Attentive Makeup Morphing (AMM) module to conduct makeup transfer accurately. The experiments demonstrate our approach can achieve state-of-the-art transfer results on both frontal facial images and facial images that have various poses and expressions. Also, with the spatial-aware makeup matrices, PSGAN can transfer the makeup partially and adjust the shade of transfer, which greatly broadens the application range of makeup transfer. Moreover, we believe our novel framework can be used in other conditional image synthesis problems that require customizable and precise synthesis.

Acknowledgement This work is partially supported by the National Natural Science Foundation of China (Grant 61572493, Grant 61876177), Beijing Natural Science Foundation (L182013, 4202034) and Fundamental Research Funds for the Central Universities. This work is also sponsored by Zhejiang Lab (No.2019KD0AB04). We also thank Jinyu Chen for his help.

References

[1] Taleb Alashkar, Songyao Jiang, Shuyang Wang, and Yun Fu. Examples-rules guided deep neural network for makeup recommendation. In *AAAI*, 2017. 2

[2] Huiwen Chang, Jingwan Lu, Fisher Yu, and Adam Finkelstein. Pairedcyclegan: Asymmetric style transfer for applying and removing makeup. In *CVPR*, 2018. 1, 2, 7

[3] Hung-Jen Chen, Ka-Ming Hui, Sishui Wang, Li-Wu Tsao, Hong-Han Shuai, Wen-Huang Cheng, and National Chiao Tung. Beautyglow : On-demand makeup transfer framework with reversible generative network. In *CVPR*, 2019. 1, 2, 5, 7

[4] Yunjey Choi, Min-Je Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*, 2017. 4

[5] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. *ArXiv*, abs/1610.07629, 2016. 2

[6] Chen Gao, Yunpeng Chen, Si Liu, Zhenxiong Tan, and Shuicheng Yan. Adversarialnas: Adversarial neural architecture search for gans. *ArXiv*, abs/1912.02037, 2019. 3

[7] Leon A. Gatys, Matthias Bethge, Aaron Hertzmann, and Eli Shechtman. Preserving color in neural artistic style transfer. *ArXiv*, abs/1606.05897, 2016. 2

[8] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. A neural algorithm of artistic style. *ArXiv*, abs/1508.06576, 2015. 2

[9] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *CVPR*, 2016. 2

[10] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014. 1

[11] Qiao Gu, Guanzhi Wang, Mang Tik Chiu, Yu-Wing Tai, and Chi-Keung Tang. Ladn: Local adversarial disentangling network for facial makeup and de-makeup. In *ICCV*, 2019. 1, 2, 7

[12] Dong Guo and Terence Sim. Digital face makeup by example. In *CVPR*, 2009. 2

[13] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2017. 3

[14] Xun Huang and Serge J. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, 2017. 2

[15] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016. 2

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ArXiv*, abs/1412.6980, 2014. 6

[17] Chen Li, Kun Zhou, and Stephen Lin. Simulating makeup through physics-based manipulation of intrinsic image layers. In *CVPR*, 2015. 2

[18] Tingting Li, Ruihe Qian, Chao Dong, Si Liu, Qiong Yan, Wenwu Zhu, and Liang Lin. Beautygan: Instance-level facial makeup transfer with deep generative adversarial network. In *ACM MM*, 2018. 1, 2, 5, 7

[19] Jing Liao, Yuan Yao, Lu Yuan, Gang Hua, and Sing Bing Kang. Visual attribute transfer through deep image analogy. *ACM TOG*, 2017. 7

[20] Luoqi Liu, Junliang Xing, Si Liu, Hui Xu, Xi Zhou, and Shuicheng Yan. "wow! you are so beautiful today!". In *ACM MM*, 2013. 2

[21] Si Liu, Xinyu Ou, Ruihe Qian, Wei Wang, and Xiaochun Cao. Makeup like a superstar: Deep localized makeup transfer network. In *IJCAI*, 2016. 2

[22] Fujun Luan, Sylvain Paris, Eli Shechtman, and Kavita Bala. Deep photo style transfer. In *CVPR*, 2017. 2

[23] Volodymyr Mnih, Nicolas Manfred Otto Heess, Alex Graves, and Koray Kavukcuoglu. Recurrent models of visual attention. In *NeurIPS*, 2014. 3

[24] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*, 2019. 3

[25] Alexander M. Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. In *EMNLP*, 2015. 3

[26] Yaniv Taigman, Adam Polyak, and Lior Wolf. Unsupervised cross-domain image generation. *ICLR*, 2016. 2

[27] Wai-Shun Tong, Chi-Keung Tang, Michael S. Brown, and Ying-Qing Xu. Example-based cosmetic transfer. In *PG*, 2007. 2

[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 3

[29] Xiaolong Wang, Ross B. Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2017. 3

[30] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015. 3

[31] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *Signal Processing Letters*, 2016. 4

[32] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2016. 6

[33] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 1, 5, 7