Local Facial Makeup Transfer via Disentangled Representation

Zhaoyang Sun †, Wenxuan Liu †, Feng Liu †,
Ryan Wen Liu ‡, Shengwu Xiong †*  
† School of Computer Science and Technology,
‡School of Navigation, Wuhan University of Technology, Wuhan, China

Abstract. Facial makeup transfer aims to render a non-makeup face image in an arbitrary given makeup one while preserving face identity. The most advanced method separates makeup style information from face images to realize makeup transfer. However, makeup style includes several semantic clear local styles which are still entangled together. In this paper, we propose a novel unified adversarial disentangling network to further decompose face images into four independent components, i.e., personal identity, lips makeup style, eyes makeup style and face makeup style. Owing to the further disentangling of makeup style, our method can not only control the degree of global makeup style, but also flexibly regulate the degree of local makeup styles which any other approaches can’t do. For makeup removal, different from other methods which regard makeup removal as the reverse process of makeup, we integrate the makeup transfer with the makeup removal into one uniform framework and obtain multiple makeup removal results. Extensive experiments have demonstrated that our approach can produce more realistic and accurate makeup transfer results compared to the state-of-the-art methods.

Keywords: makeup transfer, disentangled representation, generative adversarial network

1 Introduction

In daily life, it has always been of special interest to humans to improve looks. Consider this scenario: when people see a favorite makeup style, they may always involuntarily envision what effects will be if they wear that makeup. Some applications now offer this virtual makeup function, such as TAAZ, MEITU XIUXIU and DailyMakever1. However, these tools only provide a limited number of defined makeup styles and sometimes require user’s interaction. Makeup transfer is a way to handle the transfer of arbitrary makeup styles without user’s interaction. Due to the diversity and complexity of makeup styles, accurate makeup transfer has always been a very challenging task.

* Corresponding author: xiongw@whut.edu.cn
1 taaaz.com, xiuxiu.web.meitu.com, dailymakeover.com
The input of makeup transfer is a non-makeup source image and an arbitrary makeup reference image, the output result receives the makeup style from the reference image while preserving face identity from the source image. Existing approaches \[12,1,5,21\] based on deep learning to makeup transfer have yielded visually pleasant results. Among them, the state-of-the-art method \[5\] extracts makeup information from face images, which not only achieves exciting results, but also can control the degree of makeup style. However, makeup style includes several semantic clear local styles, such as lipstick, eye shadow and foundation, which are still tangled in the makeup information.

This paper introduces an autoencoder architecture to solve this problem. We assume that the identity information adds the makeup style information which composed of different semantic regions equals the makeup image. So our generator contains four encoders to extract the information of personal identity, lips makeup style, eyes makeup style and face makeup style respectively. Inspired by recent advances in disentangled representation \[7,9,15,13\], the local makeup loss function we designed forces the disentangling of information, instead of dividing the face image into different regions according to the semantic information and feeding them to the corresponding encoders. For makeup transfer, we feed the identity latent variable from the non-makeup image and the corresponding makeup style variables into the decoder to obtain the local and global makeup transfer results, as shown in the Fig. 1.

For makeup removal, unlike the existing method \[12,15\], we consider a face without makeup to be a special case of the makeup face. Therefore, our approach treats makeup removal and makeup transfer as the same problem. We generate makeup removal results by feeding the identity latent variable from makeup image and the makeup style variables from non-makeup image into the decoder, as illustrated in Fig. 2. The makeup removal process could generate multiple results as the input non-makeup image changes.

The main contributions of this paper are summarized below:
(1) The proposed autoencoder framework can seamlessly generate more faithful and more realistic makeup transfer results compared to the state-of-the-art methods.

(2) With the further disentangling of makeup style, our method can not only transfer the local and global makeup style, but also flexibly control the degree of every local makeup styles.

(3) We integrate the makeup transfer with the makeup removal into one uniform framework and obtain multiple makeup removal results.

2 Related Work

Makeup transfer: To address this issue, Tong et al. [20] mapped the cosmetic contributions of color to non-makeup source images. [6,10] decomposed images into several layers and transferred each layer by warping the reference makeup image to the non-makeup one. These previous methods require accurate face feature point detection and pairs of images of the same person. Inspired by recent successful style transfer [4], Liu et al. [13] applied the style transfer technique on facial local components and achieved the makeup transfer. Li et al. [12] tackled the makeup transfer problem by incorporating an instance-level makeup loss into the CycleGAN [22] and generated visually pleasant makeup transfer results. However, the makeup loss is a crude estimate of the makeup similarity between two images. Chang et al. [1] extended the CycleGAN [22] to asymmetric networks to enable transferring specific style and removing makeup style together. But the process is completed within a black-box network and can’t control the makeup degree. The state-of-the-art method proposed by Gu et al. [5] achieved disentanglement of makeup latent variable from non-makeup features and enabled successfully transferring makeup style on conventional styles and complex styles by using multiple and overlapping local discriminators. However, the separated makeup variables still contain several semantic clear local makeup styles which are tangled, our method further decomposes the makeup component into lips makeup style, eyes makeup style and face makeup style for local makeup transfer. As our knowledge, we are the first to achieve the disentanglement of local makeup styles from face images.

Disentangled representation: Disentangled representation means learning several independent representations from the input data. In unsupervised image-to-image translation tasks, Huang et al. [7] and Lee et al. [9] decomposed image representation into a domain-invariant content variable and a domain-specific style variable to generate multi-modal outputs. Ma et al. [15] disentangled a person’s image into three main factors, namely foreground, background and pose, then manipulated the factors to generate a new image. Lorenz et al. [14] introduced an approach for disentangling appearance and shape by learning parts consistently over all instances of a category. Esser et al. [3] enforced disentanglement of the information by an additional classifier that estimates the minimal amount of regularization required. Inspired by these advances in disentangled
Our generator contains four encoders \( \{E^C, E^L, E^S, E^F\} \) to extract the information of personal identity, lips makeup style, eyes makeup style and face makeup style respectively. We exchange all the makeup latent variables for global makeup transfer and makeup removal in (a). The corresponding local makeup latent variables are recombined for local makeup transfer in (b).

representation, we disentangle an arbitrary face image into four independent components, including one personal identity and three local makeup styles, then realize makeup transfer and makeup removal by exchanging the corresponding makeup style components, see Fig. 2.

3 Method

3.1 Problem Formulation

Let image sets of non-makeup faces and makeup faces be \( X \subset \mathbb{R}^{H \times W \times 3} \) and \( Y \subset \mathbb{R}^{H \times W \times 3} \) respectively. \( \{x_i\}_{i=1,\ldots,M}, x_i \in X \) and \( \{y_j\}_{j=1,\ldots,N}, y_j \in Y \) represent non-makeup examples and makeup examples respectively, where the \( M,N \) are the number of non-makeup images and makeup images. In the unsupervised setting, there are not the same makeup style and the same identity in the dataset used.

For makeup transfer, the goal is learning mapping functions \( \Phi_{\text{transfer}} : x_i, y_j \rightarrow y_i^{\text{transfer}} \) where \( y_i^{\text{transfer}} \) has the same makeup style with \( y_j \) while preserving the identity from \( x_i \). The local makeup transfer problem can be defined as \( \Phi_k : x_i, y_j \rightarrow y_i^k \) where \( k \) represents different semantic regions and \( k \in \{ \text{lips, eyes, face} \} \) in this paper, \( y_i^k \) receives the local makeup style of \( k \) from \( y_j \) while other regions should be identical to \( x_i \). Note that \( k \) is not limited to this assignment and the face region defined here does not include lips and eyes.

For makeup removal, since the makeup process may conceal the original appearance, the original face behind cosmetics may have multiple possible results. We assume that a face without makeup to be a special case of the makeup face. Under this assumption, our approach treats makeup removal and makeup transfer as the same problem, makeup removal aims to transfer the makeup style from
a non-makeup face image to a makeup face. It sounds strange, because we regard the original appearance of the skin as a makeup style. The makeup removal problem can be similarly defined as $\Phi_{\text{removal}} : y_j, x_i \rightarrow x_j^{\text{removal}}$, an unsupervised image translation problem with conditioning, where $x_j^{\text{removal}}$ receives the identity from $y_j$ and the makeup style from $x_i$.

### 3.2 Network Architecture

For makeup transfer and removal, we separate several independent local makeup style latent variables from non-makeup features. Therefore, we define the identity content space $C$ which includes the non-makeup features, the lips attribute space $L$, eyes attribute space $S$ and face attribute space $F$ that capture the lips makeup style, eyes makeup style, face makeup style respectively. As illustrated in Fig. 2, our generator architecture consists of a content encoder $\{E^C\}$, three attribute encoders $\{E^L, E^S, E^F\}$ and a decoder $\{G\}$. Inspired by the U-Net structure \[17\], we add skip connections between the content encoder $\{E^C\}$ and the decoder $\{G\}$ for capturing more identity details. The latent variables $C, L, S, F$ are concatenated along the channel at the bottleneck. See Fig. 3 for the whole network architecture.

### 3.3 Makeup Transfer and Removed

As shown in Fig. 3 we extract the personal identity, lips makeup style, eyes makeup style and face makeup style from a non-makeup image and a makeup image, denote as $C_i = E^C(x_i), L_i = E^L(x_i), S_i = E^S(x_i), F_i = E^F(x_i)$ and $C_j = E^C(y_j), L_j = E^L(y_j), S_j = E^S(y_j), F_j = E^F(y_j)$, which are then fed into the decoder $G$ to generate the makeup transfer result $y_i^{\text{transfer}}$ and makeup removal result $x_j^{\text{removal}}$. The formula is described as follows:

$$y_i^{\text{transfer}} = G(C_i, L_j, S_j, F_j)$$

$$x_j^{\text{removal}} = G(C_j, L_i, S_i, F_i)$$

A difficult question lies ahead of us: how to evaluate the makeup similarity of a pair of facial images? Inspired by \[1\], we firstly generate synthetic ground truth $W(x_i, y_j)$ by warping $y_j$ onto $x_i$ according to the facial landmarks, then use Colour Profile (CP) loss proposed in \[18\] as the makeup loss to evaluate the makeup similarity of generated image $y_i^{\text{transfer}}$ and synthetic ground truth $W(x_i, y_j)$. In a similar way, we use Colour Profile (CP) loss to evaluate the makeup similarity of generated image $x_j^{\text{removal}}$ and synthetic ground truth $W(y_j, x_i)$. Makeup loss function is as follows:

$$L_{\text{makeup}} = -(CP(y_i^{\text{transfer}}, W(x_i, y_j)) + CP(x_j^{\text{removal}}, W(y_j, x_i)))$$
Fig. 3. The architecture of our whole network. The input $x_i$ and $y_j$ through the encoders to get four independent latent variables. First, feed the variables directly to the decoder to get $\tilde{x}_i^{\text{self}}$ and $\tilde{y}_j^{\text{self}}$. After that, the variables that extracted the local makeup information are exchanged and fed into the decoder to generate $y_i^{\text{transfer}}$ and $x_j^{\text{removal}}$. Finally, feed the outputs $\tilde{y}_i^{\text{transfer}}$ and $\tilde{x}_j^{\text{removal}}$ as inputs to the network again to obtain $\tilde{x}_i^{\text{cross}}$ and $\tilde{y}_j^{\text{cross}}$. For local makeup transfer, the corresponding local makeup latent variables are recombined and fed into the decoder.

3.4 Local Makeup Transfer

For the local makeup transfer, we only exchange the local disentangled makeup latent variable.

$$y_i^L = G(C_i, L_j, S_i, F_i), y_i^S = G(C_i, L_i, S_j, F_i), y_i^F = G(C_i, L_i, S_i, F_j)$$ (4)

where the $y_i^L$, $y_i^S$, $y_i^F$ represent the results of the local makeup transfer of the lips, eyes and face regions respectively.

The target of local makeup transfer has two points. 1) The generated result has the same makeup style with the reference makeup image in the specified semantic region. 2) The other semantic regions of the generated result should be identical to the non-makeup image. We use the L1 loss to encourage such local invariant and the local makeup loss function is as follows:

$$L_{\text{local}} = \sum_{k \in \{L, S, F\}} (\lambda_k CP(y_k^k \circ M_k, W(x_i, y_j) \circ M_k) + \mu_k \| y_k^k \circ M_k - x_i \circ \overline{M}_k \|_1)$$ (5)

where $\lambda_k, \mu_k$ are the weights, $\circ$ denotes element-wise multiplication, $M_k$ denotes the mask of corresponding face semantic region and $\overline{M}_k$ stands for reverse. Note
that this local makeup loss function drives the disentangling of latent variables instead of feeding the corresponding encoders with the images which segmented by semantics.

3.5 Other Loss Functions

**Reconstruction loss:** Inspired by CycleGAN [22], we introduce the reconstruction loss function into the network. The reconstruction loss function consists of two parts, one is the self reconstruction, the other is the cross-cycle reconstruction. We feed \( C_i, L_i, S_i, F_i \) into \( G \) to generate \( \tilde{x}_i^{\text{self}} \), feed \( C_j, L_j, S_j, F_j \) into \( G \) to obtain \( \tilde{y}_j^{\text{self}} \), which should be identical to \( x_i, y_j \) respectively. After obtaining the makeup transfer result \( y_{i}^{\text{transfer}} \) and makeup removal result \( x_{j}^{\text{removal}} \), we feed them as input to the network again and generate \( \tilde{x}_i^{\text{cross}}, \tilde{y}_j^{\text{cross}} \), which should be also identical to \( x_i, y_j \) respectively. We define the reconstruction loss as:

\[
L_{\text{rec}} = \| \tilde{x}_i^{\text{self}} - x_i \|_1 + \| \tilde{y}_j^{\text{self}} - y_j \|_1 \tag{6}
\]

\[
L_{\text{cycle}} = \| \tilde{x}_i^{\text{cross}} - x_i \|_1 + \| \tilde{y}_j^{\text{cross}} - y_j \|_1 \tag{7}
\]

**Perceptual loss:** We employ the perceptual loss to calculate the difference between \( x_i \) and \( y_i^k \) in the \( l \)-th layer of VGG-16 [19] which pre-trained on ImageNet Dataset to preserve the personal identity.

\[
L_{\text{per}} = \sum_{i=k}^{k \in \{L,S,F\}} \| A_l(y_i^k) - A_l(x_i) \|_2 \tag{8}
\]

where \( \| \cdot \|_2 \) is the L2 loss and \( A_l(\cdot) \) denotes the output of the \( l \)-th layer.

**Adversarial loss:** We employ generated adversarial network to match the distribution of translated images to the target data distribution. Our network contains three discriminators, \( D_X, D_Y, D_{XY} \). \( D_X \) distinguishes the generated image \( x_{j}^{\text{removal}} \) from real samples in set X. Similarly, \( D_Y \) try to distinguish the makeup transfer result \( y_{i}^{\text{transfer}} \) from real samples in set Y. Due to \( y_i^k \) only receives local makeup style from the makeup images, \( D_{XY} \) distinguishes the local makeup transfer result \( y_i^k \) from real samples in set X and set Y. We replace the negative log likelihood objective by a least square loss [16] in adversarial loss:

\[
L_X = \mathbb{E}_{x_i}[(D_X(x_i) - 1)^2] + \mathbb{E}_{x_j}^{\text{removal}}[(D_X(x_j^{\text{removal}}))^2] \tag{9}
\]

\[
L_Y = \mathbb{E}_{y_j}[(D_Y(y_j) - 1)^2] + \mathbb{E}_{y_i}^{\text{transfer}}[(D_Y(y_i^{\text{transfer}}))^2] \tag{10}
\]

\[
L_{XY} = \mathbb{E}_{x_i}[(D_{XY}(x_i) - 1)^2] + \mathbb{E}_{y_j}[(D_{XY}(y_j) - 1)^2] + \sum_{i=k}^{k \in \{L,S,F\}} \mathbb{E}_{y_i^k}[(D_{XY}(y_i^k))^2] \tag{11}
\]
Global local consistent loss: Understandably, the results of the local makeup transfer should be consistent with the global makeup transfer result in the corresponding semantic region. We use the L1 loss to encourage such global local consistency:

$$L_{gl} = \sum_{i=k}^{k \in \{L,S,F\}} \| y^{transfer}_{i} \circ M_k - y^{k}_{i} \circ M_k \|_1$$  (12)

Total loss: To sum up, our total loss is

$$L_{total} = \lambda_{makeup} L_{makeup} + \lambda_{local} L_{local} + \lambda_{rec} L_{rec} + \lambda_{cycle} L_{cycle} + \lambda_{per} L_{per} + \lambda_{adv} L_{adv} + \lambda_{gl} L_{gl}$$  (13)

where $\lambda_{makeup}, \lambda_{local}, \lambda_{rec}, \lambda_{cycle}, \lambda_{per}, \lambda_{adv}, \lambda_{gl}$ are weights that control the importance of different objectives.

4 Experiments

4.1 Data Set and Training Details

We use the makeup transfer data set released by Li et al. [5] to conduct all the experiments, which contains 333 non-makeup and 302 makeup high-quality face images. During the experiments, we remove some extreme makeup images, because the semantic regions of the faces in these images are not clear.

During training, the input images are resized to 286 × 286, randomly cropped to 256 × 256 and horizontally flipped with a probability of 0.5 for data augmentation. We set $\lambda_{makeup} = 4, \lambda_{local} = 1, \lambda_{rec} = 5, \lambda_{cycle} = 40, \lambda_{per} = 0.1, \lambda_{adv} = 6, \lambda_{gl} = 20$ to balance different objectives. In $L_{local}$, we give more attention to the lips and eyes makeup, because these two semantic regions are smaller than the face region. We set $\lambda_L = 50, \lambda_S = 10, \lambda_F = 1$ and set $\mu_L = 1, \mu_S = 1, \mu_F = 4$ for the same reason. In $L_{per}$, the $relu_{4\_1}$ feature layer is applied for identity preserving. We employ the Adam [8] optimizer to train our network for 600 epochs in all, where the learning rate is fixed as 0.0002. The batch size is set as 1. The specific structure of the content encoder, attribute encoders and decoder we refer to [5]. The only difference is that the number of output channels of the attribute encoder is reduced by half. For discriminators, we leverage identical 70 × 70 PatchGANs [11], which distinguishes local image patches to be real or fake. Since the generator is a much harder problem, we train the generator ten times more frequently than the discriminators.

4.2 Makeup Transfer Results

As demonstrated in Fig. 4 we choose 5 images with very different makeup styles for reference and compare our results with three previous methods, BeautyGAN [12], DMT [21], LADN [5], from qualitative perspectives. The results of other
methods are derived from official code or trained models. BeautyGAN and DMT could generate visually pleasant transferring results with the histogram match loss. For simple eye shadows, like the first and third rows, the results of BeautyGAN and DMT are generally satisfactory. But when the eye shadow style is more complicated, the style of eye shadows have not been correctly transferred as well. For example, the eye shadow is not reflected in the generated results of BeautyGAN and DMT in row 2, row 4 and row 5. LADN is the state-of-the-art for facial makeup transfer by incorporating multiple style discriminators specialized for different facial patches. We observed that LADN could handle most eye shadow styles well, but the results are not satisfactory in the last two rows. Meanwhile, the resulting foundation color is different from the reference image. This phenomenon is evident in the results of row 1, row 2, and row 5. By contrast, no matter what kind of makeup styles, our methods have yielded satisfactory results. And our outputs are highly consistent with the makeup style of reference images, no matter in lipstick, eye shadow or foundation, while better
Fig. 5. The makeup removal results. Our approach treats makeup removal and makeup transfer as the same problem. The difference is that the roles of the non-makeup images and the makeup pictures are changed. The first row is the non-makeup reference images and the first column is the makeup source images. The corresponding makeup removal results receive the identity from the makeup source images and the makeup style from the non-makeup reference images.

protecting the character’s identity. Our method can even transfer the shadow on both sides of the cheek to the results, see row 2. We provide additional results in the supplementary material.

4.3 Makeup Removal Results

The use of many cosmetics masks the natural appearance of the face. Restore the effect without makeup from the makeup image, there may be a variety of the results. Other methods [115] regard makeup removal as a problem without conditioning. This assumption is more realistic, but yields only a single result. Under our assumption, makeup removal is treated as an unsupervised image translation problem with conditioning. We can get multiple realistic cosmetic removal results by feeding the makeup components from different non-makeup images into the decoder, as demonstrated in Fig. 5. Extensive experimental results further verify the validity of our assumption.
4.4 Local Makeup Transfer Results

The methods \cite{13} train multiple networks to perform local makeup transfer and can’t control the degree of makeup style. By contrast, our approach decomposes the latent variables of makeup style into lips style, eyes style and face style in a network, and can not only the local makeup transfer, but also flexibly control the degree of local makeup style which will be shown in the next section. The local makeup transfer results are shown in Fig. 6, our results effectively transfer the local makeup style while keeping the rest regions of the face image unchanged.

4.5 Interpolated Makeup Transfer Results

Because of separating several makeup style latent variables from non-makeup features, we can control the degree of local makeup styles. The formula is described as follows:

\[ y_{i}^{\text{inter}} = G(C_i, \alpha_L L_i + (1 - \alpha_L)L_j, \alpha_S S_i + (1 - \alpha_S)S_j, \alpha_F F_i + (1 - \alpha_F)F_j) \]  

where \( \alpha_L, \alpha_S, \alpha_F \in [0, 1] \) are the weights to control the degree of makeup style. We set \( \alpha_L, \alpha_S, \alpha_F \) to be the same value and generate the interpolated results shown in Fig. 7. Then we fix two of them to 0 or 1 and gradually change the other in Fig. 8 and Fig. 9. We have observed that no matter which kind of interpolation transfer, the generator can produce smooth, realistic results.
Fig. 7. The interpolated results of global makeup style transfer. The first and second columns are the non-makeup images and the makeup reference images respectively. The remaining columns from left to right are the interpolated results. We gradually change $\alpha_L, \alpha_S, \alpha_F$ from 1 to 0.

4.6 Mix Local Makeup Transfer Results

In this article, we will further try a very challenging task, mix local makeup transfer we called. We would extract the lips style, eyes style, and face style from three different makeup images and then mix them into one non-makeup image while preserving face identity. This puts forward higher requirements on the disentangling degree of information and the effect of generator. The formula is described as follows:

$$y_{mix}^i = G(C_i, L_p, S_q, F_r),$$

(15)

where $L_p$, $S_q$ and $F_r$, respectively, denote the lips, eyes and face latent variables from three different makeup images. The result show in Fig. 10. Our results are as consistent as possible with the local makeup styles without losing the sense of authenticity. More results will be provided in our supplementary material.

5 Conclusion

In conclusion, our method decomposes face images into four independent components, including personal identity, lips makeup style, eyes makeup style and face makeup style, then exchange the corresponding makeup components to achieve local or global makeup transfer. Benefit by the disentangling of information, our method can control the degree of local makeup styles and produce a smooth transition effect. For makeup removal, we consider a face without makeup to be a special case of the makeup face and integrate the makeup problem with the makeup removal problem into one uniform framework to obtain multiple
Fig. 8. The interpolated results of local makeup style transfer. The first two rows are the interpolated results of lips makeup style, we fix $\alpha_S = 1, \alpha_F = 1$, and gradually change $\alpha_L$ from 1 to 0. The middle two rows are the interpolated results of eyes makeup style, we fix $\alpha_L = 1, \alpha_F = 1$, and gradually change $\alpha_S$ from 1 to 0. The last two rows are the interpolated results of face makeup style, we fix $\alpha_L = 1, \alpha_S = 1$, and gradually change $\alpha_F$ from 1 to 0.

Fig. 9. The interpolated results of local makeup style transfer. The first two rows are the interpolated results of lips makeup style, we fix $\alpha_S = 0, \alpha_F = 0$, and gradually change $\alpha_L$ from 1 to 0. The middle two rows are the interpolated results of eyes makeup style, we fix $\alpha_L = 0, \alpha_F = 0$, and gradually change $\alpha_S$ from 1 to 0. The last two rows are the interpolated results of face makeup style, we fix $\alpha_L = 0, \alpha_S = 0$, and gradually change $\alpha_F$ from 1 to 0.
Fig. 10. The mix local makeup transfer results. The last row is the results of the mix local makeup transfer, which receive personal identity from the first row, the lips style from the second row, the eyes style from the third row and the face style from the fourth row.

makeup removal results. Extensive experiments have verified the effectiveness of our method compared with other methods.
References

1. Chang, H., Lu, J., Yu, F., Finkelstein, A.: Pairedcyclegan: Asymmetric style transfer for applying and removing makeup. In: CVPR (2018)
2. Chen, H.J., Hui, K.M., Wang, S.Y., Tsao, L.W., Shuai, H.H., Cheng, W.H.: Beautyglow: On-demand makeup transfer framework with reversible generative network. In: CVPR (2019)
3. Esser, P., Haux, J., Ommer, B.: Unsupervised robust disentangling of latent characteristics for image synthesis. In: ICCV (2019)
4. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: CVPR (2016)
5. Gu, Q., Wang, G., Chiu, M.T., Tai, Y.W., Tang, C.K.: Ladn: Local adversarial disentangling network for facial makeup and de-makeup. In: ICCV (2019)
6. Guo, D., Sim, T.: Digital face makeup by example. In: CVPR (2009)
7. Huang, X., Liu, M.Y., Belongie, S.J., Kautz, J.: Multimodal unsupervised image-to-image translation. In: ECCV (2018)
8. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
9. Lee, H.Y., Tseng, H.Y., Huang, J.B., Singh, M., Yang, M.H.: Diverse image-to-image translation via disentangled representations. In: ECCV (2018)
10. Li, C., Zhou, K., Lin, S.: Simulating makeup through physics-based manipulation of intrinsic image layers. In: CVPR (2015)
11. Li, C., Wand, M.: Precomputed real-time texture synthesis with markovian generative adversarial networks. In: ECCV (2016)
12. Li, T., Qian, R., Dong, C., Liu, S., Yan, Q., Zhu, W., Lin, L.: Beautygan: Instance-level facial makeup transfer with deep generative adversarial network. In: ACM MM (2018)
13. Liu, S., Ou, X., Qian, R., Wang, W., Cao, X.: Makeup like a superstar: Deep localized makeup transfer network. In: IJCAI (2016)
14. Lorenz, D., Bereska, L., Milbich, T., Ommer, B.: Unsupervised part-based disentangling of object shape and appearance. In: CVPR (2019)
15. Ma, L., Sun, Q., Georgoulis, S., Gool, L.V., Schiele, B., Fritz, M.: Disentangled person image generation. In: CVPR (2018)
16. Mao, X., Li, Q., Xie, H., Lau, R.Y.K., Wang, Z.: Multi-class generative adversarial networks with the l2 loss function. arXiv preprint arXiv:1611.04076 (2016)
17. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI (2015)
18. Sarfraz, M.S., Seibold, C., Khalid, H., Stiefelhagen, R.: Content and colour distillation for learning image translations with the spatial profile loss. In: BMVC (2019)
19. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)
20. Tong, W.S., Tang, C.K., Brown, M.S., Xu, Y.Q.: Example-based cosmetic transfer. In Proceedings of the Pacific Conference on Computer Graphics and Applications, Pacific Graphics 2007 (2007)
21. Zhang, H., Chen, W., He, H., Jin, Y.: Disentangled makeup transfer with generative adversarial network. arXiv preprint arXiv:1907.01144 (2019)
22. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: ICCV (2017)