Local Facial Makeup Transfer via Disentangled Representation

Zhaoyang Sun[1], Feng Liu[1], Wen Liu[2], Shengwu Xiong[1], and Wenxuan Liu[1]

[1] School of Computer Science and Technology, Wuhan University of Technology, Wuhan, China
xiongsw@whut.edu.cn
[2] 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 disentangled makeup representation, our method can not only flexible control the degree of local makeup styles, but also can transfer local makeup styles from different images into the final result, which any other approaches fail to handle. For makeup removal, different from other methods which regard makeup removal as the reverse process of makeup transfer, 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 visually pleasant and accurate makeup transfer results compared to the state-of-the-art methods.

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 this makeup. Some applications now offer this virtual makeup function, such as TAAZ, MEITU XIUXIU and DailyMakever[3]. However, these tools only provide a limited number of defined makeup styles and sometimes require specific interaction. Makeup transfer is a way to handle the transfer of arbitrary makeup styles without specific interaction. Due to the diversity and complexity of makeup styles, accurate makeup transfer has always been a very challenging task in both academia and industry.

In practical applications, users strongly hope that the combinations of different makeup styles can be freely achieved. It means that makeup transfer can be performed like trying on clothes, and can transfer local makeup styles from

[3] taaz.com, xiuxiu.web.meitu.com, dailymakeover.com

Fig. 1. Our results of the mix local makeup transfer. The last column (e) are the mix local makeup results, which have the identity information from (a), the lips style from (b), the eyes style in (c) and face style from (d).

different images into the final result, shown in Fig. 1. Existing approaches [1–6] based on deep learning to makeup transfer have yielded visually pleasant results. For example, Li *et al.* [1] adopted the structure of dualGAN [7] and histogram matching as the makeup loss to achieve pleasant results. Chang *et al.* [2] proposed a style discriminator to promote the makeup transfer in the asymmetric network. The above methods do not separate the makeup style information, the makeup transfer process is thus completed in a black-box network. Recently, the method [5] successfully extracts makeup information from face images, which not only achieves exciting results, but also can control the degree of makeup style. However, the extracted makeup information includes several semantic clear local makeup styles, such as lipstick, eye shadow and foundation, which are still tangled in the makeup information. Therefore, this method still fail to meet the higher requirements of practical applications.

This paper introduces an autoencoder architecture to solve this problem. In particular, our generator contains four encoders to extract the information of personal identity, lips makeup style, eyes makeup style, and face makeup style, respectively. After training, we feed the identity latent variable from the non-makeup image and the local makeup style variables from different makeup images into the decoder to obtain the result of local makeup style combination. The proposed framework is called the mix local makeup transfer in this work. Inspired by recent advances in disentangled representation [8–12], 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 the case of makeup removal, unlike the existing methods [1, 2, 5], we consider a face without makeup to be a special case of the makeup face. Our approach could thus treat the 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, illustrated in Fig. 2. The makeup removal process could generate multiple results as the input non-makeup image changes. The main contributions of this work are summarized as follows:

– The local makeup loss function we designed forces the disentangling of information, our method decompose face images into four independent components, personal identity, lips makeup style, eyes makeup style and face makeup style.
– With the further disentangling of makeup style, our method can not only flexibly control the degree of every local makeup styles, but also transfer local makeup styles from different images into the final result.
– We integrate the makeup transfer with the makeup removal into one uniform framework and obtain multiple makeup removal results.

2 Related Work

Makeup transfer: 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. To address this issue, Tong *et al.* [13] mapped the cosmetic contributions of color in the reference image to the non-makeup source image. [14, 15] decomposed face images into several layers and transferred each layer by warping the reference makeup image to the non-makeup one. Inspired by recent successful style transfer [16], Liu *et al.* [17] applied the style transfer technique on facial local components and achieved the makeup transfer. Li *et al.* [1] tackled the makeup transfer problem by incorporating a instance-level makeup loss into the dualGAN [7] and generated visually pleasant makeup transfer results. Chang *et al.* [2] extended the CycleGAN [18] to asymmetric networks to enable transferring specific style and removing makeup style together. But the processing of the above methods is completed within a black-box network and can't control the makeup degree. Recently, the method proposed by Gu *et al.* [5] achieved disentanglement of makeup latent variable from non-makeup features and exchange the makeup information of two pictures to realize the makeup transfer. However, the separated makeup variable contains several semantic clear local makeup styles which are still tangled. So this method still can't transfer local makeup styles from different makeup images into the final result. Our method further decomposes the makeup component into lips makeup style, eyes makeup style and face makeup style by the local makeup loss. 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.* [8] and Lee *et al.* [9] decomposed image

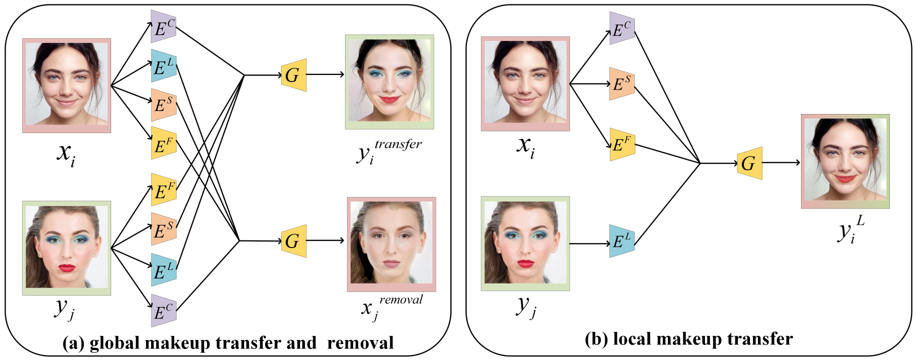

Fig. 2. Our generator contains four encoders $\{E^C, E^L, E^S, E^F\}$ to extract the information of personal identity, lips makeup style, eyes makeup style and face makeup style respectively. We exchange all the makeup latent variables for global makeup transfer and makeup removal in (a). The corresponding local makeup latent variables are recombined for local makeup transfer in (b).

representation into a domain-invariant content variable and a domain-specific style variable to generate multi-modal outputs. Ma *et al.* [10] 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.* [11] introduced an approach for disentangling appearance and shape by learning parts consistently over all instances of a category. Esser *et al.* [12] enforced disentanglement of the information by an additional classifier that estimates the minimal amount of regularization required. Inspired by these advances in disentangled 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. $\{x_i\}_{i=1,\cdots,M}$, $x_i \in X$ and $\{y_j\}_{j=1,\cdots,N}$, $y_j \in Y$ represent non-makeup examples and makeup examples, respectively. Here, M, N denote the numbers of non-makeup images and makeup images.

For makeup transfer, the goal is learning mapping functions $\Phi : x_i, y_j \rightarrow y_i^{transfer}$ where $y_i^{transfer}$ has the same makeup style with y_j while preserving the identity from x_i. The local makeup transfer problem can be defined as $\Phi_k : x_i, y_j \xrightarrow{k} y_i^k$, where k represents different semantic regions and $k \in \{lips, eyes, face\}$ in this paper, y_i^k receives the local makeup style of k from y_j

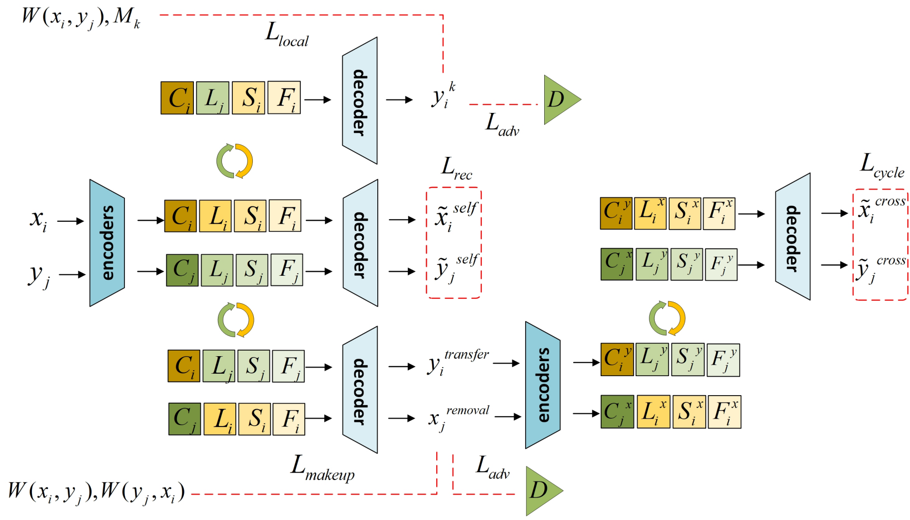

Fig. 3. The architecture of our whole network. The input x_i and y_j through the encoders to get four independent latent variables. First, feed the variables directly to the decoder to get $\widetilde{x}_i^{self}$ and $\widetilde{y}_j^{self}$. After that, the variables that extracted the local makeup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

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 $\Phi : y_j, x_i \rightarrow x_j^{removal}$, an unsupervised image translation problem with conditioning, where $x_j^{removal}$ receives the identity from y_j and the makeup style from x_i.

3.2 Makeup Transfer and Removed

As illustrated in Fig. 2, our generator architecture consists of a identity encoder $\{E^C\}$, a lip style encoder $\{E^L\}$, a eye style encoder $\{E^S\}$, a face style encoder $\{E^F\}$ and a decoder $\{G\}$. First of all, 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 $y_i^{transfer}$ and makeup removal result $x_j^{removal}$. The formula is described as follows:

$$y_i^{trasnfer} = G(C_i, L_j, S_j, F_j) \tag{1}$$

$$x_j^{removal} = G(C_j, L_i, S_i, F_i) \tag{2}$$

A difficult question lies ahead of us: how to evaluate the makeup similarity of a pair of facial images ? 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 loss proposed in [4] as the makeup loss to evaluate the makeup similarity of generated image $y_i^{transfer}$ and synthetic ground truth $W(x_i, y_j)$. In a similar way, we use Colour Profile loss to evaluate the makeup similarity of generated image $x_j^{removal}$ and synthetic ground truth $W(y_j, x_i)$. Makeup loss function is as follows:

$$L_{makeup} = -(CP(y_i^{trasnfer}, W(x_i, y_j)) + CP(x_j^{removal}, W(y_j, x_i))) \tag{3}$$

3.3 Local Makeup Transfer

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

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

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 :

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

where λ_k, μ_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. Face parsing is only used for network training to calculate loss functions, but not for testing.

3.4 Other Loss Functions

Reconstruction loss: As illustrated in Fig. 3, the reconstruction loss function consists of two parts, one is the self reconstruction, the other is the cross-cycle reconstruction [18]. We feed C_i, L_i, S_i, F_i into G to generate $\widetilde{x}_i^{self}$, feed C_j, L_j, S_j, F_j into G to obtain $\widetilde{y}_j^{self}$, which should be identical to x_i, y_j respectively. After obtaining the makeup transfer result $y_i^{trasnfer}$ and makeup removal result $x_j^{removal}$, we feed them as input to the network again and generate $\widetilde{x}_i^{cross}$, $\widetilde{y}_j^{cross}$, which should be also identical to x_i, y_j respectively. We define the reconstruction loss as :

$$L_{rec} = (\|\widetilde{x}_i^{self} - x_i\|_1 + \|\widetilde{y}_j^{self} - y_j\|_1) + \lambda_{scale}(\|\widetilde{x}_i^{cross} - x_i\|_1 + \|\widetilde{y}_j^{cross} - y_j\|_1) \quad (6)$$

Adversarial loss: We employ adversarial loss to improve the quality of generated images. The discriminator D distinguishes all the fake results from real samples in set X and set Y and the generator G tries to fool the discriminator D. We replace the negative log likelihood objective by a least square loss [19] in adversarial loss:

$$\begin{aligned} L_{adv} =& \mathbb{E}_{x_i}[(D(x_i) - 1)^2] + \mathbb{E}_{y_j}[(D(y_j) - 1)^2] + \\ & \mathbb{E}_{x_j^{removal}}[(D(x_j^{removal}))^2] + \mathbb{E}_{y_i^{trasnfer}}[(D(y_i^{trasnfer}))^2] + \\ & \sum_{i=k}^{k \in \{L,S,F\}} \mathbb{E}_{y_i^k}[(D(y_i^k))^2] \end{aligned} \quad (7)$$

Total loss: To sum up, our total loss is

$$L_{total} = \lambda_{makeup} L_{makeup} + \lambda_{local} L_{local} + \lambda_{rec} L_{rec} + \lambda_{adv} L_{adv} \quad (8)$$

where $\lambda_{makeup}, \lambda_{local}, \lambda_{rec}, \lambda_{adv}$ 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 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 $\lambda_{makeup} = 4, \lambda_{local} = 1, \lambda_{rec} = 5, \lambda_{adv} = 1$ to balance different objectives. In L_{local}, we gives 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.

Fig. 4. Ablation study on the local makeup loss. The images from (a) to (f) are the non-makeup source image, makeup reference image, lips makeup transfer result, eyes makeup transfer result, face makeup transfer result and global makeup transfer result respectively. The first and third rows are the results with local makeup loss and the second and fourth rows are the results without local makeup loss.

In L_{rec}, the λ_{scale} is set 8. We employ the Adam [20] optimizer to train our network for 1000 epochs in all, where the learning rate is fixed as 0.0002. The batch size is set as 1. For capturing more identity details, we add skip connections [21] between the encoder $\{E^C\}$ and the decoder $\{G\}$. The latent variables are concatenated along the channel at the bottleneck. 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 the PatchGANs [22], which distinguishes local image patches to be real or fake.

4.2 Does the Local Makeup Loss Work?

In the network training process, the input to each makeup encoder is the same entire unpreprocessed makeup reference image. The local makeup loss function we designed forces different encoders to learn different information and the disentangling of information. To evaluate the effect of the local makeup loss, the ablation of the local makeup loss function is studied. As shown in Fig. 4, the results with local makeup loss achieved the local makeup transfer target , which have the same makeup style with the reference makeup image in the specified

Fig. 5. Qualitative comparisons between our results and others. The columns from left to right are non-makeup images, referenced makeup images, the results of BeautyGAN, DMT, LADN and our method.

semantic region and the other semantic regions should be identical to the non-makeup image. When we remove this loss function, the effect of local makeup transfer disappears. The ablation experiment further illustrates that the local makeup loss function forces different encoders to learn different makeup information and promotes the decoupling of local makeup information.

4.3 Compare with other methods

Qualitative Comparison As demonstrated in Fig. 5, we compare our results with three previous methods, BeautyGAN [1], DMT [6], LADN [5], from qualitative perspectives. The results of other methods are derived from official code or trained models. BeautyGAN and DMT could generate visually realistic transferring results with the histogram match loss. But for the eye area, the style of eye shadows have not been correctly transferred as well. For example, the eye shadow is not reflected in the generated results in row 1, row 4. LADN achieves facial makeup transfer by incorporating multiple local style discriminators. We observed that LADN could handle most eye shadow styles well, but the results are not satisfactory in the last row. Meanwhile, the resulting foundation color is different from the reference image. This phenomenon is evident in the results of row 3, row 4. 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. Our method can even transfer the shadow on both sides of the cheek to the results, see row 1.

Table 1. Quantitative Comparison

Methods	Our	beautyGAN	DMT	LADN
percentage	48.35%	30.94%	12.23%	8.47%

Quantitative Comparison We randomly selected 5 non-makeup source images and 10 makeup reference images, and generated 50 results of makeup transfer using four methods respectively. Then 17 volunteers were recruited and asked to choose the result they were most satisfied with. As shown in the Table 1, Our method has a higher selection rate 48.35% than other methods beautyGAN 30.94%, DMT 12.23%, LADN 8.47%.

4.4 Other Results

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 [2, 5] 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. 6. Extensive experimental results further verify the validity of our assumption.

Local Makeup Transfer Results The methods [17, 2] 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. Our method 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. 7, our results effectively transfer the local makeup style while keeping the rest regions of the face image unchanged.

Interpolated Makeup Transfer Results Because of separating several makeup style latent variables from non-makeup features, we can control the degree of

Fig. 6. 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.

local makeup styles. The formula is described as follows:

$$y_i^{inter} = G(C_i, \alpha_L L_i + (1-\alpha_L)L_j, \alpha_S S_i + (1-\alpha_S)S_j, \alpha_F F_i + (1-\alpha_F)F_j) \quad (9)$$

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 global interpolated results. Then we fix two of them to 0 or 1 and gradually change the other. As shown in Fig. 8, we have observed that no matter which kind of interpolation transfer, the generator can produce smooth, realistic results.

Mix Local Makeup Transfer Results At the end of this article, we will further try a very challenging task mentioned at the beginning, 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

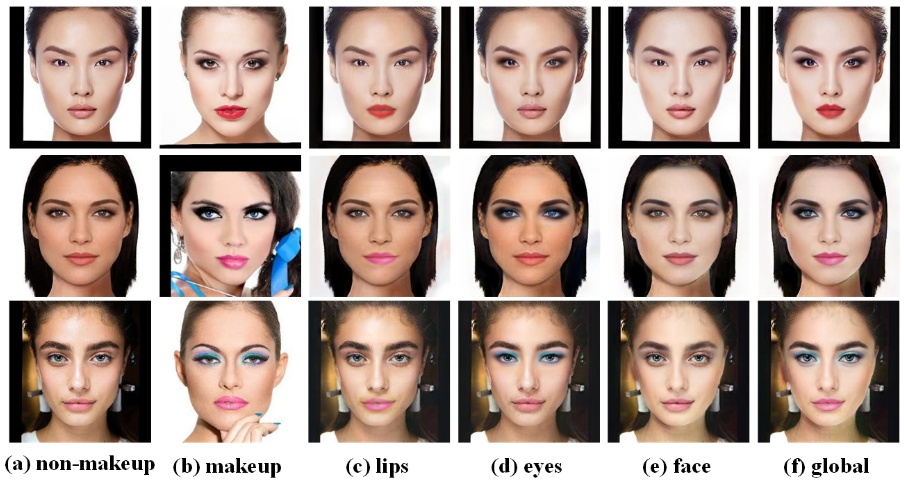

Fig. 7. The local makeup transfer results. The first and second columns are non-makeup images and makeup images respectively. The remaining columns from left to right are the results of lips style, eyes style, face style and global style transfer.

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:

$$y_i^{mix} = G(C_i, L_p, S_q, F_r), \tag{10}$$

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. 9. Our results are as consistent as possible with the local makeup styles without losing the sense of authenticity.

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. Benefit by the disentangling of information, our method can not only flexible control the degree of local makeup styles, but also can transfer local makeup styles from different images into the result. For makeup removal, we integrate the makeup transfer with the makeup removal into one uniform framework and obtain multiple makeup removal results. Extensive experiments have verified the effectiveness of our method compared with other methods. In addition, we tested the generalization capability of our method in complex s-cenarios, see Fig. 10. For faces with large-poses, our method can still obtain relatively satisfactory results. But a few failure cases of our method were caused by significantly different lighting conditions, which is the focus of our next work.

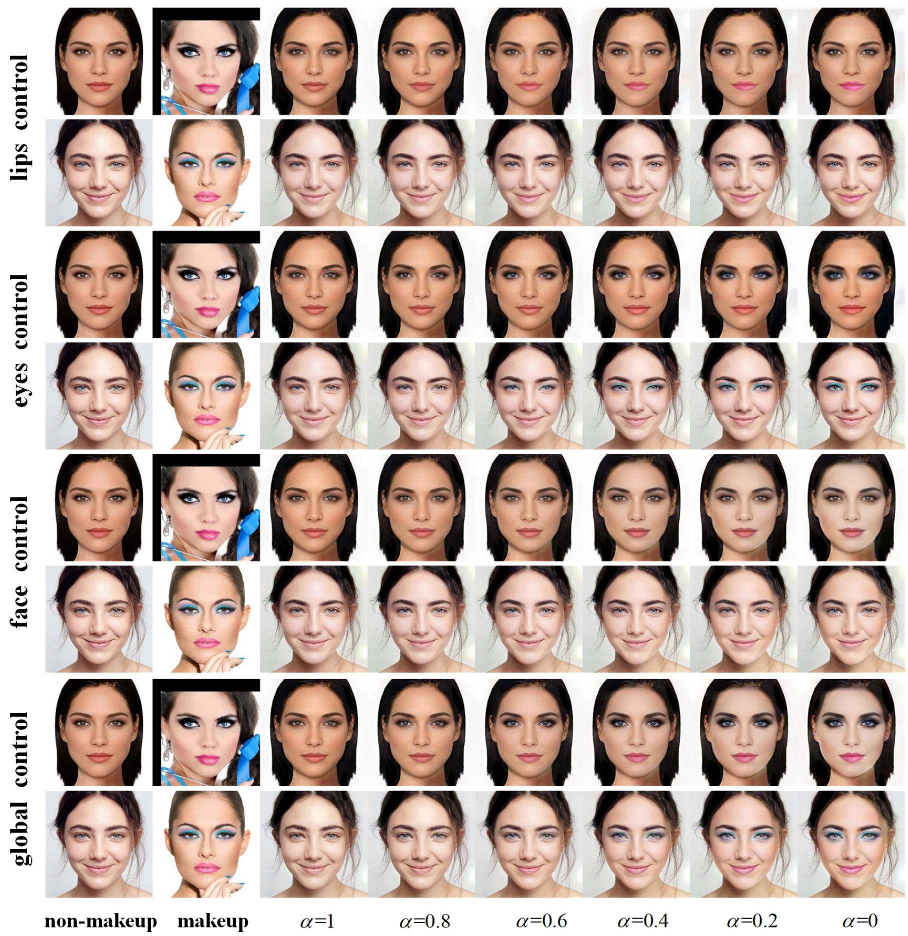

Fig. 8. The interpolated results. The first and second rows are the interpolated results of lips makeup style, we fix $\alpha_S = 1, \alpha_F = 1$, and gradually change α_L from 1 to 0. The third and fourth rows are the interpolated results of eyes makeup style, we fix $\alpha_L = 1, \alpha_F = 1$, and gradually change α_S from 1 to 0. The fifth and sixth rows are the interpolated results of face makeup style, we fix $\alpha_L = 1, \alpha_S = 1$, and gradually change α_F from 1 to 0. In last two rows, we gradually change $\alpha_L, \alpha_S, \alpha_F$ from 1 to 0.

Fig. 9. 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.

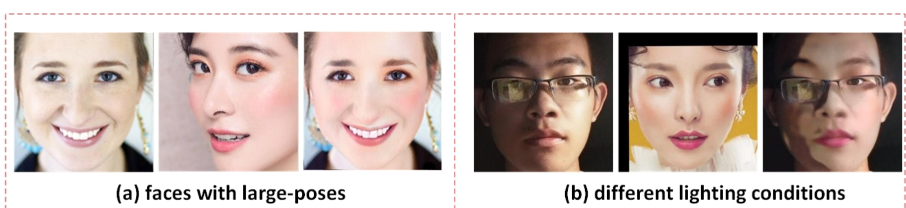

Fig. 10. The results of our approach under large-poses and different lighting conditions.

References

1. 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)
2. Chang, H., Lu, J., Yu, F., Finkelstein, A.: Pairedcyclegan: Asymmetric style transfer for applying and removing makeup. In: CVPR (2018)
3. 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)
4. 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)
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. Zhang, H., Chen, W., He, H., Jin, Y.: Disentangled makeup transfer with generative adversarial network. arXiv preprint arXiv:1907.01144 (2019)
7. Yi, Z., Zhang, H., Tan, P., Gong, M.: Dualgan: Unsupervised dual learning for image-to-image translation. In: ICCV (2017)
8. Huang, X., Liu, M.Y., Belongie, S.J., Kautz, J.: Multimodal unsupervised image-to-image translation. In: ECCV (2018)
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. Ma, L., Sun, Q., Georgoulis, S., Gool, L.V., Schiele, B., Fritz, M.: Disentangled person image generation. In: CVPR (2018)
11. Lorenz, D., Bereska, L., Milbich, T., Ommer, B.: Unsupervised part-based disentangling of object shape and appearance. In: CVPR (2019)
12. Esser, P., Haux, J., Ommer, B.: Unsupervised robust disentangling of latent characteristics for image synthesis. In: ICCV (2019)
13. 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)
14. Guo, D., Sim, T.: Digital face makeup by example. In: CVPR (2009)
15. Li, C., Zhou, K., Lin, S.: Simulating makeup through physics-based manipulation of intrinsic image layers. In: CVPR (2015)
16. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: CVPR (2016)
17. Liu, S., Ou, X., Qian, R., Wang, W., Cao, X.: Makeup like a superstar: Deep localized makeup transfer network. In: IJCAI (2016)
18. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: ICCV (2017)
19. 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)
20. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
21. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI (2015)
22. Li, C., Wand, M.: Precomputed real-time texture synthesis with markovian generative adversarial networks. In: ECCV (2016)