BeautyGlow: On-Demand Makeup Transfer Framework with Reversible Generative Network

Hung-Jen Chen, Ka-Ming Hui, Szu-Yu Wang, Li-Wu Tsao, Hong-Han Shuai, Wen-Huang Cheng
National Chiao Tung University, Taiwan
{hjc.eed07g,ming2242.cm06g,joey840404.eed03,leo159753.me05,hhshuai,whcheng}@nctu.edu.tw

Abstract

As makeup has been widely-adopted for beautification, finding suitable makeup by virtual makeup applications becomes popular. Therefore, a recent line of studies proposes to transfer the makeup from a given reference makeup image to the source non-makeup one. However, it is still challenging due to the massive number of makeup combinations. To facilitate on-demand makeup transfer, in this work, we propose BeautyGlow that decompose the latent vectors of face images derived from the Glow model into makeup and non-makeup latent vectors. Since there is no paired dataset, we formulate a new loss function to guide the decomposition. Afterward, the non-makeup latent vector of a source image and makeup latent vector of a reference image and are effectively combined and revert back to the image domain to derive the results. Experimental results show that the transfer quality of BeautyGlow is comparable to the state-of-the-art methods, while the unique ability to manipulate latent vectors allows BeautyGlow to realize on-demand makeup transfer.

1. Introduction

As the slogan says, “Life isn’t perfect, but your makeup can be.” Nowadays, makeup provides a convenient way to improve the facial appearance, e.g., powder foundations for hiding the skin imperfections, blushes for creating chubby cheeks. Although makeup is ubiquitous in daily life, it is not easy to find the best-suited makeup because 1) makeup trials are time-consuming and inconvenient, and 2) there are thousands and thousands of cosmetic products, varying from brands, functions, colors, and way-to-use, which leads to a more intractable number of the makeup style combinations. To address the first issue, virtual makeup applications that provide beautification function come up to facilitate the makeup trials, such as YoucamMakeup[1] and Meitu[2]. However, how to find suitable makeup for each user is still an open problem.

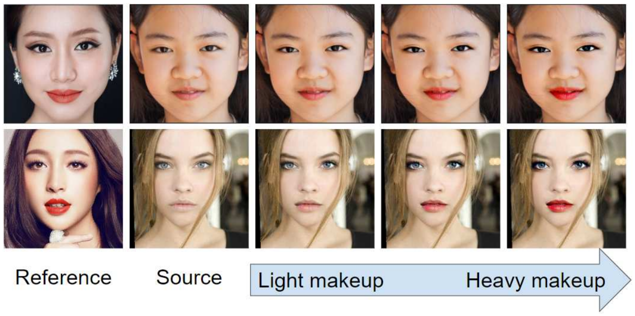


Figure 1: *The makeup features such as eyeshadows and lip gloss are extracted from reference makeup images and are transferred to source non-makeup images. The lightness of the makeup can be tuned by adjusting the magnification in the latent space.*

One of the possible solutions is to let users select the makeup styles from the reference photos of celebrities or friends and transfer the makeup to users’ faces. For example, Guo *et al.* [10] propose to decompose the images into three layers, i.e., face structure layer, skin detail layer, and color layer, and transfer the information from each layer of reference images to the corresponding layer of the target images. However, the predefined layers and transfer function are not data-driven and thus are inclined to generate artifacts in many cases. To directly learn from the data, a dataset of photo triplets (user face, reference makeup, and transferred results) is desirable for learning. Nevertheless, most of the existing datasets only provide the images with makeup and non-makeup pairs. To avoid the need for such triplet datasets, [23] proposes to use CNN for identifying different makeup functions and extracting the corresponding features. Afterward, different loss functions are defined for different parts of makeup to make the after-makeup images natural, e.g., foundations are transferred by regularizing the inner product of the feature maps for smoothing the

[1] https://www.perfectcorp.com/app/ymk
[2] http://makeup.meitu.com/en/

skin's texture. Although previous work generates good results on makeup transfer, the domain knowledge is required to design different functions to generate different makeup.

Recent years, Generative Adversarial Networks (GANs) are widely used for generating high-resolution realistic images. For style transfer on whole images (e.g., painting style), [30] proposes CycleGAN to incorporate the cycle consistency loss to train the mapping function between two domains by adopting two generators and discriminators. However, CycleGAN can only generate a general makeup style rather than a specific makeup style. Based on CylceGAN, Chang *et al.* [1] introduce an asymmetric function and train a makeup removal and transfer network together to preserve the face identity. Moreover, [19] utilizes the GAN framework and further proposes pixel-level histogram loss to maximize the similarity of makeup style, while perceptual loss and cycle consistency loss are designed to preserve identity. Nonetheless, there is no encoder in GAN-based methods and thus cannot adjust the makeup extent by interpolating the latent space. Adjusting the makeup from light to heavy is important for finding the best-suited makeup since users can obtain many candidates from one reference images.

A recent line of studies synthesizes images by flow-based generative models, of which the encoder function is reversible and supports different applications by manipulating the latent space and reserving it back to real image space [4, 15]. Due to the ability of manipulating latent space, in this paper, we propose an unsupervised on-demand makeup transfer approach, namely, BeautyGlow, based on the Glow architecture [15]. Specifically, Glow provides a general framework to learn a generative network with invertible functions that encode input images into a meaningful latent space and enable modification of existent data points. Although Glow allows manipulating the latent vectors, it is challenging to transfer the local makeup details of the reference image to the target image since the makeup and face are mixed as the latent vector. One of the possible approaches is to find the average latent vector of makeup images and average latent vector of non-makeup images, and then use the difference as the direction of manipulating. However, this approach contains two major issues: 1) it can only find the general makeup but not the user-specified makeup, and 2) it requires a lot of images from the same person or same makeup to find the correct average latent vector.

To address this issue, BeautyGlow first defines a transformation matrix that decomposes the latent vectors into latent vector of makeup features and latent vector of facial identity features. However, due to the lack of paired data, we further formulate a new loss function containing perceptual loss, makeup loss, intra-domain loss, inter-domain loss, and cycle consistency loss, to guide the decomposition. Compared with other methods based on GANs, BeautyGlow does not need to train two large networks, i.e., generator and discriminator, which makes it more stable. Most importantly, the invertible meaningful latent space with the transformation matrix facilitated the on-demand makeup transfer, that is, users can freely adjusting the makeup from light to heavy and BeautyGlow spontaneously synthesizes the results (less than 1 second).[3] The contributions of this paper are summarized as follows.

- Inspired by Glow, we propose BeautyGlow that can transfer the makeup from reference image to target image. The meaningful latent space facilitates on-demand makeup adjustment. To the best of our knowledge, this is the first Glow-based makeup transfer framework.
- New transformation matrix and loss function are formulated to guide the model training. It is worth noting that the proposed framework can be easily extended to other applications that require decomposing the latent image vector into two latent vectors, e.g., rain removal, fog removal.
- Experimental results on quantitative and qualitative comparison manifest that the proposed BeautyGlow is comparable to the state-of-the-art methods, while the manipulation on latent vectors can generate realistic images from light makeup to heavy makeup.

2. Related works

2.1. Makeup Studies

Traditionally, image processing techniques are applied for makeup transfer. For example, image analogy [11] introduces a framework that requires a pair of well-aligned before-makeup and after-makeup photos of the same person for makeup transfer. Guo *et al.* decompose facial details by decomposing reference images into three layers and transfer information from each layer to the corresponding layer of the other image [10]. Moreover, [28] proposes to detect the face landmarks and transfer the makeup by adjusting the landmark with skin color GMM-based segmentation. In addition to process the makeup, [17] manipulates the intrinsic image layers with a physically-based reflectance model to simulate the makeup with different lighting conditions in a photo. Compared with these work, the proposed BeautyGlow can perform makeup transfer and makeup removal simultaneously with on-demand makeup adjustment by manipulating the latent space without the need of post-processing.

[3] A demo is available at https://beautyglow.github.io/.

2.2. Style Transfer

To mix the content and style together, most works [7, 29] use different layers in CNN to represent styles and content and swap the style for style transfer. The style transfer can be categorized into three classes. 1) Per-Style-Per-Model (PSPM) [14, 25, 26, 8, 21, 20]. PSPM requires training different models for different styles. As the number of styles becomes large (e.g., makeup), it is difficult and inconvenient to use PSPM for style transfer. 2) Multiple-Style-Per-Model (MSPM) [5, 2, 6]. MSPM only requires a trained model to transfer multiple styles by slightly adjusting the instance normalization layers. 3) Arbitrary-Style-Per-Model (ASPM) [13, 9, 12]. ASPM focuses on the zero-shot style transfer, which can transfer the new styles (unseen in the training data) by finding the transformation function via a style prediction network. It is worth noting that the proposed BeautyGlow belongs to ASPM. However, compared with style transfer, the requirement of the details for makeup transfer is much higher than that for style transfer.

2.3. GAN for Style/Makeup Transfer

Due to the good performance of generating realistic images by adversarial network, GAN is widely-used for image-to-image translation. For example, Taigman *et al.* propose a method to transfer a realistic photo to comic characters by GAN and VAE. Moreover, [18, 24, 16, 22] aim to transfer style, attributes or specific details by exploiting GANs. CycleGAN further introduces two coupled generator and a cycle consistency loss to transfer images between two different domains [30], which avoids the requirement of input-output pairs. Although CycleGAN can be directly exploited to makeup transfer, i.e., one generator transfers a general makeup to non-makeup faces, and the other generator removes makeup from makeup faces, it cannot transfer a user-specified style. To fulfill the need to transfer a specific makeup to a non-makeup face, [1] alters the coupled generator in CycleGAN by using two images (source and target) for the makeup generator. The generator generates a makeup mask referenced from the target image and covers it on the source image to complete the makeup transfer. On the other hand, BeautyGAN adopts similar idea with dual input and output for makeup transfer and removal and enhance the correctness of instance-level makeup transfer by matching the color histogram in different segments of the face [19]. However, GAN-based methods contain no encoder to construct the latent space from the data and thus can not realize on-demand makeup transfer by simply interpolating the latent vectors.

2.4. Latent Space Adaption

The likelihood-based approaches, such as Variational AutoEncoders (VAE) and Glow, generate realistic images with a latent space, while manipulating the latent space model can facilitate developing new image applications, e.g., image editing [31], avatar synthesis [27]. Specifically, VAE optimizes the likelihood by maximizing the variational lower bound and learns a latent space that represents all the data points. On the other hand, flow-based generative models [3, 4] construct a bijective reversible transfer function F from the image space to the latent space so that modifying latent vectors can be mapped back to real images with F^{-1}. Glow further improves the model architecture by a new permutation layer which is learned after updating the model rather than the fixed permutation layer [15].

3. BeautyGlow

3.1. Glow

We first present the Glow framework [15] as the preliminary. Glow introduces an invertible function f comprising a sequence of transformation matrices for mapping images into a latent space, where f is required to fulfill the following properties. (1) The determinant of the Jacobian matrix of the transformation matrix should be calculated efficiently. (2) From the sampled datapoint z in latent space, mapping it back to the data x by using the inverse transformation $x = f^{-1}(z)$ should also be obtained easily. To accomplish these goals, Glow utilizes the additive layer and affine coupling layer introduced in [4]. A coupling layer represents a bijection function, which can update part of the input vector or latent vectors efficiently. Also, squeezing layer is adapted to implement the multi-scale architecture, which divides the image into sub-images of shape $2 \times 2 \times c$, then reshapes them into $1 \times 1 \times 4c$ for squeezing image information into channels. On top of that, before propagating the output of the current level to the next level, half of the dimensions of the output are factored out and dumped into latent space to reduce the computational cost and the number of parameters. Furthermore, Glow replaces the fixed permutation in [4] with a new layer, Invertible 1×1 conv, to avoid some components of the input vector is left unchanged in the coupling layer. Based on this model, we are able to transform input images into meaningful vectors in latent space and manipulate the vectors for on-demand makeup transfer.

3.2. Formulation

Based on the latent space derived from Glow, the goal is to extract the makeup features from the reference makeup image and apply it to the source non-makeup image. Specifically, considering two domains, the non-makeup images domain denoted as $X \subset \mathbb{R}^{\mathrm{h} \times \mathrm{w} \times \mathrm{c}}$ where **h**, **w**, and **c** represent the height of input images, the width of input images, and the number of RGB channels, respectively. The makeup images domain denoted as $Y \subset \mathbb{R}^{\mathrm{h} \times \mathrm{w} \times \mathrm{c}}$, which are encoded into the latent space denoted as$Z \subset \mathbb{R}^{\mathrm{c} \times \mathrm{h} \times \mathrm{w}}$

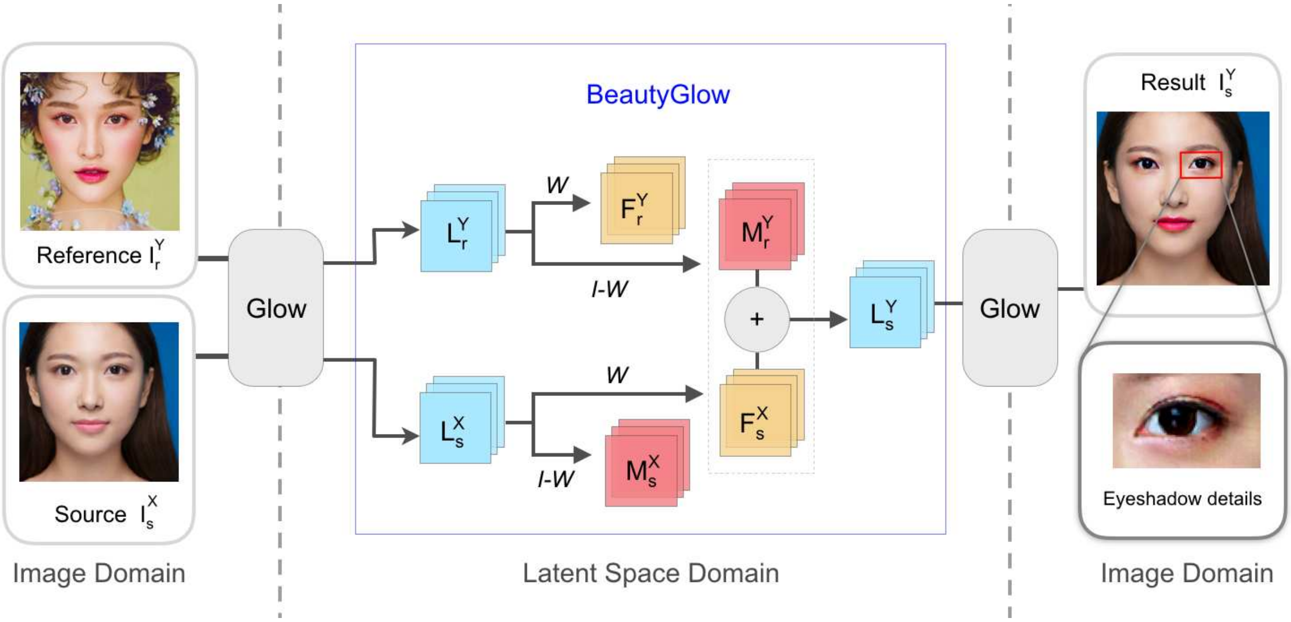


Figure 2: *The framework of the BeautyGlow. Glow transforms all images in makeup domain I^Y and all images in the non-makeup domainI^X as inputs: they are encoded into latent space Z, denoted as L^X and L^Y respectively. a vector obtained from L^X as source non-makeup, L_s^X, and a vector obtained from L^Y as reference makeup image, L_r^Y, are decomposed into two latent features (F_s^X, M_s^X), (F_r^Y, M_r^Y) respectively through a transformation matrix W. L_s^Y is generated by adding F_s^X and M_r^Y together and L_s^Y can also be decomposed into reconstruction latent features, (F_s^Y, M_s^Y).*

through Glow[15].

Given two images as inputs: a source image $I_s^X \in X$ and a reference image $I_r^Y \in Y$ are encoded into two latent vectors, $L_s^X \in Z$ and $L_r^Y \in Z$ respectively with Glow. After that, a transformation matrix W is expected to extract facial features denoted as $F_s^X = L_s^X \cdot W$ and extract makeup features denoted as $M_r^Y = L_r^Y \cdot (I - W)$, where I is identity matrix. Here, we adopt the additive formulation, i.e., adding F_s^X and M_r^Y to form after-makeup latent vector $L_s^Y \in Z$, since 1) additive formulation can be computed on-the-fly, and 2) the interpolation in additive formulation is intuitive. The after-makeup vector L_s^Y is then decoded back to image domain as the after-makeup image $I_s^Y \in Y$. The model architecture of BeautyGlow is shown in Figure 2.

However, how to learn the transformation matrix W that perfectly decomposes the latent vectors into latent facial features and latent makeup features remains challenging.

3.3. Objective

To tackle this challenge, we introduce several losses to guide the decomposition. 1) The Perceptual loss is proposed to extract the facial features. 2) Intra-domain loss and inter-domain loss are designed for ensuring the after-makeup image and de-makeup image being at the makeup and non-makeup domain respectively. 3) Makeup loss is to extract the makeup features. 4) Cycle consistency loss maintains the face and makeup information as input features. The details of loss functions and model architecture are presented as follows, while the effects of different losses are evaluated in Section 4.

Perceptual Loss. We first introduce the perceptual loss $\mathcal{L}_p$ to teach W how to extract facial features. Since the original source image is assumed to be non-makeup, we define the perceptual loss as follows.

$$\mathcal{L}_p = \|F_s^X - L_s^X\|_2, \tag{1}$$

which constrains the distance between the facial latent feature F_s^X and the latent features of original source image L_s^X.

Makeup Loss. Since the latent vector of the reference image includes makeup features, which means the W should be able to discriminate face features and makeup features. However, there is no image representing makeup styles, and thus the makeup features cannot be derived by transferring the image through Glow. Therefore, it is challenging to train how to describe a makeup style with several latent features. Here, we first tackle this problem by assuming that the latent features of a human face image are composed of facial features and makeup features. When the facial features are removed, the rest is makeup features. Meanwhile, as shown in Glow[15], attributes could be extracted through manipulating the latent features for images. Therefore, we first calculate the average latent vector of all images with makeup, denoted as $\bar{L}^Y$, and the average latent vector of all images without makeup, denoted as $\bar{L}^X$. Since the difference $(\bar{L}^Y - \bar{L}^X)$ represents the direction from non-makeup latent vector to makeup latent vector, which should be similar to M_r^Y, we formulate the makeup loss $\mathcal{L}_m$ as follows.

$$\mathcal{L}_m = \|M_r^Y - (\bar{L}^Y - \bar{L}^X)\|_2. \tag{2}$$

Intra-Domain Loss. The makeup loss only forces M_r^Y to be similar with the direction from non-makeup latent vector to makeup latent vector, while the centroids of non-makeup latent vectors and makeup latent vectors are not fully exploited. The facial latent vectors of reference images, F_r^Y, are supposed to be close to non-makeup domain rather than makeup domain. Moreover, the after-makeup latent vectors (L_s^Y) are supposed to be close to the makeup domain instead of the non-makeup domain. Therefore, we use the average latent vector of all non-makeup images, $\bar{L}^X$, to represent the centroid of non-makeup domain, and the average latent vector of all makeup images, $\bar{L}^Y$ to represent the centroid of makeup domain. The intra-domain loss is then defined as

$$\mathcal{L}_{intra} = \|F_r^Y - \bar{L}^X\|_2 + \|L_s^Y - \bar{L}^Y\|_2. \tag{3}$$

Inter-Domain Loss. In addition to the intra-domain loss, we further introduce inter-domain loss to ensure that F_r^Y is away from the centroid of makeup domain to clearly decompose the facial latent vectors and makeup latent features effectively. Meanwhile, L_s^Y is also supposed to be away from the centroid of non-makeup domain. As such, we formulate the inter-domain loss as follows. Instead of the L2-norm, we calculate the similarity between two latent vectors A and B as

$$Sim(A, B) = \frac{sum(A \bigotimes B)}{|A||B|}, \tag{4}$$

where $\bigotimes$ denotes element-wise multiplication and $|\cdot|$ denote the L2-norm of the matrix. The loss function is expressed as

$$\mathcal{L}_{inter} = (1 + Sim(F_r^Y, \bar{L}^Y)) + (1 + Sim(L_s^Y, \bar{L}^X)) \tag{5}$$

Cycle Consistency Loss. In order to maintain the facial and makeup information, two cycle consistency losses are also designed in the latent space. Specifically, if we multiply the after-makeup latent vector, i.e., L_s^Y, with transformation matrix W, it supposed to be close to the facial latent vectors of the source image, i.e., F_s^Y. On the other hand, if we multiply L_s^Y with $(I - W)$, it is supposed to be close as makeup latent features of reference latent features M_r^Y. Therefore, the loss function is formulated as

$$\mathcal{L}_{cyc} = \|L_s^Y W - F_s^X\|_2 + \|L_s^Y (I - W) - M_r^Y\|_2 \tag{6}$$

Total loss. In sum, the overall loss function $\mathcal{L}$ is defined as follows.

$$\mathcal{L} = \mathcal{L}_p + \lambda_{cyc}\mathcal{L}_{cyc} + \lambda_m\mathcal{L}_m + \lambda_{ia}\mathcal{L}_{intra} + \lambda_{ie}\mathcal{L}_{inter} \tag{7}$$

where λ_{cyc}, λ_m, λ_{ia}, and λ_{ie} are the weights to control the relative importance of each term. The transformation matrix W can be trained via the objective function in Equation 7 with gradient descent based method, which will be detailed later.

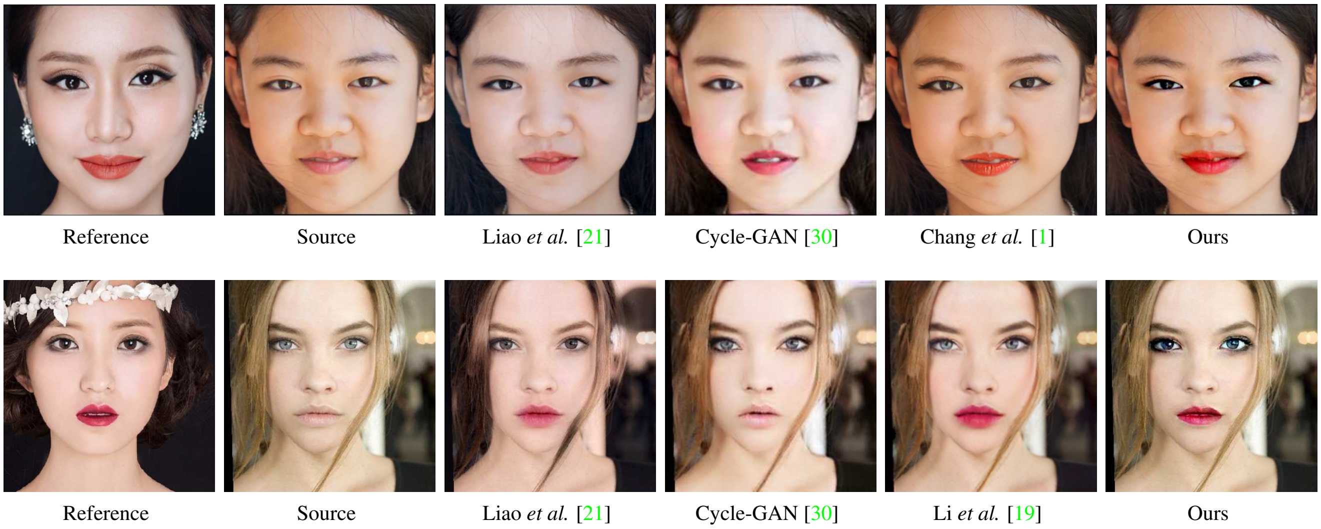
4. Experimental Results

In this section, we first describe the implementation details and training parameter setting. Afterward, we briefly introduce the baselines, which are the state-of-the-art methods in makeup transfer. We conduct both qualitative and quantitative experiments to compare with baselines. A qualitative analysis on different losses of BeautyGlow is also presented. Finally, we show the advantages of flow-based method by presenting the makeup results from light to heavy, which is simply derived by interpolating the latent vectors and inverting back to image domain.

4.1. Implementation Details

Training Pipeline. We evaluate our method on MT dataset [19] consisting of about 4000 female images (1000 non-makeup images and 3000 makeup images). MT dataset includes different races, poses, expression and makeup styles varying from subtle to heavy. Therefore, we first follow the architecture of Glow [15] with pre-trained weights. Glow was originally trained on 5 machines with each 8 GPUs. Due to the limitation of the number of GPUs and GPU memories, we resize the training image size to 128x128. It takes about 3 days to train a Glow model with MT dataset.

Moreover, due to the resolution constraint, we first apply face parsing, i.e., separate different face parts and keep makeup details. Images are separated (with/without makeup) into left eye, right eye, lips, and the rest since eyes and lips are significantly changed by makeup. Please note that we only train single W which is shared by different parts of the faces. In post-processing, we use Poisson blending to integrate the generated makeup facial parts and the source image.



Reference | Source | Liao *et al.* [21] | Cycle-GAN [30] | Chang *et al.* [1] | Ours

Reference | Source | Liao *et al.* [21] | Cycle-GAN [30] | Li *et al.* [19] | Ours

Figure 3: Makeup Transfer Results

Training Details. Given the dataset of unpaired makeup and non-makeup images, we first use Glow to transform the images into the latent space and calculate the centroids of makeup and non-makeup latent domains. Afterward, we train the transformation matrix W via the objective function in Equation 7 with Adam optimizer, which is a classic extension of stochastic gradient descent procedure to update model weights iteratively based on the training data. The learning rate is set as 0.001 with the batch size as 100. The size of the transformation matrix W is 128 × 128, while the control parameters of perceptual loss, makeup loss, intra-domain loss, inter-domain loss, and cycle consistency loss are set as $(\lambda_p = 0.01, \lambda_{cyc} = 0.001, \lambda_m = 0.1, \lambda_{intra} = 0.1, \lambda inter = 1000)$, respectively.

4.2. Baselines

BeautyGlow is compared with several methods including traditional method, style transfer and image domain adoption via qualitative and quantitative experiments as listed below.

Deep Image Analogy [21] extracts features by CNN and adapts the notion of image analogy to the deep feature space for obtaining semantically-meaningful dense correspondences. Here, we add the WLS filter to keep details for photo-to-photo transfer.

CycleGAN [30] is an unsupervised image-to-image translation work, which does not require paired images. Therefore, makeup images and non-makeup images are regarded as two different domains for training.

PairedCycleGAN [1] is a paired image-to-image makeup transfer method which aims to transfer a specific reference makeup to a source image.

BeautyGAN [19] is an instance-level makeup transfer method for makeup transfer and removal via GAN. The results are enhanced by matching the color histogram in different segments of the face.

4.3. Qualitative Comparison

In Figure 3, we demonstrate a qualitative comparison of the baselines and our results. The results produced by Image Analogy show that eyeshadows and eyeliners disappear and only subtle color changes stay around the eyes. This is because it applies whole image transformation, parts aside from makeup including skin tone and hair might also be changed to an unnatural color. Compared with Image Analogy, image domain adoption methods can create a more realistic look. CycleGAN [30] only synthesizes a general makeup on the non-makeup face. The makeup style or amount cannot be adjusted since the images are transferred in only two general domain, makeup and non-makeup. The general makeup, though realistic, cannot be used for specific makeup trials. Moreover, PairedCycleGAN [1] can transfer makeup realistically and correctly in pairs. However, the disadvantages of PairedCycleGAN is that makeup are generated by a fixed adversarial network so that users cannot adjust the makeup to be lighter or heavier. Therefore, when transferring some heavy makeup, the results may look unpleasant and sometimes unrealistic, but users can only find new reference images with lighter makeup, which is time-consuming and even unfeasible.

Comparing to the baselines, our method successfully decomposes the makeup latent vectors and non-makeup latent vectors, so it does not change the look of the image severely like style transfer. Meanwhile, we can generate a realistic

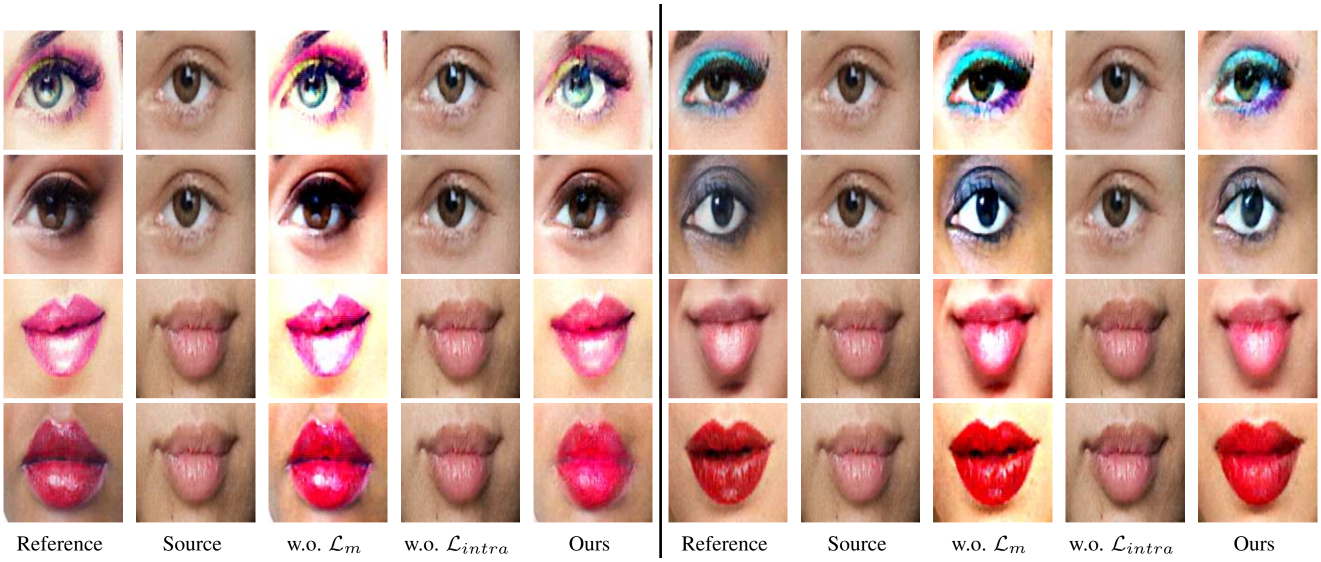


Figure 4: Analysis of Different Loss Terms of BeautyGlow.

Table 1: User Study Results

preference comparison	our result	baseline
Ours / Liao *et al.* [21]	60%	40%
Ours / Li *et al.* [19]	55%	45%
Ours / Chang *et al.* [1]	45%	55%

and accurate makeup for a non-makeup face comparable to PairedCycleGAN. In Section 4.6, we show the power of the ability to fine-tune the amount of makeup and customize it for each face to get the perfect result.

4.4. Quantitative Comparison

For quantitative evaluation of BeautyGlow, a user study is conducted with 50 volunteers (34 males and 16 females) aged from 18 years old to 35 years old. To simplify the process of comparison, a standard A/B test is used. Therefore, each user is asked to compare the results of BeautyGlow with Image Analogy [21], BeautyGlow with PairedCycleGAN [1], and BeautyGlow with BeautyGAN [19]. For each user, we randomly choose fifteen pairs of source and reference images and generate the results by different methods. The positions of different methods for the A/B test is also randomly selected to avoid the bias. Table 1 shows the results of the A/B test. For the ease of comparison, we normalized the votes and get the preference percentage. The results show that BeautyGlow significantly outperforms Image Analogy while is comparable to BeautyGAN and PairedCycleGAN. Note that we observe some unusual situations that user preference decreases while transferring heavy makeup. Some exaggerated makeup is not suitable for the source image, forcing users to choose a rather subtle makeup face. As a result, with the latent space manipulation in our model, we can fine-tune the makeup amount for a more preferred look.

4.5. Loss Analysis

We introduce five losses for training the transformation function. In Figure 4, we show the results of two important loss terms by training the transformation function W without the loss term. When removing $\mathcal{L}_m$, the transformation matrix learns makeup features without average makeup features. Hence, the makeup style is over-enhanced, leading to unnatural after-makeup images, especially in the regions of eyes. On the other hand, without $\mathcal{L}_{intra}$, the after-makeup image contains no makeup components because there is no loss term constraining the resulting image to be close to the centroid of the makeup images. In addition, facial features are supposed to be at the group of non-makeup images, so the facial features extracted from makeup images would include makeup features.

We show the results of face components like eyes and mouth for different pairs of a reference image and a source image. In the first two rows, without using makeup loss, the makeup style is over-emphasized and the color of other regions is changed dramatically. When the intra-loss term is removed, the result looks like the same as the source image. Moreover, in the second row, the color of the results without makeup loss is obviously brighter than the source and the reference image.

4.6. Latent Space Manipulation

The highlighted property of BeautyGlow is the ability to manipulate the latent vectors for generating the sane makeup style with different lightness. To show the after-

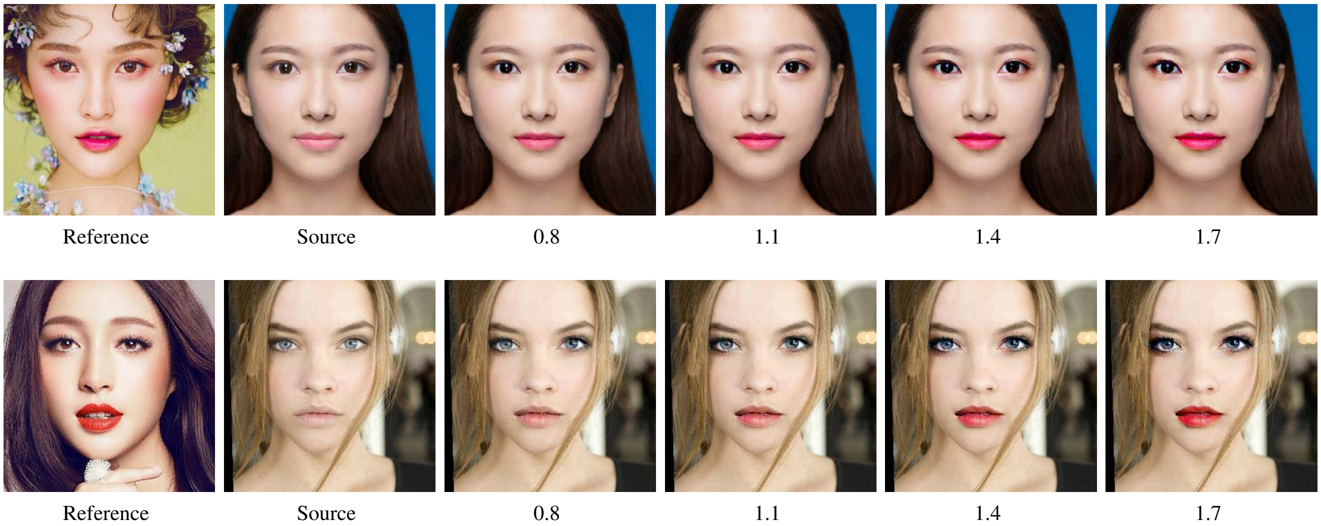


Figure 5: The obviousness of the makeup

makeup face with different makeup lightness, we manipulate the weight of the makeup features in the latent space. In Figure 5, we synthesize 3 different lightness of two makeup styles on different faces with the magnification of 0.7, 1.0, 1.5. In the first row, as the magnification increases, the eyeshadows and lip color become redder and redder. In the second row, the eyeshadows also become darker and the lip color becomes redder. It is worth noting that even though the makeup is magnified, the face is still not distorted, which means the transformation matrix completely decomposes the latent vector into the facial features and makeup features for the manipulations.

5. Conclusions and Future Work

As quoted from Estée Lauder, “Glow is the essence of beauty,” which is also true for makeup transfer. In this paper, we propose BeautyGlow to decompose the latent vectors derived from the Glow model into makeup and non-makeup latent vectors. Since there is no paired dataset, we formulate a new loss function containing perceptual loss, makeup loss, inter-domain loss, intra-domain loss, and cycle consistency loss, to guide the optimization of the transformation matrix. Experimental results show that the makeup transfer quality of BeautyGlow is comparable to the state-of-the-art methods, while the unique ability to manipulate latent vectors allows BeautyGlow to realize on-demand makeup transfer. In the future, we plan to apply the proposed general framework to other image synthesis applications.

6. Acknowledgements

This work was supported in part by the Ministry of Science and Technology of Taiwan under grants MOST-107-2218-E-009-062, MOST-107-2218-E-002-054, MOST-107-2221-E-182-025-MY2, MOST-105-2628-E-009-008-MY3, MOST-106-2221-E-009-154-MY2, and the Grant Number MOST 108-2634-F-009-006- through Pervasive Artificial Intelligence Research (PAIR) Labs.

References

[1] Huiwen Chang, Jingwan Lu, Fisher Yu, and Adam Finkelstein. Pairedcyclegan: Asymmetric style transfer for applying and removing makeup. In *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2, 3, 6, 7

[2] Dongdong Chen, Lu Yuan, Jing Liao, Nenghai Yu, and Gang Hua. Stylebank: An explicit representation for neural image style transfer. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[3] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. 2014. 3

[4] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. 2017. 2, 3

[5] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. *2017 International Conference for Learning Representations (ICLR)*, 2017. 3

[6] Michael Elad and Peyman Milanfar. Style transfer via texture synthesis. *IEEE Transactions on Image Processing*, 2017. 3

[7] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 3

[8] Leon A Gatys, Alexander S Ecker, Matthias Bethge, Aaron Hertzmann, and Eli Shechtman. Controlling perceptual fac-

tors in neural style transfer. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[9] Golnaz Ghiasi, Honglak Lee, Manjunath Kudlur, Vincent Dumoulin, and Jonathon Shlens. Exploring the structure of a real-time, arbitrary neural artistic stylization network. *2017 British Machine Vision Conference (BMVC)*, 2017. 3

[10] Dong Guo and Terence Sim. Digital face makeup by example. In *2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 1, 2

[11] Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless, and David H. Salesin. Image analogies. In *SIGGRAPH '01*, 2001. 2

[12] Wei-Lin Hsiao and Kristen Grauman. Learning the latent look: Unsupervised discovery of a style-coherent embedding from fashion images. In *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017. 3

[13] Xun Huang and Serge J Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017. 3

[14] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *2016 European Conference on Computer Vision (ECCV)*, 2016. 3

[15] Diederik P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *arXiv preprint arXiv:1807.03039*, 2018. 2, 3, 4, 5

[16] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. *2018 The European Conference on Computer Vision (ECCV)*, 2018. 3

[17] Chen Li, Kun Zhou, and Stephen Lin. Simulating makeup through physics-based manipulation of intrinsic image layers. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[18] Minjun Li, Haozhi Huang, Lin Ma, Wei Liu, Tong Zhang, and Yugang Jiang. Unsupervised image-to-image translation with stacked cycle-consistent adversarial networks. *2018 European Conference on Computer Vision (ECCV)*, 2018. 3

[19] Tingting Li, Ruihe Qian, Chao Dong, Si Liu, Qiong Yan, Wenwu Zhu, and Liang Lin. Beautygan: Instance-level facial makeup transfer with deep generative adversarial network. In *2018 ACM Multimedia Conference on Multimedia Conference (ACMMM)*, 2018. 2, 3, 5, 6, 7

[20] Yijun Li, Ming-Yu Liu, Xueting Li, Ming-Hsuan Yang, and Jan Kautz. A closed-form solution to photorealistic image stylization. *2018 European Conference on Computer Vision (ECCV)*, 2018. 3

[21] Jing Liao, Yuan Yao, Lu Yuan, Gang Hua, and Sing Bing Kang. Visual attribute transfer through deep image analogy. *2017 ACM Transactions on Graphics (TOG)*, 2017. 3, 6, 7

[22] Jianxin Lin, Yingce Xia, Tao Qin, Zhibo Chen, and Tie-Yan Liu. Conditional image-to-image translation. In *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[23] Si Liu, Xinyu Ou, Ruihe Qian, Wei Wang, and Xiaochun Cao. Makeup like a superstar: Deep localized makeup transfer network. 2016 International Joint Conference on Artificial Intelligence (IJCAI), 2016. 1

[24] Amélie Royer, Konstantinos Bousmalis, Stephan Gouws, Fred Bertsch, Inbar Moressi, Forrester Cole, and Kevin Murphy. Xgan: Unsupervised image-to-image translation for many-to-many mappings. *2018 International Conference for Learning Representations (ICLR)*, 2017. 3

[25] Dmitry Ulyanov, Vadim Lebedev, Andrea Vedaldi, and Victor S Lempitsky. Texture networks: Feed-forward synthesis of textures and stylized images. In *2016 International Conference on Machine Learning (ICML)*, 2016. 3

[26] Dmitry Ulyanov, Andrea Vedaldi, and Victor S Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[27] Lior Wolf, Yaniv Taigman, and Adam Polyak. Unsupervised creation of parameterized avatars. In *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017. 3

[28] Lin Xu, Yangzhou Du, and Yimin Zhang. An automatic framework for example-based virtual makeup. In *2013 IEEE International Conference on Image Processing (ICIP)*, 2013. 2

[29] Yexun Zhang, Ya Zhang, and Wenbin Cai. Separating style and content for generalized style transfer. In *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[30] J Zhu, T Park, P Isola, and AA Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks, 2017. 2, 3, 6

[31] Jun-Yan Zhu, Eli Shechtman Philipp Krähenbühl, and Alexei A. Efros. Generative visual manipulation on the natural image manifold. In *2016 European Conference on Computer Vision (ECCV)*, 2016. 3