ISSN 2464-4617 (print)
ISSN 2464-4625 (DVD)
Computer Science Research Notes
CSRN 3201
WSCG 2022 Proceedings

Supervised Learning for Makeup Style Transfer

Natalia Strawa
Warsaw University of Technology
Faculty of Electrical Engineering
ul. Koszykowa 75
00-662 Warsaw, Poland
natalia.strawa2.stud@pw.edu.pl

Grzegorz Sarwas
Warsaw University of Technology
Faculty of Electrical Engineering
ul. Koszykowa 75
00-662 Warsaw, Poland
grzegorz.sarwas@pw.edu.pl

Abstract

This paper addresses the problem of using deep learning for makeup style transfer. For solving this problem, we propose a new supervised method. Additionally, we present a technique for creating a synthetic dataset for makeup transfer used to train our model. The obtained results were compared with six popular methods for makeup transfer using three metrics. The tests were carried out on four available data sets.

The proposed method, in many respects, is competitive with the methods used in the literature. Thanks to images of faces with generated synthetic makeup, the proposed method learns to better transfer details, and the learning process is significantly accelerated.

Keywords

Makeup Transfer, Image Style Transfer, CycleGAN, GAN, Image Processing, Deep Learning

1 INTRODUCTION

The development of generative adversarial network (GAN) architectures triggers the proposal of many practical solutions to help us in life. We can observe the increased popularity of applications for virtualizing showrooms. Virtual trying on clothes, glasses, etc., became a popular e-commerce solution. Among these apps, we can find technology for makeup transfer, which refers to transferring a reference makeup to a face without makeup and maintaining the original appearance of the plain face and the makeup style of the reference face.

Makeup transfer entails many difficulties and challenges. This process can be divided into two main steps. The first one is responsible for extracting the makeup from the face with the makeup pattern we would like to transfer. Meanwhile, the second is connected with makeup applying.

In the first step, the most challenging task is properly separating the color of the foundation on the skin. The complete transfer of one person's skin color to another person's face is undesirable, especially in the case of people who are different in complexion. The same is true of other elements that may sometimes appear in the face photo. Examples of such details that should not be transferred are freckles, wrinkles, discoloration, and pieces that obscure the face in the image.

The second stage, applying the extracted makeup to the other face, is equally non-trivial. Usually, two people have different face shapes, so the makeup needs to be fitted to the face we want to transfer it. In addition, the photos may show differences in position and facial expressions. For example, only half of the makeup will be visible on a face positioned in profile. Still, the method should consider that the makeup is usually symmetrical and transfer the complete makeup to the target face. Similarly, the unusual facial expression should not disturb the algorithm.

To solve these problems, many makeup transfer techniques were developed [Ma21]. These methods can be categorized into two main groups: traditional makeup transfer [Ton07; Guo09; Sch11] and makeup transfer based on deep learning [Liu16; Joh16; Lia17; Li18; Che19].

The main contributions of this work are: (1) We propose an algorithm for generating synthetic makeup useful for creating synthetic dataset; (2) We propose a new, competitive supervised makeup transfer method. The conducted experiment confirmed that our solution is better at transferring makeup details, and the learning process was significantly accelerated.

2 RELATED WORK

Most of the traditional makeup transfer methods have high requirements regarding the reference image and the target image [Ma21]. To do this process successfully, pose, and light conditions must be similar in both images, which is very hard to achieve. Among these solutions, we can find algorithms based on supervised [Ton07] and unsupervised learning [Guo09]. For the training process, supervised makeup transfer methods require a dataset with a target image and pair of reference images before makeup transfer and after. These algorithms are usually based on three main steps. First, calculate the color and lighting changes of the image before and after applying makeup. Then modify the skin texture and the color difference between the reference and target surfaces. Finally, transfer the makeup and adjust the makeup style to match the target face. Since the makeup transfer requires the transfer of many different elements, hence manual processing needs an extensive sequence of operations, such as, e.g., Bayesian matting [Chu01], graph cutting texture synthesis algorithm [Kwa03], Independent Component Correlation Algorithm (ICA) [Tsu03].

With some classic makeup methods, the makeup is transferred pixel-by-pixel, which is prone to the slightest facial shifts in the photo [Ton07]. Another possible approach is to use a 3D model of facial deformation, making it easier to target the right pixels [Bla99].

Traditional unsupervised makeup transfer methods use the distribution of the image in CIELAB color spaces. They then use the WLS algorithm or bilateral filtering method [Tom98] to perform edge smoothing and to smooth out the brightness layer to obtain the face structure layer [Ma21].

Whether it is a traditional makeup transfer method based on a supervised model or an unsupervised model, the pose and illumination requirements of the input image are relatively high [Ma21]. These disadvantages were eliminated by using deep learning technology. Deep neural network architectures allow achieving more realistic results. We can find solutions based on pixel iteration ([Liu16; Xu13; Gat16]) and model iteration methods based on GAN [Goo20] or Glow [Kin18]. Among the second and third types, we can distinguish the following methods [Li18; Cha18; Jia20; Che19].

The CycleGAN model [Zhu17] can be seen as a fusion of two GANs. This model can apply makeup without face makeup and remove makeup from the reference face but can only do general makeup transfer, and the quality of the generated image is not very high. Pix2pixHD [Wan18] uses the multi-scale cGAN structure [Mir14] for image transformation. The StarGAN model [Cho18] preserves more facial features, provides better image quality, and provides better transfer results compared to CycleGAN and cGAN by mapping across multiple domains using only a pair of generators and discriminators and effectively training images. In BeautyGAN [Li18] the discriminator distinguishes the generated image from the real samples of the domain. Based on the transfer of a set of domains, it uses pixels based on different areas of the face. Instance-level migration is achieved with the help of a level histogram loss. The preservation of facial integrity and elimination of artifacts is achieved by adding to the perceptual and cyclic consistency loss of the overall objective loss function. This model can transfer makeup images but not instance-level makeup images. Jiang et al. proposed the PSGAN [Jia20] to solve the problem of the difference between a reference face and a face without makeup. This model uses a bottleneck encoder in the generator structure in StarGAN to extract facial features and then uses the attention mechanism to adaptively modify the makeup matrix. The FSGAN [Nir19] assesses the occlusion area by combining facial segmentation. The SCGAN [Den21] breaks down the makeup transfer problem into two stages: extraction and allocation. The part-specific style encoder extracts features of each part and maps them into a disentangled style latent space, while the face identity encoder extracts the facial identity features of the target image. The makeup fusion is done by a decoder that combines the style code with the facial identity characteristics. [Ngu21a] proposed to build a unified template that can adjust the 3D head position, face shape, and facial expressions of the source and target images with the makeup transferring based on BeautyGAN method. They also proposed to use the UV texture map instead of the original image to replace the makeup.

The flow-based generation model was noticed after the publication of the Glow article [Kin18]. In the case of makeup transfer, this model does not require the training of two large networks of discriminators and generators, and the time of automatic synthesis of results is very short.

The Glow model introduces a reversible convolution based on RealNVP [Din17] and simplifies some of its components. An example of the Glow model for makeup transfer is BeautyGlow method [Che19]. It uses the latent space of the input image (the makeup reference image without the target image) and decomposes the latent space according to the facial features and makeup features, respectively. Finally, the reference image makeup features and the target image's facial features are added to get the target image's latent space with makeup. The Glow model is used to reverse and transform it into the target RGB image with makeup.

In this paper, we propose a supervised learning algorithm. To create this model, we prepared architecture for generating images with synthetic makeup used in the training process.

3 PROPOSED SOLUTION

3.1 Formulation

We consider two image domains, non-makeup image domain denoted as $N \subset \mathbb{R}^{H\times W\times 3}$ and makeup image domain denoted as $M \subset \mathbb{R}^{H\times W\times 3}$. Our goal is to learn the mapping function between these domains, denoted as $G : \{n_{src}, m_{ref}\} \rightarrow \{m^G_{src}, n^G_{ref}\}$. This means that given two input images: source image n_{src} and reference image m_{ref}, the network is expected to generate makeup transfer result m^G_{src} and makeup removal result n^G_{ref}. The first one receives makeup style from the reference image and preserves the facial features of the source image, while the second one has makeup removed from the reference image.

3.2 Dataset generation

The generator is trained in a supervised manner. Unfortunately, among the available datasets for makeup transfer, none contains pairs of before and after makeup images. We introduce a new dataset generated using an algorithm for synthetic makeup application to address this issue. The algorithm uses two available models for landmark detection: Dlib [Kaz14] and Mediapipe [Kar]. It can be used to apply eyeliner, lipstick, eye shadows, and blushes. The simplified diagram illustrating the creation of the specific makeup elements can be found in Fig. 1.

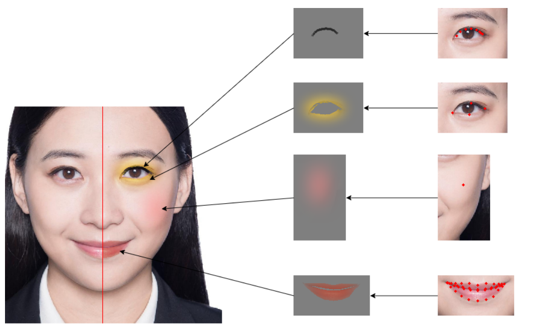

Figure 1: Dataset generation.

In the case of eyeliner, one can choose the position of lines on the eyelids (e.g., only on the upper eyelids, only on lower eyelids, or on both eyelids), their thickness, and transparency. When applying lipstick, the color, as well as its intensity and transparency, can be chosen. Eye shadows are determined by an ellipse with a center located approximately in the center of the eye and axes with lengths approximately equal to the width and height of the eye. The position of the center of the ellipse and the length of the axis can be modified by adding or subtracting from the initial values. This results in changing the shape of the shadows applied to the eyelids. The shape can also be controlled by changing the ellipse rotation angle. Besides, the color, transparency, and blur can also be modified. By shifting the center of the ellipse, it is possible to apply eyeshadows that appear only on the upper eyelid, only on the lower eyelid, or on both of them. Various unique shapes can be generated by modifying the ellipse's axis length and rotation angle. Blushes are created in a very similar way to eyeshadows. The ellipse center can be located at one of three points on the cheek. As with eyeshadows, it is possible to control the length of the ellipse axis, as well as the color and its parameters.

3.3 Dataset

To create the dataset, the parameters of the previously mentioned algorithm were randomized to make the makeups look natural but still have some diversity. For the generation of the synthetic dataset, non-makeup images from the Makeup Transfer [Li18] dataset were used. From this subset, 5000 pairs of images were sampled, and the same makeup style was applied to both images in every pair. One image from the pair represents the reference image, while the other represents the source image with the expected makeup style transfer. In total, 10000 images were created with synthetically generated makeup. The exemplary generated images are shown in Fig. 2.

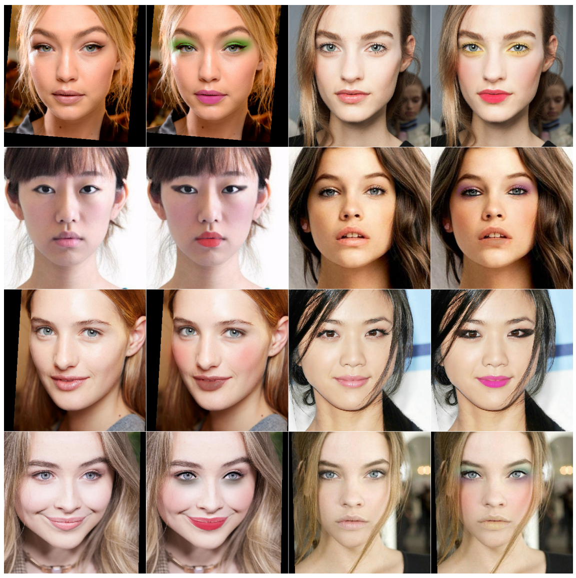

Figure 2: Examples of images from the newly generated dataset.

3.4 Framework

In our proposed method, we assume the training of four networks:

- Generator G,
- Discriminator D_N,
- Discriminator D_M,
- Discriminator D_S.

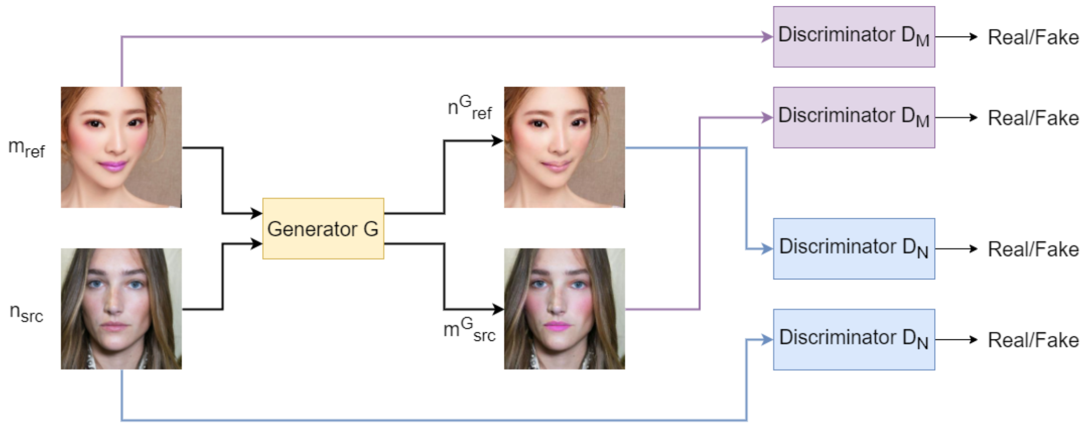

Figure 3: Framework. Domain-Level Makeup Transfer.

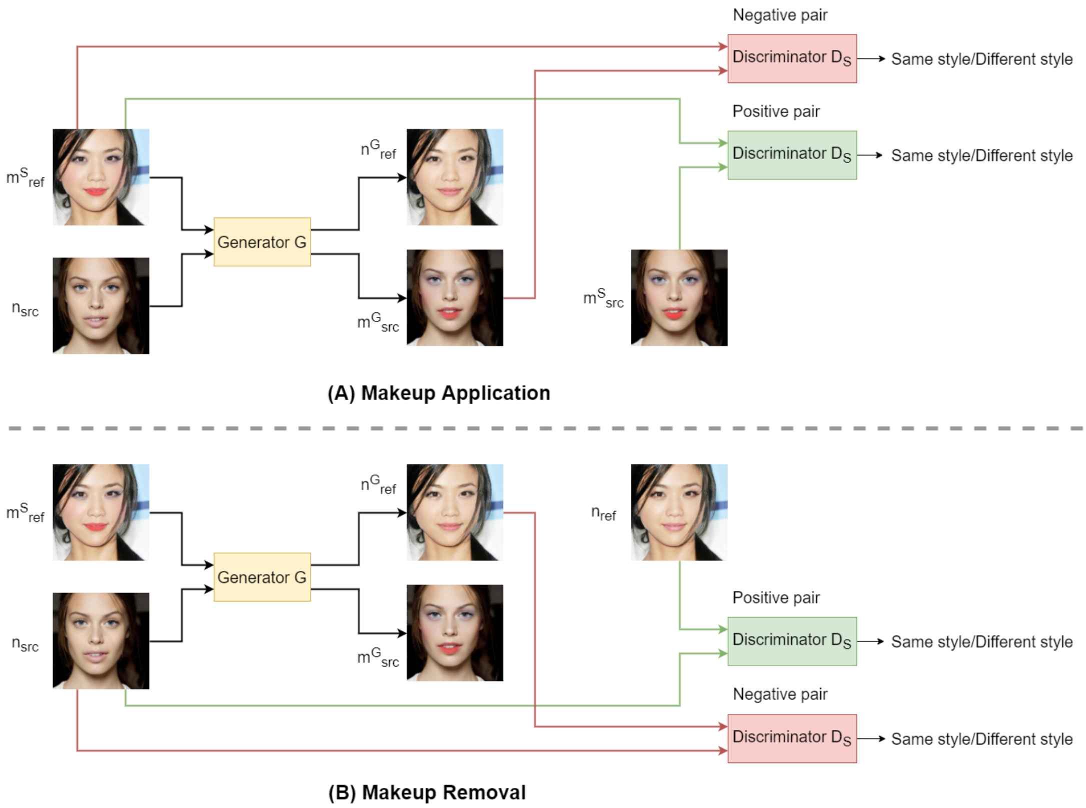

Figure 4: Framework. Instance-Level Makeup Transfer.

Generator takes as input a pair of images: source image n_{src} and reference image m_{ref} and generates two RGB masks R_1 for removing makeup and R_2 for applying makeup. In addition, weights W_1 and W_2 with values in the range $[0,1]$ are generated for each mask to determine their transparency in various areas. The parameters of the generator layers, such as filter size and stride, were derived from BeautyGAN. Additionally, as in PairedCycleGAN, we used dilated residual blocks. The discriminators D_N and D_M learn to determine the probability of whether the images belong to their corresponding domain or not, as shown in Fig. 3. Inspired by PairedCycleGAN [Cha18] we train an additional style discriminator D_S as shown in Fig. 4. It takes a pair of images as input and learns to determine whether they contain the same makeup style. The same discriminator is used to train the generator in the makeup transfer and removal process. Due to the lack of paired images with faces before and after makeup, a new dataset with synthetically generated makeup was created. In the case of makeup transfer, the positive pair were taken by the discriminator D_S consisting of a reference image m^S_{ref} and a source image with the same makeup m^S_{src}. The negative pair consists of the reference image m^S_{ref} and the makeup transfer result m^G_{src}. In the case of makeup removal, the positive pair includes the source image n_{src} and the reference image without makeup n_{ref}, and

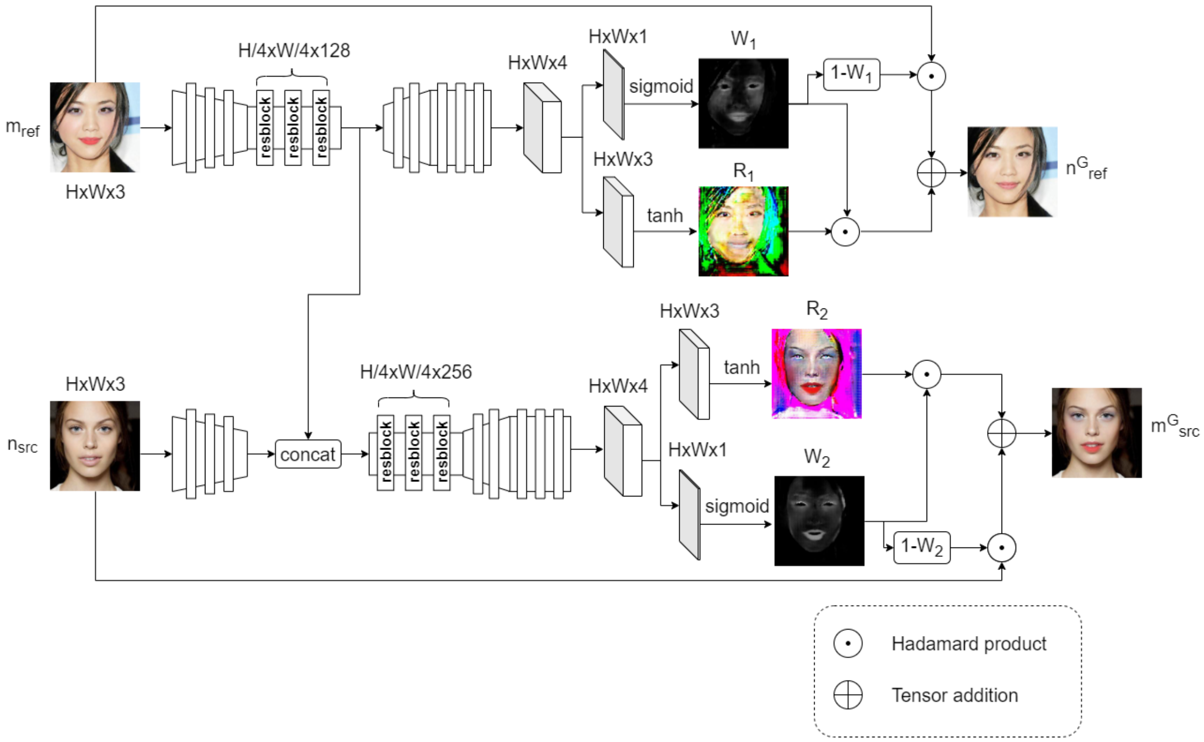

Figure 5: Architecture of generator.

the negative pair includes the source image n_{src} and the makeup removal result n^G_{ref}. The superscript ^{S} denotes the image with synthetically generated makeup, while the superscript ^{G} indicates the image generated by the generator.

Generator

The architecture of the proposed generator is shown in Fig. 5. In the beginning, two input images n_{src} and m_{ref} are passed into several downsampling convolutional layers of the two separate branches. The feature maps extracted from the makeup image are then forwarded to several residual blocks. Then the extracted feature maps are concatenated with the feature maps extracted from the non-makeup image and passed into several residual blocks. Finally, the feature maps pass through several upsampling convolutional layers, and an image with the same size as the input image and with four channels is returned. After applying the tanh activation function, the first three channels create an RGB mask for makeup transfer. The last channel represents the weights for the mask. A sigmoid activation function was used for the weights to be in the range $[0,1]$. The generation of the resulting image M^G_{src} can be written as follows:

$$m^G_{src} = n_{src} \odot (1 - W_2) + R_2 \odot W_2, \tag{1}$$

where $\odot$ denotes the Hadamard product.

The same applies to the makeup image. The output feature maps from the residual blocks are passed to several upsampling convolutional layers, and then an RGB mask and weights for makeup removal are created in the same way as above. The resulting image is created as follows:

$$n^G_{ref} = m_{ref} \odot (1 - W_1) + R_1 \odot W_1, \tag{2}$$

where $\odot$ denotes the Hadamard product.

3.5 Objective Function

Adversarial loss

The generator is guided by adversarial loss to provide more realistic results. We employed two discriminators, D_N and D_M, to distinguish the generated samples from real samples in domains N and M, respectively. Adversarial losses for discriminators D_N and D_M are defined as follows:

$$\begin{aligned} L_{D_N} =& \mathbb{E}_{n_{src}}[(D_N(n_{src}) - 1)^2] \\ &+ \mathbb{E}_{n_{src},m_{ref}}[D_N(n^G_{ref})^2], \end{aligned} \tag{3}$$

$$\begin{aligned} L_{D_M} =& \mathbb{E}_{m_{ref}}[(D_M(m_{ref}) - 1)^2] \\ &+ \mathbb{E}_{n_{src},m_{ref}}[D_M(m^G_{src})^2], \end{aligned} \tag{4}$$

where $\mathbb{E}$ is an expected value.

In order to train the generator to transfer a specific makeup, we introduced an additional style discriminator D_S to determine whether the same makeup is present in two images. Adversarial loss for this discriminator is defined as:

$$\begin{aligned} L_{D_S} =& \mathbb{E}_{n_{src}}[(D_S(n_{src}, n_{ref}) - 1)^2] \\ &+ \mathbb{E}_{n_{src}, m_{ref}}[D_S(n_{ref}^G)^2] \\ &+ \mathbb{E}_{m_{ref}}[(D_S(m_{ref}, m_{src}^S) - 1)^2] \\ &+ \mathbb{E}_{n_{src}, m_{ref}}[D_S(m_{src}^G)^2]. \end{aligned} \tag{5}$$

The adversarial loss for the generator is defined as follows:

$$\begin{aligned} L_{adv} =& \mathbb{E}_{n_{src}, m_{ref}}[(D_N(n_{ref}^G) - 1)^2] \\ &+ \mathbb{E}_{n_{src}, m_{ref}}[(D_M(m_{src}^G) - 1)^2] \\ &+ \mathbb{E}_{n_{src}, m_{ref}}[(D_S(n_{ref}^G) - 1)^2] \\ &+ \mathbb{E}_{n_{src}, m_{ref}}[(D_S(m_{src}^G) - 1)^2]. \end{aligned} \tag{6}$$

Mask loss

The mask loss L_{mask} is used to reduce the weights of the mask in areas of the image that should be unaffected. This includes the background, eyes, teeth, ears, hair, and neck. The mentioned loss can be expressed in the following way:

$$L_{mask} = ||W_{1background}||_1 + ||W_{2background}||_1, \tag{7}$$

where

$$W_{1background} = W_1 \odot L_{1background},$$

$$W_{2background} = W_2 \odot L_{2background},$$

and $\odot$ denotes the Hadamard product, $L_{1background}$ and $L_{2background}$ are the binary masks specifying the aforementioned background elements.

The full objective function of generator G contains two types of losses: adversarial loss and mask loss

$$L_G = L_{adv} + L_{mask}. \tag{8}$$

4 EXPERIMENTS

4.1 Training details

The generator convolutional layer parameters such as filter size and stride were derived from BeautyGAN [Li18]. However, the network structure has been slightly modified. At the same time, the normalization of weights was changed from Instance Norm to Batch Norm. Additionally, inspired by the PairedCycleGAN [Cha18], we also applied dilated residual blocks to the generator. The architecture of discriminators D_N and D_M was taken entirely from the BeautyGAN [Li18]. However, the discriminator D_S was modified to accept two images, which were then concatenated. The discriminator D_B was trained on a dataset containing real makeups, the style discriminator D_S was trained on the synthetically generated dataset, while the generator G was trained indirectly on both datasets.

For all experiments, the images have been resized to 256×256. We train the model for 25 epochs optimized by Adam [Kin15] with a learning rate of 0.0002 and a batch size of 4.

From initial experiments, it appears that our method learns faster than other methods. On the AMD Ryzen 1920X with NVidia RTX 2080Ti, it took about 4 hours to train. In comparison, the learning time of different methods is usually counted in tens of hours. E.g., implementation of BeautyGAN needs about one week with RTX 1080Ti. One branch of CPM consists of a generator from BeautyGAN, so this method requires at least as much time as BeautyGAN. The authors of BeautyGlow indicated that fine-tune Glow took 3 days. Unfortunately, the model of GPU used for training is unknown.

4.2 Comparisons to Baselines

We compared our method with six state-of-the-art methods for makeup transfer: BeautyGAN [Li18], BeautyGlow [Che19], CPM [Ngu21b], LADN [Gu19], PSGAN [Jia20] and SCGAN [Den21]. We used our implementation of the BeautyGlow model for testing, so the results may differ from those that the original model would have returned. Also, we used an available online implementation of the BiSeNet [Yu18] model to create segmentation masks for SCGAN so that the lower quality results may be the result of non-ideal masks.

4.2.1 Used Datasets

We performed tests on four available datasets: Makeup Transfer [Li18], Makeup Wild [Jia20], CPM-Synt-2 [Ngu21b], and a dataset shared by the authors of LADN [Gu19]. From the first three datasets, 2000 unique pairs consisting of a source and reference image were sampled. Later, 2000 examples were generated from the sampled pairs using each method. Based on the CPM-Synt-2 dataset, 1115 examples were generated. The CPM-Synt-2 dataset is the only one that contains pairs of before and after makeup images. First, pairs of pictures without makeup were sampled from the Makeup Transfer dataset to create the dataset. Then the same makeup was applied to both photos using BeautyGAN. The result was two images, where one was the reference image, and the other was the source image with the expected makeup.

4.2.2 Qualitative Comparison

The results of the visual comparison are shown in Fig. 6. BeautyGAN, PSGAN, and SCGAN significantly transfer skin color and shadows from the reference image to the source image. These methods

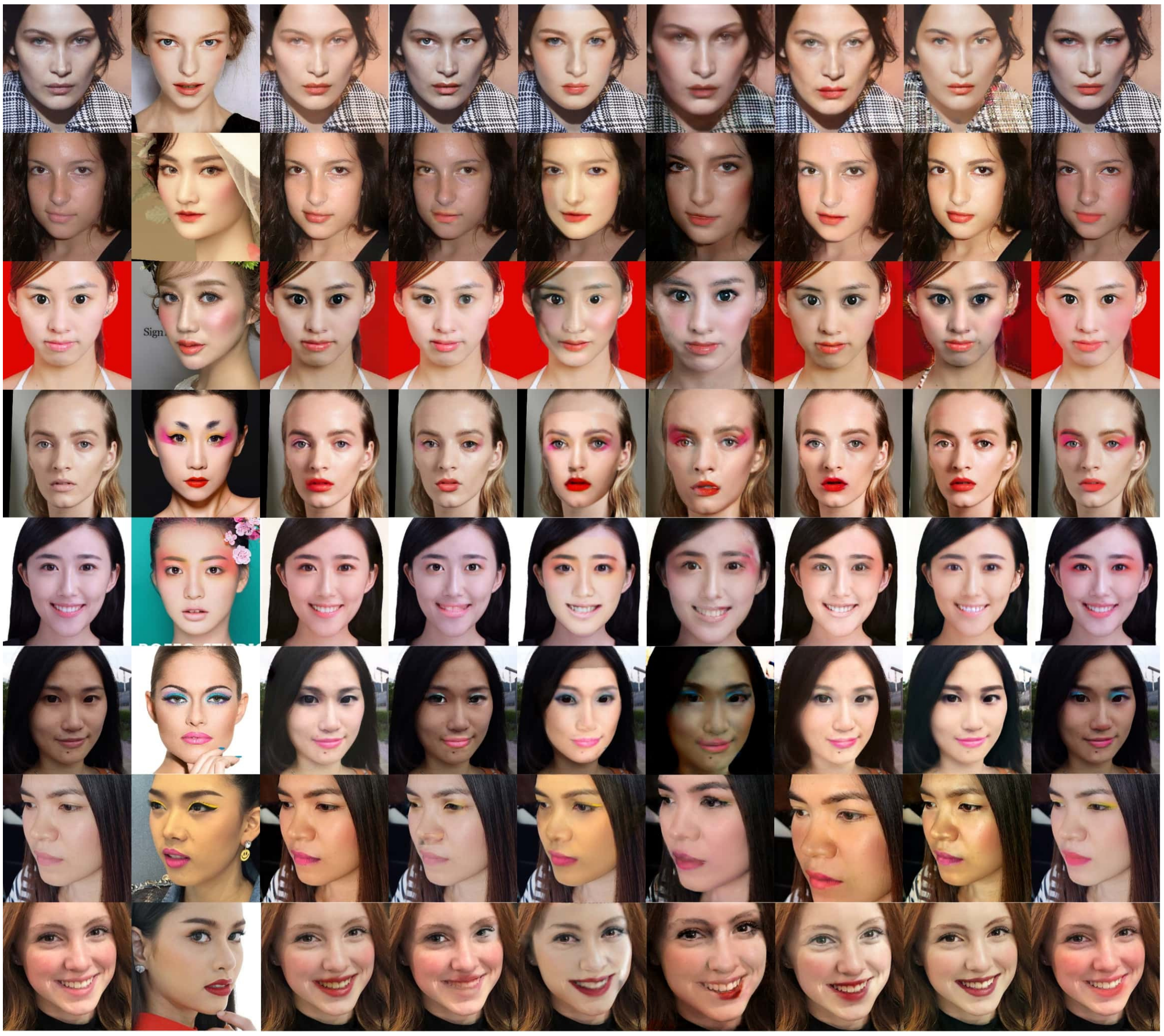

Figure 6: Comparison with state-of-the-art methods. First row: example from the CPM-Synt-2 [Ngu21b] dataset. Rows 2 and 3: examples from the Makeup Transfer [Li18] dataset. Rows 4, 5, and 6: examples from the dataset provided by the authors of the LADN [Gu19] method. Last two rows: examples from the Makeup Wild [Jia20] dataset.

transfer lip makeup very well, blush slightly less well, and eye makeup the least well. When they transfer eyeshadows, they lack details and color is often missing or blurred. BeautyGlow does not transfer makeup very well. Additionally, it transfers facial features from the reference face to the source face. CPM usually transfers almost the entire face from the reference image to the source image. Additionally, it performs poorly with makeup transfer when there are significant pose differences between the faces and often produces artifacts in images. On the other hand, it transfers makeup details and colors much better than the previously mentioned methods. LADN, like CPM, transfers makeup details quite well; however, it significantly lowers the quality of the images. The proposed method does not transfer skin color or shadows. In addition, it transfers color and makeup details well without affecting image quality. The disadvantage of this method is that it does not always transfer makeup well when there are differences in facial poses between images, as can be seen in the last row in Fig. 6.

4.2.3 *Quantitative Comparison*

Evaluation of the quality of generated images is a complex problem. It is even harder to evaluate whether the identical make-up has been transferred. Because of that, there is no one best metric to assess the quality of the model.

In our evaluation we used three metrics: FID (Fréchet inception distance) [Heu17], PSNR (Peak Signal-to-Noise Ratio) [Pon11], MS-SSIM (Multi-Scale Structural similarity) [Wan03]. FID correlates with the qual-

Method/Dataset	FID ↓				MS-SSIM ↑	PSNR ↑
	CPM-Synt-2	LADN	Makeup Transfer	Makeup Wild		
BeautyGAN	5.404	75.254	50.467	89.408	0.988	31.531
BeautyGlow	12.73	68.46	54.598	93.13	0.921	20.662
CPM	17.073	54.181	41.691	79.124	0.841	20.617
LADN	32.818	59.102	45.988	101.872	0.874	17.576
PSGAN	12.616	66.178	41.69	91.952	0.935	23.863
SCGAN	17.182	64.61	36.429	80.081	0.953	25.021
Ours	10.743	73.286	51.717	90.823	0.938	20.790
Unmodified images	-	-	-	-	0.944	20.944

Table 1: FID scores for the four datasets and MS-SSIM and PSNR scores for the CPM-Synt-2 dataset.

ity of the image generated but does not account for the transfer quality. The two other metrics evaluate how well the model transfers makeup from reference to the source image (based on labels from BeautyGAN as described in Section 4.2.1).

The FID score was calculated for all datasets. However, the other metrics were only calculated for the CPM-Synt-2 dataset because it was the only one labeled. Table 1 shows the values of the metrics.

The metrics results do not fully capture the quality of the makeup transfer for two reasons. First, some of the methods tested were trained on the datasets used for testing. The split between the training and testing datasets was unknown. Therefore, some results may be slightly biased. Second, the metric values do not fully correlate with the visual results, as seen in the CPM example. For several datasets, this method obtains the best FID metric scores. However, in Fig. 6, it can be seen that the results it produces often have visible artifacts. Since the scores obtained by our method are not significantly different from the scores obtained by other methods, and the visual results are similar, it can be concluded that this method is comparable to state-of-the-art methods.

5 CONCLUSION

In this paper, we propose a new supervised method for makeup transfer. Using a newly created dataset containing pairs of before and after makeup images, we were able to simplify the solution's architecture and accelerate the learning process. Methods using warping or histogram matching guide the generator towards results that are, by definition, not optimal. The proposed method does not suffer from such a problem, and the only thing that limits it is the quality and variety of makeup generated by the algorithm.

However, such an algorithm can be further developed and enriched with new types of makeup, which cannot be said about the other methods of this type. The new dataset helps better transfer makeup details such as eyeshadows and blush. One limitation of our approach is that it performs poorly with significant pose differences between faces. The presented visual and quantitative comparison shows that the proposed method is competitive with state-of-the-art methods for makeup transfer.

REFERENCES

[Bla99] V. Blanz and T. Vetter. “A Morphable Model for the Synthesis of 3D Faces”. In: *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH ’99. USA: ACM Press/Addison-Wesley Publishing Co., 1999, pp. 187–194. ISBN: 0201485605. DOI: 10 . 1145 / 311535 . 311556. URL: https://doi.org/10.1145/311535.311556.

[Cha18] H. Chang et al. “PairedCycleGAN: Asymmetric Style Transfer for Applying and Removing Makeup”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 40–48. DOI: 10.1109/CVPR.2018.00012.

[Che19] H-J Chen et al. “BeautyGlow: On-Demand Makeup Transfer Framework With Reversible Generative Network”. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 10034–10042. DOI: 10.1109/CVPR.2019.01028.

[Cho18] Y. Choi et al. “StarGAN: Unified Generative Adversarial Networks for Multi-domain Image-to-Image Translation”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 8789–8797. DOI: 10.1109/CVPR.2018.00916.

[Chu01] Y-Y Chuang et al. “A Bayesian approach to digital matting”. In: *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*. Vol. 2. 2001, pp. II–II. DOI: 10.1109/CVPR.2001.990970.

[Den21] H. Deng et al. "Spatially-Invariant Style-Codes Controlled Makeup Transfer". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2021, pp. 6549–6557.

[Din17] L. Dinh, J. S-D, and S. Bengio. "Density estimation using Real NVP". In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. 2017.

[Gat16] L. A. Gatys, A. S. Ecker, and M. Bethge. "Image Style Transfer Using Convolutional Neural Networks". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 2414–2423. DOI: 10.1109/CVPR.2016.265.

[Goo20] I. Goodfellow et al. "Generative Adversarial Networks". In: *Commun. ACM* 63.11 (Oct. 2020), pp. 139–144. ISSN: 0001-0782. DOI: 10.1145/3422622. URL: https://doi.org/10.1145/3422622.

[Gu19] Q. Gu et al. "LADN: Local Adversarial Disentangling Network for Facial Makeup and De-Makeup". In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 10480–10489. DOI: 10.1109/ICCV.2019.01058.

[Guo09] D. Guo and T. Sim. "Digital face makeup by example". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 73–79. DOI: 10.1109/CVPR.2009.5206833.

[Heu17] M. Heusel et al. "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/8a1d694707eb0fefe65871369074926d-Paper.pdf.

[Jia20] W. Jiang et al. "PSGAN: Pose and Expression Robust Spatial-Aware GAN for Customizable Makeup Transfer". In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 5193–5201. DOI: 10.1109/CVPR42600.2020.00524.

[Joh16] J. Johnson, A. Alahi, and L. Fei-Fei. "Perceptual Losses for Real-Time Style Transfer and Super-Resolution". In: *Computer Vision – ECCV 2016*. Ed. by B. Leibe et al. Cham: Springer International Publishing, 2016, pp. 694–711. ISBN: 978-3-319-46475-6.

[Kar] Y Kartynnik et al. "Real-time facial surface geometry from monocular video on mobile GPUs. arXiv 2019". In: *arXiv preprint arXiv:1907.06724* ().

[Kaz14] Vahid Kazemi and Josephine Sullivan. "One millisecond face alignment with an ensemble of regression trees". In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 1867–1874. DOI: 10.1109/CVPR.2014.241.

[Kin15] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015. URL: http://arxiv.org/abs/1412.6980.

[Kin18] D. P. Kingma and P. Dhariwal. "Glow: Generative Flow with Invertible 1x1 Convolutions". In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio et al. Vol. 31. Curran Associates, Inc., 2018.

[Kwa03] V. Kwatra et al. "Graphcut Textures: Image and Video Synthesis Using Graph Cuts". In: *ACM Trans. Graph.* 22.3 (July 2003), pp. 277–286. ISSN: 0730-0301. DOI: 10.1145/882262.882264. URL: https://doi.org/10.1145/882262.882264.

[Li18] T. Li et al. "BeautyGAN: Instance-Level Facial Makeup Transfer with Deep Generative Adversarial Network". In: *Proceedings of the 26th ACM International Conference on Multimedia*. MM '18. Seoul, Republic of Korea: Association for Computing Machinery, 2018, pp. 645–653. ISBN: 9781450356657. DOI: 10.1145/3240508.3240618. URL: https://doi.org/10.1145/3240508.3240618.

[Lia17] J. Liao et al. "Visual Attribute Transfer through Deep Image Analogy". In: *ACM Trans. Graph.* 36.4 (July 2017). ISSN: 0730-0301. DOI: 10.1145/3072959.

3073683. URL: https://doi.org/10.1145/3072959.3073683.

[Liu16] S. Liu et al. "Makeup like a Superstar: Deep Localized Makeup Transfer Network". In: *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*. IJCAI'16. New York, New York, USA: AAAI Press, 2016, pp. 2568–2575. ISBN: 9781577357704.

[Ma21] X. Ma et al. "Deep learning method for makeup style transfer: A survey". In: *Cognitive Robotics* 1 (2021), pp. 182–187. ISSN: 2667-2413. DOI: https://doi.org/10.1016/j.cogr.2021.09.001. URL: https://www.sciencedirect.com/science/article/pii/S266724132100015X.

[Mir14] M. Mirza and S. Osindero. "Conditional Generative Adversarial Nets". In: *CoRR* abs/1411.1784 (2014). arXiv: 1411.1784. URL: http://arxiv.org/abs/1411.1784.

[Ngu21a] T. Nguyen, A. T. Tran, and M. Hoai. "Lipstick ain't enough: Beyond Color Matching for In-the-Wild Makeup Transfer". In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 13300–13309. DOI: 10.1109/CVPR46437.2021.01310.

[Ngu21b] T. Nguyen, A. T. Tran, and M. Hoai. "Lipstick ain't enough: Beyond Color Matching for In-the-Wild Makeup Transfer". In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 13300–13309. DOI: 10.1109/CVPR46437.2021.01310.

[Nir19] Y. Nirkin, Y. Keller, and T. Hassner. "FSGAN: Subject Agnostic Face Swapping and Reenactment". In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 7183–7192. DOI: 10.1109/ICCV.2019.00728.

[Pon11] Nikolay Ponomarenko et al. "Modified image visual quality metrics for contrast change and mean shift accounting". In: *2011 11th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)*. 2011, pp. 305–311.

[Sch11] K. Scherbaum et al. "Computer-suggested Facial Makeup". In: *Comp. Graph. Forum (Proc. Eurographics 2011)* 30.2 (2011).

[Tom98] C. Tomasi and R. Manduchi. "Bilateral filtering for gray and color images". In: *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*. 1998, pp. 839–846. DOI: 10.1109/ICCV.1998.710815.

[Ton07] W-S Tong et al. "Example-Based Cosmetic Transfer". In: *15th Pacific Conference on Computer Graphics and Applications (PG'07)*. 2007, pp. 211–218. DOI: 10.1109/PG.2007.31.

[Tsu03] N. Tsumura et al. "Image-Based Skin Color and Texture Analysis/Synthesis by Extracting Hemoglobin and Melanin Information in the Skin". In: *ACM SIGGRAPH 2003 Papers*. SIGGRAPH '03. San Diego, California: Association for Computing Machinery, 2003, pp. 770–779. ISBN: 1581137095. DOI: 10.1145/1201775.882344. URL: https://doi.org/10.1145/1201775.882344.

[Wan03] Z. Wang, E.P. Simoncelli, and A.C. Bovik. "Multiscale structural similarity for image quality assessment". In: *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*. Vol. 2. 2003, 1398–1402 Vol.2. DOI: 10.1109/ACSSC.2003.1292216.

[Wan18] T-C Wang et al. "High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 8798–8807. DOI: 10.1109/CVPR.2018.00917.

[Xu13] L. Xu, Y. Du, and Y. Zhang. "An automatic framework for example-based virtual makeup". In: *2013 IEEE International Conference on Image Processing*. 2013, pp. 3206–3210. DOI: 10.1109/ICIP.2013.6738660.

[Yu18] Changqian Yu et al. "Bisenet: Bilateral segmentation network for real-time semantic segmentation". In: *European Conference on Computer Vision*. Springer. 2018, pp. 334–349.

[Zhu17] J-Y Zhu et al. "Unpaired Image-To-Image Translation Using Cycle-Consistent Adversarial Networks". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Oct. 2017, pp. 2242–2251. DOI: 10.1109/ICCV.2017.244.