Lipstick ain't enough: Beyond Color Matching for In-the-Wild Makeup Transfer

Thao Nguyen[1]   Anh Tuan Tran[1,2]   Minh Hoai[1,3]

[1]VinAI Research, Hanoi, Vietnam, [2]VinUniversity, Hanoi, Vietnam,
[3]Stony Brook University, Stony Brook, NY 11790, USA

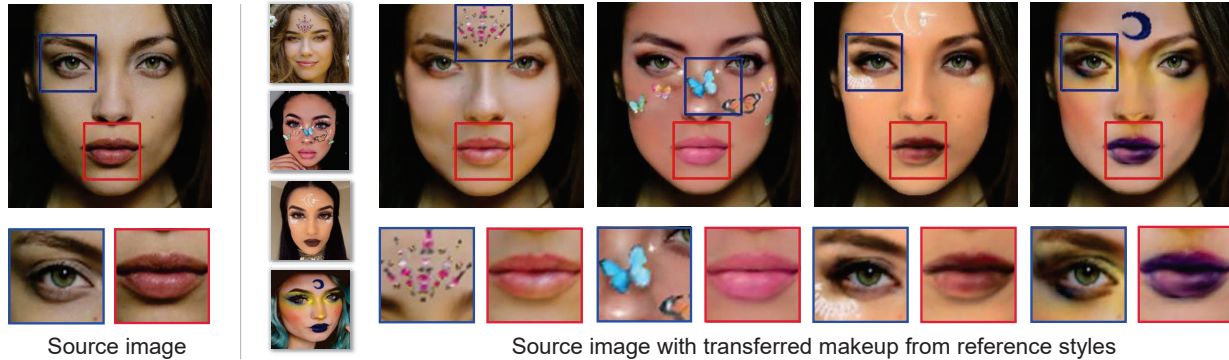{v.thaontp79,v.anhtt152,v.hoainm}@vinai.io

Figure 1: In-the-wild facial makeup consists of both color transfer and pattern addition. We propose a holistic method that can transfer the color and pattern from a reference makeup style to another image.

Abstract

Makeup transfer is the task of applying on a source face the makeup style from a reference image. Real-life makeups are diverse and wild, which cover not only color-changing but also patterns, such as stickers, blushes, and jewelries. However, existing works overlooked the latter components and confined makeup transfer to color manipulation, focusing only on light makeup styles. In this work, we propose a holistic makeup transfer framework that can handle all the mentioned makeup components. It consists of an improved color transfer branch and a novel pattern transfer branch to learn all makeup properties, including color, shape, texture, and location. To train and evaluate such a system, we also introduce new makeup datasets for real and synthetic extreme makeup. Experimental results show that our framework achieves the state of the art performance on both light and extreme makeup styles. Code is available at https://github.com/VinAIResearch/CPM.

1. Introduction

Across thousands of years of history, humankind has been fascinated with facial beauty. Humans, particularly females, want to be attractive, and facial appearance is a crucial part of this. The cosmetic industry, as reported in 2007, generates a turnover of about $170 billion each year [27].

Among face beautification techniques, makeup is the most popular method, accompanied by a wide range of commercial products including foundation, eye shadow, lipsticks, blushes, stickers, facial drawings, and facial accessories.

Due to the popularity of makeup, makeup try-on is an vital application in both retail and entertainment. Among makeup try-on techniques, makeup transfer is the most convenient and effective way. Makeup transfer is the task of transferring the makeup style from one reference face to another face. This task is not trivial; it needs to extract makeup components from the composited reference image. It also needs to analyze the face structure to transfer makeup components between unaligned faces correctly, and there are many factors to account for, including head pose, illumination, facial expressions, and occlusions.

Deep-learning-based generative models are leading methods in tackling this problem. BeautyGAN [16] and BeautyGlow [6] can provide realistic after-makeup images for simple styles on frontal faces. PSGAN [12] manages to handle faces at various head poses and expressions, while CA-GAN [14] focuses on fine-grained makeup-color matching. However, these methods can only work with simple makeup styles based on color distributed in cosmetic regions such as skin foundations, lipsticks, and eye-shadows. They fail miserably on the complex makeups that rely on shape, texture, and location, such as blushes, face paintings, and makeup jewelries. Only LADN [10] considers these ex-

treme makeups, but its results are far from satisfactory.

In this work, we consider makeup as a combination of color transformation and pattern addition. We aim to transform the color distribution like previous methods while also preserving the shape and appearance of the makeup pattern. To achieve this objective, we introduce a framework with two branches: Color Transfer Branch and Pattern Transfer Branch, which could be run independently in parallel. In the Color Transfer Branch, we employ a CycleGAN-like network structure driven by Histogram Matching as suggested by BeautyGAN [16]. In the Pattern Transfer Branch, we learn to extract the makeup pattern mask in a supervised manner. Noticeably, unlike previous methods, both our branches work on warped faces in UV space, thus discarding the discrepancy between these faces in terms of shape, head pose, and expression. The results of the two branches are fused to generate the desired output.

We also introduce new makeup-transfer datasets, consisting of both synthetic and real images, and covering a wide range of makeup styles. They include extreme makeup styles, which do not exist in previous makeup datasets.

Using the novel network architecture and the newly collected datasets for training, we obtain an all-inclusive makeup transfer method that outperforms all previous methods in terms of coverage, as shown in Fig. 2b. We also run comprehensive experiments, both qualitative and quantitative, and proposed makeup-transfer benchmarks. Our method outperforms other methods on both light and extreme makeup transfer by a wide margin.

In short, our contributions are: (1) We pose makeup as a combination of color transformation and pattern addition, and develop a comprehensive makeup transfer method that works for both light and extreme styles. (2) We design a novel architecture with two branches for color and pattern transfer, and we propose to use warped faces in the UV space when training two network branches to discard the discrepancy between input faces in terms of shape, head pose, and expression. (3) We introduce new makeup-transfer datasets containing extreme styles that have not been considered in the previous datasets. (4) We obtain state-of-the-art quantitative and qualitative performance.

2. Related Work

Facial Makeup Transfer. Facial makeup has been studied [18] in computer vision. Given an arbitrary facial image with the desired makeup style, makeup transfer aims to analyze and replicate that makeup to a source image.

Traditional methods [11, 22] focused on image prepossessing techniques, such as landmark extraction and adjustment [28] or reflectance manipulation [15]. Recently, due to high-performance hardware and the ability to generate aesthetic images, GANs are widely-used for image-to-image translation tasks, including facial makeup synthesis.

CycleGAN-based models [2, 32] were introduced to transfer face-to-face makeup styles in an unsupervised manner. For more realistic outcomes, BeautyGAN [16] used Histogram Matching at each facial region to guide the instance-level makeup synthesis. BeautyGLOW [6] proposed to decompose makeup and non-makeup components in the latent space, using the GLOW framework [13]. LADN [10] incorporated multiple and overlapping local discriminators for extreme makeup transfers. PSGAN [12] employed an Attentive Makeup Morphing module to handle transfer across different head poses and facial expressions. Lately, CA-GAN [14] proposed color discriminators to improve fine-grain makeup color transfer at the lips and eye regions.

Most aforementioned methods only consider light makeup based on color transformation in cosmetic regions such as lips and eye-shadows. In-the-wild makeup styles, however, can also cover pattern-based components such as stickers, face drawings, and decoration. To the best of our knowledge, only LADN [10] focused on those extreme makeup styles, however, it has several limitations. First, due to the unsupervised setup, it cannot handle complicated makeup patterns with fine details. Second LADN suffers when the head pose of the source and the reference faces are different, producing noticeable artifacts. Finally, it generates low-quality outputs with evident image degradation traits such as JPEG compression noise and blurry edges.

In this paper, we propose a holistic makeup framework that handles both makeup color and pattern transfer. Our method overcomes the limitations of LADN; it can deal with complicated makeup patterns, be robust to head pose, and produce high-quality outputs.

Although several datasets of makeup faces have been assembled [4, 5, 7, 10, 16], they mainly cover either light or color-focused makeup styles. Since adding patterns is an important part of makeup, pattern-included makeup transfer datasets should be built. We, therefore, introduce such novel datasets for both real and synthetic makeups.

3D Face Modelling from a single image. To transfer makeup components between faces, we need to understand their facial structures. The human face is a 3D object, and there are many 3D features affecting its appearance in images such as shape, pose, and expression. Thus, reconstructing a 3D face for each input image is crucial to our task.

3D face modeling from single images has been studied for more than two decades [26]. Among various classical approaches, 3D-morphable models (3DMMs) [1, 17, 20] were the most successful. A 3DMM is a parameterized statistical representation of the 3D faces' manifold learned from 3D face scans. Basel Face Model (BFM) [17] is the most popular 3DMM, which approximates any 3D face as a weighted sum of a mean face and principal shape components. The 3D modeling task is then converted to weight optimization so that the composited 3D face is similar to the
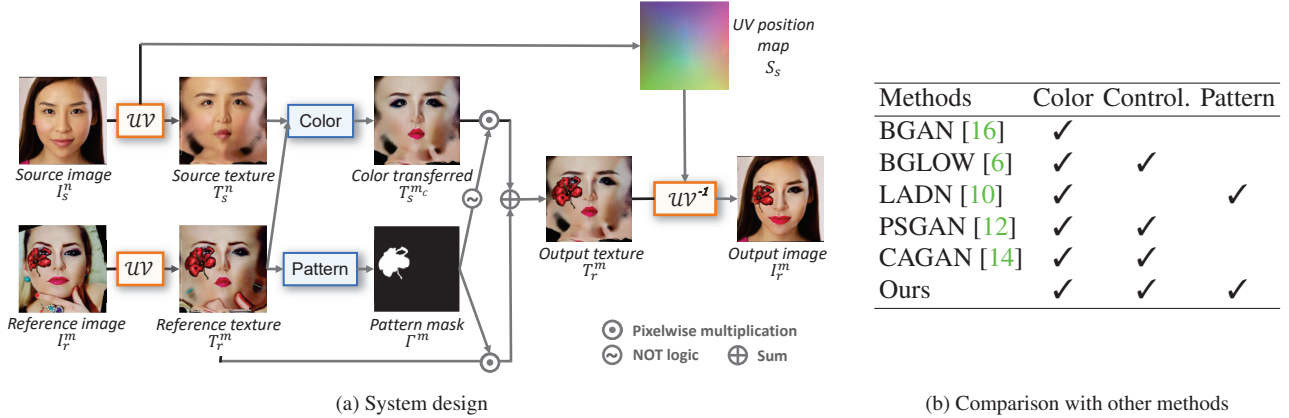
(a) System design   (b) Comparison with other methods

Methods	Color	Control.	Pattern
BGAN [16]	✓		
BGLOW [6]	✓	✓	
LADN [10]	✓		✓
PSGAN [12]	✓	✓	
CAGAN [14]	✓	✓	
Ours	✓	✓	✓

Figure 2: Overview of the proposed framework. 'Control.' indicates controllable for partial makeup transfer.

one in the input image.

Like most other computer vision tasks, 3D face modeling is experiencing fast growth in recent years thanks to deep learning. The first attempts to apply deep learning on this task relied on 3DMM parameter regression via supervised training [19, 23, 33]. MoFA [21] proposed to use an auto-encoder for unsupervised training. Tran and Liu [25] exploited the auto-encoder to learn a nonlinear 3DMM model for better 3D fitting. PRNet [9] designed a 2D representation, called UV position map, to encode the aligned 3D face shape, thus converting 3D face modeling to an image-to-image translation problem. The UV representation is easy to use and manipulate, thanks to its 2D form while removing the effect of head poses and expressions. The later 3D face modeling works mainly focused on reconstructing 3D details for realistic modeling [3, 24, 29].

In this paper, we employ PRNet [9] in our system since its accuracy is sufficiently good for our task. Furthermore, we take advantage of its UV representation for effective makeup swapping across faces.

3. Color-&-Pattern Makeup Transfer Method

Let I_s^n denote the source image (s) with non-makeup (n) and I_r^m the reference image (r) with the desired makeup (m). Our goal is to obtain I_s^m, an image of the source face with transferred makeup from the reference image. It requires learning a function $\mathcal{F}$ such that:

$$\mathcal{F}(I_s^n, I_r^m) = I_s^m \qquad (1)$$

In this section, we will describe our method to learn this function. Our method is designed based on the following two insights. First, the source and target images are not aligned due to different 3D head poses, face shapes, and facial expressions. To remove the misalignment, we should register these images to a uniform template before transferring makeup, and we specifically propose to use the UV map representation. Second, makeup transfer should be viewed as a combination of color transformation and pattern

addition. Pattern addition is categorically different from color transformation, and it should be explicitly handled by a specific module of the proposed solution.

In overview, our method consists of the following three steps. First, the input images I_s^n, I_r^m are converted to UV texture maps T_s^n, T_r^m. Second, the texture maps are passed to two parallel branches for color-based and pattern-based makeup transferred. Third, the makeup-transferred texture T_s^m is formed by combining the outputs of those branches, and this UV texture map is converted to the image space to obtain the final output I_s^m. The pipeline of our method is depicted in Fig. 2a. In the rest of this section, we will describe the details of the main components.

3.1. UV map conversion

UV map representation is a common technique for 3D object texture mapping in computer graphics. The object's texture is flattened into a 2D image, and each 3D vertex of the object is associated with a 2D location on the image, called UV coordinates, for color sampling. PRNet [9] extended this idea and introduced a UV position map representation to encode any 3D face shape. It is a 2D image with three channels encoding the XYZ coordinates of the 3D face with respect to the camera coordinates. This UV map is well registered; each pixel in the map corresponds to a fixed semantic point on the face regardless of the input head pose. Alongside the UV position map, we do texture mapping to get the paired texture map. The UV position map packs all information about face shape, head pose, and facial expression, while the mapped texture is invariant to those aspects.

Given an input facial image I, we can use the pre-trained model of PRNet, denoted as $\mathcal{UV}$, to extract the corresponding UV position map S and the UV texture T. The input image can be recovered from these UV representations via a rendering function $\mathcal{UV}^{-1}$.

$$S, T := \mathcal{UV}(I) \quad \text{and} \quad I := \mathcal{UV}^{-1}(S, T). \qquad (2)$$

To transfer makeup between source and reference images

with different head poses, we use these UV map presenta-
tions. First, we apply the conversion function $\mathcal{UV}$ on each
input image I_s^n and I_r^m to get the corresponding UV maps
(S_s, T_s^n) and (S_r, T_r^m). Note that the UV position maps S_s
and S_r depend only on 3D face shapes, thus being indepen-
dent of the makeup styles. Then, we pass the texture maps
T_s^n and T_r^m to the color and pattern transfer branches to get
makeup swapped in UV space. The outputs of two branches
are blended into final texture images T_s^m. Finally, we apply
the rendering function to convert it back to standard image
representation: $I_s^m = \mathcal{UV}^{-1}(S_s, T_s^m)$.

3.2. Color transfer branch

This branch adopts the architecture and training losses
proposed in BeautyGAN [16]. The main component is
a color-based makeup swapping network $\mathcal{C}$ that swaps
makeup color on cosmetic regions between the source and
the reference image: $T_s^{mC}, T_r^{nC} := \mathcal{C}(T_s^n, T_r^m)$. To train $\mathcal{C}$,
it uses a loss function as a weighted sum of the following:

- **Adversarial Loss** $\mathcal{L}_{adv}$ enforces the output maps T_s^{mC}
  and T_r^{nC} to be in makeup and non-makeup domain,
  respectively, using two discriminators,
- **Cycle Consistency Loss** $\mathcal{L}_{cyc}$ enforces the cycle con-
  sistency constraints proposed by CycleGAN [32],
- **Perceptual Loss** $\mathcal{L}_{per}$ aims to preserve the identity be-
  tween the before and after makeup transfer images by
  using the VGG-16 model pre-trained on ImageNet,
- **Histogram Matching Loss** $\mathcal{L}_{hist}$ aims to match the
  color distributions of the reference image and the
  source image after makeup transfer.

The first three loss functions are common, so we omit
the detailed discussion here. The final loss, i.e., $\mathcal{L}_{hist}$, is the
key loss function proposed by BeautyGAN for transferring
makeup color in cosmetic regions. It employs a Histogram
Matching (HM) function that alters the histogram of the
source image to match the reference one in each of several
predefined regions: eye shadows, lips, and facial skin. The
total loss is a weighted sum of the regional losses:

$$\mathcal{L}_{hist} = \lambda^{eyes}\mathcal{L}_{hist}^{eyes} + \lambda^{lips}\mathcal{L}_{hist}^{lips} + \lambda^{skin}\mathcal{L}_{hist}^{skin}, \quad (3)$$

where $\lambda^{eyes}, \lambda^{lips}, \lambda^{skin}$ are tunable hyper-parameters.

Each loss term $\mathcal{L}_{hist}^i$ (i can be eyes, lips, or skin)
is the distance between the after-makeup image and the
histogram-matched version:

$$\mathcal{L}_{hist}^i = \left|\left| T_s^{mC} \odot \Gamma_s^i - HM(T_s^n \odot \Gamma_s^i, T_r^m \odot \Gamma_r^i) \right|\right|. \quad (4)$$

where $\odot$ is pixel-wise multiplication, Γ_s^i and Γ_r^i are the seg-
mentation masks for region i in the source and the reference
image, respectively.

In BeautyGAN, the cosmetic regions are not aligned;
they highly differ in size, location, and perspective warping.

It severely impacts the histogram match results, reducing
the effectiveness of this histogram loss. While being simi-
lar to BeautyGAN, our Color Transfer Branch uses the UV
texture maps for makeup swapping instead of the original
images. This seemingly small innovation actually leads to
much improvement. The texture maps are registered pixel-
to-pixel, enabling the histogram matching function to work
accurately. The region mask is image-invariant and equals
to a universal mask: $\Gamma_s^i = \Gamma_r^i = \Gamma^i$.

We observe that our Color Transfer Branch produces bet-
ter results compared to BeautyGAN. It captures not only
color but also structure and location of the cosmetic make-
ups, which is crucial to some makeup components such as
blushes. We will discuss more this result in Sec. 5.5.

3.3. Pattern transfer branch

Besides the Color Transfer Branch, we propose a novel
Pattern Transfer Branch aiming to detect and transfer the
pattern-based makeup components such as stickers, facial
drawings, and decorative accessories. When transferring
these patterns, we need to keep them unchanged in terms
of shape, texture, and location but warped to the target 3D
surface. In the natural image form, this process is compli-
cated, which includes segmenting the pattern, unwarping it,
and re-warping onto the target. Thanks to the UV position
map representation, we do not need the unwarping and re-
warping steps. The problem reduces to simple image seg-
mentation.

Given the input texture map T_r^m, we aim to extract a
binary segmentation mask for its makeup patterns. We
can do so by using any segmentation network. In our im-
plementation, a typical UNet structure with a pre-trained
Resnet-50 encoder is used. We employ dice loss for train-
ing: $\mathcal{L}_{\mathcal{DC}} = \frac{2|\Gamma^{gt} \cap \Gamma^{pr}|}{|\Gamma^{gt}| + |\Gamma^{pr}|}$, where Γ^{gt} and Γ^{pr} are the ground
truth and predicted segmentation masks for the pattern.

To train this network, we need a makeup dataset with
annotated masks for the makeup patterns. However, such
datasets do not exist, so we developed ourselves a synthetic
dataset, called CPM-Synt-1, for image-mask pair training
data. Details of this dataset will be discussed in Sec. 4.2.

3.4. Combination

The output of the Pattern Transfer Branch is the pattern
mask Γ^m, while the output of the Color Transfer Branch is
an entire UV texture map T_s^{mC}. The forms of these two
outputs are different, reflecting the fundamental differences
between two makeup categories. That is why we propose
two separate branches for dedicated processing.

To get the UV texture map for the source image with
the desired transferred makeup, we can combine the out-
puts of the two branches, by blending the reference makeup
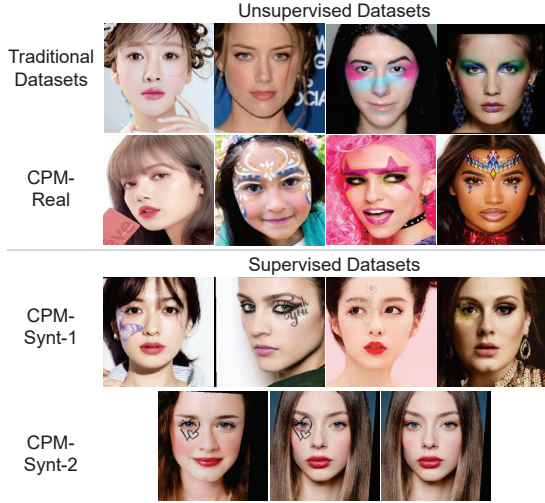pattern, defined by the predicted mask Γ^m, with the color-

Figure 3: Features of makeup datasets. First row are from MT [16] and LADN dataset [10]. The rest are from our datasets: CPM-Real, CPM-Synt-1 and CPM-Synt-2.

transferred texture map $T_s^{m_C}$ from the color transfer branch:

$$T_s^m = T_r^m \odot \Gamma^m + T_s^{m_C} \odot (1 - \Gamma^m). \qquad (5)$$

Finally, we convert this texture map to the output image I_s^m, using the rendering function $I_s^m = \mathcal{UV}^{-1}(S_s, T_s^m)$.

4. Color-&-Pattern Makeup (CPM) Datasets

Given the lack of annotated data with extreme makeup styles for the development of in-the-wild makeup transfer methods, we collected this type of data ourselves. In this section, we describe our data collection and generation procedure that led to three Color and Pattern Makeup (CPM) datasets, called CPM-Real, CPM-Synt-1, and CPM-Synt-2.

4.1. CPM-Real – In-the-Wild Makeup Dataset

This is a dataset of real faces with real in-the-wild makeups. It is very diverse in terms of makeup styles, containing both color and pattern makeups. The degree of makeup can vary from light to heavy, from color-oriented to pattern-driven. Many images contain extreme makeups, including facial gems, face paintings, hennas, and festival makeups.

To compile this dataset, we first retrieved a set of initial images using keyword searches (e.g., glitter makeup, festival makeup, creative makeup, gems makeup, face painting). We then used the MTCNN face detector [31] to detect and crop faces in each image. We discarded faces smaller than 150×150. Finally, we manually removed low-quality and inappropriate ones. The final set has 3895 makeup images, which is 43% larger than the number of makeup images in MT [16], the previously largest available makeup dataset. This dataset is designed purely for testing purposes.

4.2. CPM-Synt-1 – Added Pattern Dataset

This is a dataset of real faces with synthetically added makeup patterns. To build it, we needed a set of makeup
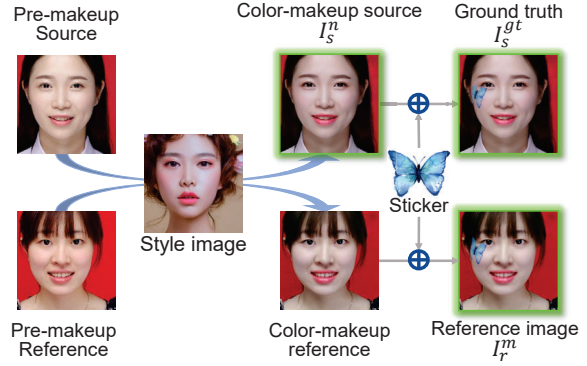

Figure 4: CPM-Synt-2 synthesizing process

	#images	Light	Heavy	Pattern
Unsupervised datasets				
MT [16]	3834	✓		
LADN [10]	698	✓	✓	
M-Wild [12]	772	✓		
CPM-Real (Ours)	3895	✓	✓	✓
Supervised datasets				
CPM-Synt-1 (Ours)	5555	✓	✓	✓
CPM-Synt-2 (Ours)	1625	✓		✓

Table 1: Overview of makeup datasets

patterns. Unfortunately, automatic segmentation of makeup patterns from real images was non-trivial, while manual annotation would be laborious due to the tiny little details in many patterns such as henna. To circumvent this problem, we collected some patterns from the Internet with keyword search (e.g., flowers, crystals, gems, henna, daisy, leaf, tattoo) to compile the so-called Stickers dataset with 577 high-quality images. We only used PNG images that had alpha channels, which could later be used for image blending.

Next, we applied the patterns on images from the MT dataset [16]. As a pattern should conform to the face's surface, we did not directly blend the pattern to the face image. Instead, we applied the blending process in the UV space. The face image was first converted to a UV texture map, as described in Sec. 3.1. We then blended the pattern on the texture map using its alpha mask with random size, location, and opacity. To make the makeup realistic, we set the pattern's size around the cheek size and put its location inside the face but not at the center. Besides creating the blended texture, we also kept the blending mask as ground-truth for training pattern segmentation module (Sec. 3.3). Finally, we rendered the blended texture to get the after-makeup facial image, together with the sticker segmentation mask.

In total, CPM-Synt-1 has 5555 after-makeup images. Each image is associated with the ground truth segmentation mask for the pattern and the corresponding UV maps. This dataset split into disjoint training and testing subsets of size 4182 and 1373, respectively. The subjects and the makeup patterns in two subsets are disjointed.

4.3. CPM-Synt-2 – Transferred Pattern Dataset

Despite having ground-truth labels, CPM-Synt-1 does not follow the transfer setup, so it cannot be used to evaluate the pattern-based makeup transfer algorithms. Hence, we built another synthetic dataset called CPM-Synt-2. This dataset contains image triplets: (source image, reference image, ground-truth), specially designed for the pattern-transferred evaluation task.

One requirement for this test is to have the source and the reference image of the same color-makeup style. Otherwise, we need to impose color-makeup transfer in the ground-truth image. Creating such ground-truth is nontrivial, and no practical solution has been proposed. We can start from non-makeup images, but even these images have a visible cosmetic color difference that requires swapping.

To overcome the mentioned problem, we rely on an assumption of makeup transfer stability: When using the same reference image, a good makeup transfer method will output images of the same makeup style. Based on this assumption, we propose a method to construct the CPM-Synt-2 dataset, as described in Fig. 4. First, two non-makeup images are randomly picked from the MT dataset. Then, we transfer both of them to the same makeup style n, defined by a Color Style image, using BeautyGAN. This process results in two images with the same color style, called Color-makeup Source I_s^n and Color-makeup Reference I_r^n, respectively. Next, we blend the sticker into both images, forming the ground-truth I_s^{gt} and reference image I_r^m. Finally, the triplets (I_s^n, I_r^m, I_s^{gt}) are formed. CPM-Synt-2 consists of 1625 triplets for evaluation purposes.

5. Experiments

5.1. Implementation Details

We implemented our system with PyTorch. The UV conversation function $\mathcal{UV}$ and the inverse rendering module were based on the existing code and model of PRNet [9]. We trained Pattern Transfer Branch and Color Transfer Branch separately, using respective training datasets.

Color Transfer Branch. For fair comparisons with other methods, we trained $\mathcal{C}$ on the MT dataset [16]. We aligned and resized all images to $256{\times}256$ and then computed their texture maps and facial segmentation in the UV space. The color transfer branch was trained in an unsupervised manner; in each iteration, we randomly sampled one makeup and one non-makeup image to form a swapping pair.

The hyper-parameters were set as follows. The weights for the loss components were: $\lambda_{adv}{=}1$, $\lambda_{cyc}{=}10$, $\lambda_{per}{=}0.005$, and $\lambda_{hist}{=}1$. The weights for histogram matching regions were: $\lambda_{skin}{=}0.1$, $\lambda_{eyes}{=}1$, and $\lambda_{lips}{=}1$. Batch size was set to one. We used Adam optimizer with learning rate 0.0002 to train the network until convergence.

Pattern Transfer Branch. This branch was trained with supervised learning, using the CPM-Synt-1 dataset. Each training image came with the pattern segmentation mask, both having size $256{\times}256$. We utilized UNet structure with Resnet-50 as the pre-trained encoder. Since the original segmentation mask was non-binary, we used sigmoid as activation function. The model was trained for 300 epochs with batch size 8, Adam optimizer, and learning rate 0.0001.

5.2. Qualitative experiments

We compared the proposed method to the state-of-the-art methods, including DMT [30], BeautyGAN [16], LADN [10], and PSGAN [12]. We skipped some baselines, such as BeautyGlow [6] and CA-GAN [14] because they are both color-only and have no released model. The evaluations were conducted on both the existing and proposed datasets. We present here a few qualitative results but more examples can be found in the supplementary.

MT dataset. We conducted an experiment on the existing MT dataset [16] to examine the ability to transfer color-based makeup styles. As can be seen in the first row of Fig. 5, our model can capture well the lips' color, similar to the state-of-the-art BeautyGAN [16]. Moreover, thanks to UV-based swapping solution, our method can successfully transfer the face blushes and is the only method that captures the glowing skin foundation.

CPM-Synt-1 dataset. We evaluated the transfer results with the presence of synthesized makeup patterns. For each reference makeup in the test set of CPM-Synt-1, we randomly picked a source image in the MT dataset and do makeup transfer. A representative result is shown in the second row of Fig. 5. Although the makeup pattern was unseen during training, our network could capture its pattern well and transport it to the output. All other methods, including LADN, failed to handle such a complicated style.

CPM-Real dataset. Finally, we tested with real in-the-wild makeup styles. This time, we used the reference makeup in the CPM-Real dataset, while the source image was still from the MT dataset. We present two examples in the last rows of Fig. 5. Although providing realistic results, color-based methods completely ignored facial drawings and decoration. LADN could partially replicate the reference styles, but its results are unnatural and unappealing. Our method could retain the makeup pattern details and return the results that are closest to desirable makeups.

5.3. User surveys

For the subjective evaluation of the results, we conducted a user survey for each dataset above. Each survey consisted of 20 questions. For each question, the participants were asked to rank the after-makeup images from the best to the worst. Subsequently, we assigned a score of 5 to the
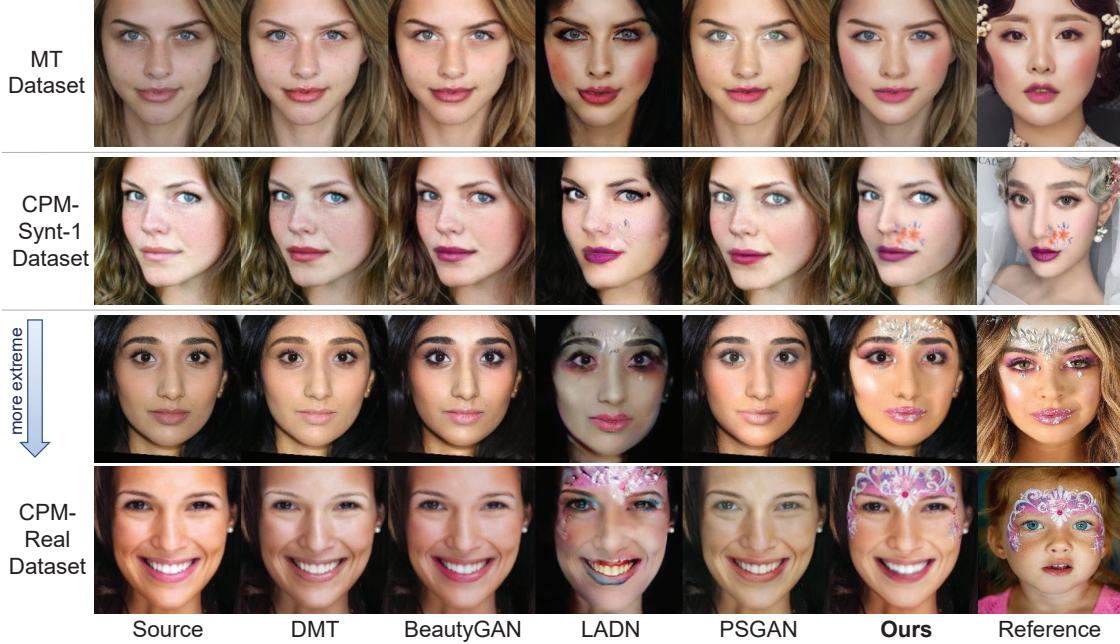
Figure 5: Qualitative Results

Dataset	DMT	BeautyGAN	LADN	PSGAN	Ours
MT [16]	2.99	3.69	2.39	3.25	**4.35**
CPM-Synt-1	3.28	3.50	1.67	2.92	**4.89**
CPM-Real	2.50	2.87	2.24	2.76	**4.60**

Table 2: **User survey results** for the qualitative results from three datasets. The numbers shown are the average user ratings, with 5 being the perfect score and 1 the lowest. Our method achieves the highest scores on all three surveys.

Dataset	Metric	DMT	BGAN	LADN	PSGAN	Ours
Synt-1	mIOU	-	-	-	-	**0.788**
Synt-2	MS-SSIM	0.918	0.918	0.656	0.723	**0.977**

Table 3: **Ground-truth experiments:** Pattern segmentation on CPM-Synt-1 (top row) and Makeup transfer on CPM-Synt-2 (bottom row).

highest-ranked method, and 1 to the lowest one. There were 40 participants, leading to 800 answers for each survey.

The average survey scores are reported in Table 2. Our method outperforms the others by a wide margin on all tests. Its scores are close to the perfect score of 5, suggesting the superiority of our method in almost all questions.

5.4. Ground-truth experiments

By building labelled datasets, we can conduct quantitative experiments that have not been done in the previous studies. We first compute our pattern segmentation network's accuracy, then evaluate the quality of the makeup-transfer results produced by ours and baseline methods.

Makeup pattern segmentation. We evaluated the performance of our pattern segmentation branch on the test set of CPM-Synt-1, and the result is shown in the first row of Table 3. Our pattern segmentation branch achieved 0.788 mIoU. It is not perfect, but sufficiently good for the downstream task of makeup transfer.

Makeup transfer quality. To quantitatively compare our method and other baselines in the makeup-transfer setting, we conducted experiments on the CPM-Synt-2 dataset. We used the MS-SSIM metric to evaluate the quality of the after-makeup images in comparison with ground-truth ones. The average score for each method is reported in the second row of Table 3. The MS-SSIM of our method is 0.977, surpassing the second method by a wide margin.

5.5. Ablation Studies

UV-based makeup transfer. As discussed in Sec. 3.3, the UV representation is critical to the Pattern Branch. Fig. 7 shows in the first row a comparison between pattern in image space and in UV space. Our method aligns the source and target faces pixel-by-pixel, removing the differences in 3D poses, shapes, and expressions. Hence, we can transfer the pattern easily and precisely.

Last row of Fig. 7 compares the makeup transfer results between BeautyGAN, trained on original faces, and our Color Transfer Branch, trained on the UV space. As can be seen, our method replicates the reference style much more accurately. It preserves both the purple eye shadow and the glowing skin foundation.
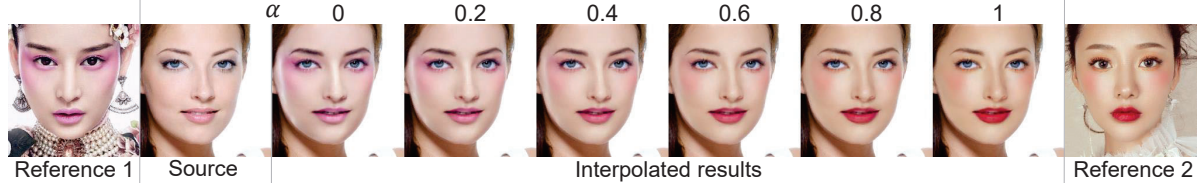
Identity preservation. We used ArcFace [8] to calcu-

Figure 6: **Makeup style interpolation**. The middle images have makeup style interpolated from two reference styles with a mixing parameter α.
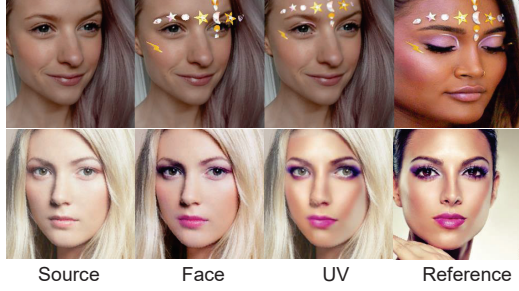

Figure 7: **Benefits of the UV space** to Pattern Transfer Branch (first row) and Color Transfer Branch (second row). From left to right: Source image, results obtained by training on the original image space, results obtained by training on the UV space, and the reference image.

Figure 8: Branch Analysis.

late the similarity score between the faces before and after makeup transfer. The average similarity scores on the MT and CPM-Synt-1 datasets are 0.851 and 0.781, respectively. Based on the recommended face verification threshold of 0.5, our makeup transfer method preserves the identities of the subjects. These similarity scores are lower than the maximum score of 1, but this is expected because real-life facial makeup may also change facial characteristics dramatically.

Branch Analysis. Both color and pattern branches are vital, as illustrated in Fig. 8. The Color Transfer Branch alone failed to bring the face drawings and stickers from the reference image to the target face. When using Pattern Branch only, the lip color of the output image stays the same as the original. We need to combine two branches to replicate all makeup components of the reference image.

5.6. Interpolation and Partial Makeup Transfer

Interpolation. Makeup interpolation is an interesting application of makeup transfer. While interpolating between makeup patterns is not practical, interpolating between makeup colors is pretty common and easy. Given a single
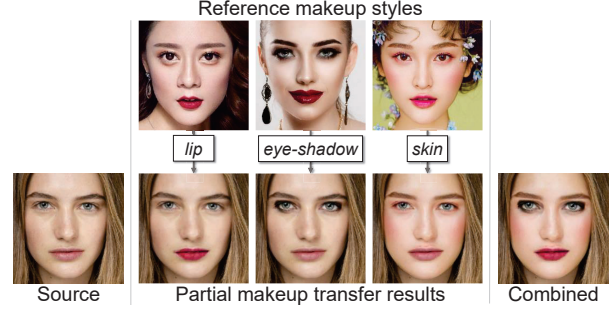

Figure 9: **Partial makeup transfer**. First rows are different styles. Second rows are results from partial makeup transfer only (lip, eye-shadow, skin), and combination of all styles.

input image I_s^n and two reference styles $I_{r_1}^{m_1}$ and $I_{r_2}^{m_2}$, we can run two makeup transfer processes in parallel to get the color-transferred texture maps $T_s^{m_1}$ and $T_s^{m_2}$. We can then mix these texture maps by a mixing parameter $\alpha \in [0, 1]$, and render to get the interpolated output. Fig. 6 displays some interpolated results in case one or two reference styles are given. The results are smooth and natural, even in extreme regions such as heavy eye-shadow and cheek color.

Partial makeup transfer. Further exploiting the UV position map, we can use it together with facial segmentation to perform partial makeup transfer. Instead of transferring makeup on the entire face, we can do it on a face region defined by some input mask. This controllable mechanism was proposed in the previous works [6, 12] and can be easily implemented in our system. Fig. 9 provides an example in which we transferred makeup partially for the lips, eye shadow, and skin region, then generated a makeup composition on the entire face.

6. Conclusion

In this paper, we extend the definition of the makeup transfer task and propose a novel holistic framework to deal with in-the-wild makeup styles. Makeup styles are now interpreted as a combination of color-matching and pattern-addition, respectively, solved by our Color Transfer Branch and Pattern Transfer Branch. UV representation is incorporated to improve the results of both branches. The experiments show our framework can achieve state-of-the-art qualitative and quantitative results. Moreover, we propose novel datasets to leverage makeup-transfer studies and encourage future development.

References

[1] V. Blanz and T. Vetter. Morphable model for the synthesis of 3D faces. In *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*, 1999.

[2] Huiwen Chang, Jingwan Lu, Fisher Yu, and Adam Finkelstein. Pairedcyclegan: Asymmetric style transfer for applying and removing makeup. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[3] Anpei Chen, Zhang Chen, Guli Zhang, Kenny Mitchell, and Jingyi Yu. Photo-realistic facial details synthesis from single image. In *Proceedings of the International Conference on Computer Vision*, 2019.

[4] C. Chen, A. Dantcheva, and A. Ross. Automatic facial makeup detection with application in face recognition. In *International Conference on Biometrics*, 2013.

[5] C. Chen, A. Dantcheva, T. Swearingen, and A. Ross. Spoofing faces using makeup: An investigative study. In *IEEE International Conference on Identity, Security and Behavior Analysis*, 2017.

[6] Hung-Jen Chen, Ka-Ming Hui, Szu-Yu Wang, Li-Wu Tsao, Hong-Han Shuai, and Wen-Huang Cheng. Beautyglow: On-demand makeup transfer framework with reversible generative network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[7] A. Dantcheva, C. Chen, and A. Ross. Can facial cosmetics affect the matching accuracy of face recognition systems? In *IEEE International Conference on Biometrics: Theory, Applications and Systems*, 2012.

[8] Jiankang Deng, Jia Guo, Xue Niannan, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, 2019.

[9] Yao Feng, Fan Wu, Xiaohu Shao, Yanfeng Wang, and Xi Zhou. Joint 3d face reconstruction and dense alignment with position map regression network. In *Proceedings of the European Conference on Computer Vision*, 2018.

[10] Qiao Gu, Guanzhi Wang, Mang Tik Chiu, Yu-Wing Tai, and Chi-Keung Tang. Ladn: Local adversarial disentangling network for facial makeup and de-makeup. In *Proceedings of the International Conference on Computer Vision*, 2019.

[11] Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless, and David H.Salesin. Image analogies. *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*, 2001.

[12] Wentao Jiang, Si Liu, Chen Gao, Jie Cao, Ran He, Jiashi Feng, and Shuicheng Yan. Psgan: Pose and expression robust spatial-aware gan for customizable makeup transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[13] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems*, 2018.

[14] Robin Kips, Pietro Gori, Matthieu Perrot, and Isabelle Bloch. Ca-gan: Weakly supervised color aware gan for controllable makeup transfer. In *ECCV Workshops*, 2020.

[15] C. Li, K. Zhou, and S. Lin. Simulating makeup through physics-based manipulation of intrinsic image layers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[16] 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 *Proceedings of the ACM international conference on Multimedia*, 2018.

[17] P Paysan, R Knothe, B Amberg, S Romhani, and T Vetter. A 3D face model for pose and illumination invariant face recognition. In *Proceedings of the IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2009.

[18] Christian Rathgeb, Antitza Dantcheva, and Christoph Busch. Impact and detection of facial beautification in face recognition: An overview. *IEEE Access*, 2019.

[19] Elad Richardson, Matan Sela, and Ron Kimmel. 3d face reconstruction by learning from synthetic data. In *International Conference on 3D Vision*, 2016.

[20] Sami Romdhani and Thomas Vetter. Efficient, robust and accurate fitting of a 3D morphable model. In *Proceedings of the International Conference on Computer Vision*, 2003.

[21] Ayush Tewari, Michael Zollhofer, Hyeongwoo Kim, Pablo Garrido, Florian Bernard, Patrick Perez, and Christian Theobalt. MoFA: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. In *Proceedings of the International Conference on Computer Vision*, 2017.

[22] Wai-Shun Tong, Chi-Keung Tang, Michael S Brown, and Ying-Qing Xu. Example-based cosmetic transfer. In *Pacific Conference on Computer Graphics and Applications*, 2017.

[23] Anh Tran, Tal Hassner, Iacopo Masi, and Gérard Medioni. Regressing robust and discriminative 3D morphable models with a very deep neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[24] Anh Tuan Tran, Tal Hassner, Iacopo Masi, Eran Paz, Yuval Nirkin, and Gérard G Medioni. Extreme 3d face reconstruction: Seeing through occlusions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[25] Luan Tran and Xiaoming Liu. Nonlinear 3d face morphable model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[26] Thomas Vetter and Volker Blanz. Estimating coloured 3d face models from single images: An example based approach. In *Proceedings of the European Conference on Computer Vision*, 1998.

[27] Wikipedia. Cosmetic industry – Wikipedia, 2020. https://en.wikipedia.org/w/index.php?title=Cosmetic_industry.

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

[29] Haotian Yang, Hao Zhu, Yanru Wang, Mingkai Huang, Qiu Shen, Ruigang Yang, and Xun Cao. Facescape: A large-scale high quality 3d face dataset and detailed riggable 3d face prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[30] Honglun Zhang, Wenqing Chen, Hao He, and Yaohui Jin. Disentangled makeup transfer with generative adversarial network. In *arXiv*, 2019.

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

[32] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the International Conference on Computer Vision*, 2017.

[33] Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z. Li. Face alignment across large poses: A 3D solution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.