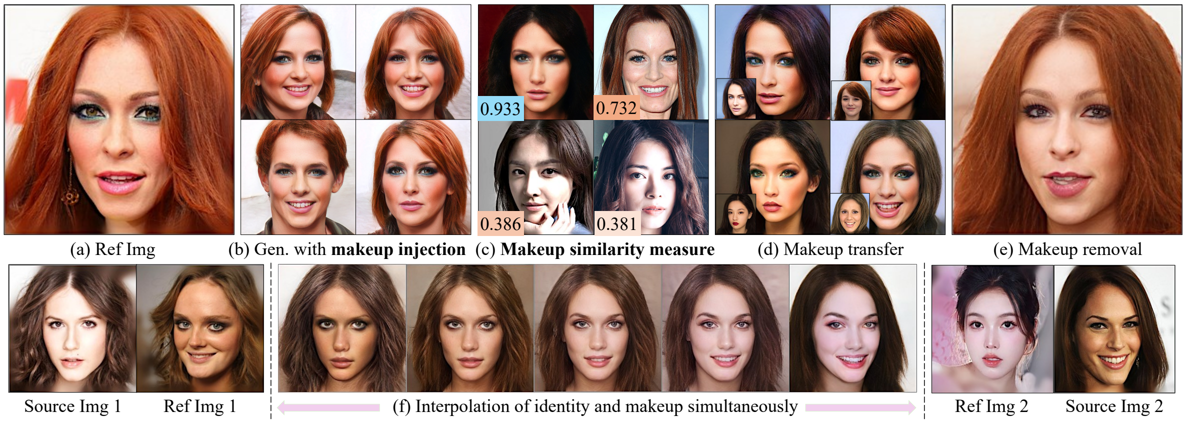
BeautyBank: Encoding Facial Makeup in Latent Space

Qianwen Lu[1,2], Xingchao Yang[1], Takafumi Taketomi[1]
[1]CyberAgent [2]The University of Tokyo
{lu_qianwen_xa, you_koutyo, taketomi_takafumi}@cyberagent.co.jp



(a) Ref Img (b) Gen. with **makeup injection** (c) **Makeup similarity measure** (d) Makeup transfer (e) Makeup removal

Source Img 1 Ref Img 1 (f) Interpolation of identity and makeup simultaneously Ref Img 2 Source Img 2

Figure 1. **Example applications of our makeup encoder (BeautyBank).** We have successfully explored a variety of applications, including using (a) images with reference makeup to (b) generate facial images with makeup injection, (c) measure makeup similarity, and (d) transfer makeup, and (e) remove makeup. Additionally, BeautyBank can utilize two different facial identity references (Source Img 1 and 2) and two different makeup references (Ref Img 1 and 2) to (f) simultaneously interpolate identity and makeup. The images generated using the makeup code from BeautyBank show high-quality details such as makeup colors, patterns, and textures across various makeup applications.

Abstract

The advancement of makeup transfer, editing, and image encoding has demonstrated their effectiveness and superior quality. However, existing makeup works primarily focus on low-dimensional features such as color distributions and patterns, limiting their versatillity across a wide range of makeup applications. Futhermore, existing high-dimensional latent encoding methods mainly target global features such as structure and style, and are less effective for tasks that require detailed attention to local color and pattern features of makeup. To overcome these limitations, we propose BeautyBank, a novel makeup encoder that disentangles pattern features of bare and makeup faces. Our method encodes makeup features into a high-dimensional space, preserving essential details necessary for makeup reconstruction and broadening the scope of potential makeup research applications. We also propose a Progressive Makeup Tuning (PMT) strategy, specifically designed to enhance the preservation of detailed makeup features while preventing the inclusion of irrelevant attributes. We further explore novel makeup applications, including facial image generation with makeup injection and makeup similarity measure. Extensive empirical experiments validate that our method offers superior task adaptability and holds significant potential for widespread application in various makeup-related fields. Furthermore, to address the lack of large-scale, high-quality paired makeup datasets in the field, we constructed the Bare-Makeup Synthesis Dataset (BMS), comprising 324,000 pairs of 512x512 pixel images of bare and makeup-enhanced faces.

1. Introduction

The rapid progress of various generative models, such as GANs and diffusion models, has significantly advanced makeup-related visual tasks [14, 20, 21, 27]. Despite their impressive performance, the algorithms are specifically designed for certain makeup tasks, such as makeup transfer and editing [17, 25, 50, 56, 59]. The primary reason is that they tend to model low-dimensional representations of

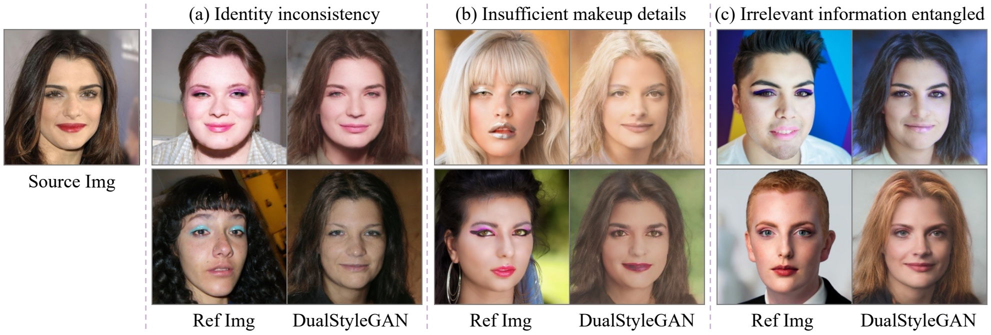

(a) Identity inconsistency
(b) Insufficient makeup details
(c) Irrelevant information entangled
Source Img
Ref Img
DualStyleGAN
Ref Img
DualStyleGAN
Ref Img
DualStyleGAN

Figure 2. **Typical issues in generated images using the baseline method.** When DualStyleGAN [51] is utilized for makeup transfer tasks, the generated images often exhibit inconsistencies in the facial identity compared to the source images. There is also a lack of detail in makeup attributes, such as local colors and patterns, and an entanglement with features that are not related to the makeup pattern.

makeup features, such as color distributions, local details, and pattern styles [25, 32, 50]. Consequently, these methods struggle to handle the diverse and intricate demands of real-world makeup applications, such as facial image generation with makeup injection and makeup similarity measure.

On the other hand, the latent code representation has shown its great performance in image generation, style transfer, and image editing [33, 51, 52]. In paticular, these methods generate high-quality style images by encoding high-dimensional style features and subsequently manipulating latent codes in semantically meaningful ways. It should be noted that these methods primarily focus on global features, including structural elements and overall color styles. However, makeup-related tasks emphasize the consistency of identity features between makeup and bare-face images, as well as the details of local colors and patterns in makeup. Directly applying existing methods to makeup encoding tasks can lead to significant facial identity changes or loss of local makeup details, as shown in Fig. 2 (a) and (b). Additionally, without disentangling makeup-irrelevant information, the generated images also exhibit significant alterations in non-facial areas such as hair and background, as shown in Fig. 2 (c).

In this paper, we propose a novel makeup encoding method that efficiently encodes facial makeup features into a high-dimensional latent space. Our method adapts to various makeup applications while preserving detailed information essential for high-quality makeup reconstruction. We initially introduce BeautyBank, a makeup encoder featuring separate paths for bare-face and makeup styles. During the training of the bare-face style path, we applied a facial enhancement loss to maintain the consistency of identity features in the bare-face code. The refined bare-face code can subsequently improve the makeup style path's ability to encode makeup representations independently. Additionally, we introduce a Progressive Makeup Tuning (PMT) strategy that employs varied training strategies and loss functions at different stages to progressively fine-tune the makeup code. BeautyBank achieves stable makeup encoding, preserves rich makeup detail features, and effectively disentangles unrelated features, such as hair and background, from the makeup encoding process. Furthermore, given the current lack of large-scale, high-quality paired makeup datasets, we construct the Bare-Makeup Synthesis Dataset (BMS), comprising 324,000 pairs of 512x512 pixel bare-face to makeup face images. This dataset provides a diverse array of makeup data for makeup encoding tasks. We generate the makeup data in the BMS dataset using LEDITS++ [6] based on style and color prompts collected from the FFHQ [20] dataset, encompassing a wide variety of makeup styles, colors, and patterns.

In summary, our contributions are threefold:

- We introduce BeautyBank, a novel makeup encoder that effectively disentangles bare-face features from makeup style features. This facilitates the encoding of makeup in a high-dimensional feature space. Our experiments demonstrate that our method expands the range of makeup applications beyond existing methods, enabling facial image generation with makeup injection and makeup similarity measure, as shown in Fig. 1.
- We design the PMT strategy that incrementally fine-tunes makeup encoding. This strategy ensures the preservation of essential makeup detail features, such as color textures, while reducing the influence of makeup-unrelated features.
- We construct the BMS datasaet, a large-scale, high-resolution makeup dataset that ensures diversity in makeup encoding. To our knowledge, this is the first large-scale dataset of its kind, consisting of paired 512x512 pixel images of bare and made-up faces. We will make this dataset publicly available and hope it can assist future makeup-related research.

2. Related Work

2.1. Facial Makeup Tasks

Facial makeup is an important aspect of human appearance. In computer vision and graphics, mainstream research focuses on makeup transfer [7–9, 13, 17, 18, 22, 25, 26, 28, 32, 40, 41, 48–50, 59], 3D makeup [16, 24, 30, 38, 54–56], and face verification [15, 39].

The task of makeup transfer is transferring a makeup pattern in a specified reference face image to a source face image. Early research focused on the color distribution of makeup [25], while more recent studies attempt to transfer complex makeup patterns [59]. In addition, several studies have analyzed factors in facial images, which allows for makeup transfer to accommodate variations such as lighting [56], occlusion [28], and head pose [17, 50]. However, most methods are limited to low resolutions, such as 256×256. 3D makeup research primarily focuses on the 3D makeup estimation or the beautification and stylization of avatars [5]. Tasks related to makeup in face verification [15, 39] underscore the importance of security and face protection. They achieve this by adding makeup to faces, thereby generating images that aid in privacy protection. It's also worth noting that research dedicated specifically to makeup recommendation is somewhat limited [4].

Although certain image generation models provide the option to generate makeup images, they typically treat makeup as a unified face feature, without offering control over its type and style [34, 45, 46, 57]. Recent studies have combined CLIP [35] or diffusion model [14] to generate high-quality images with a certain level of makeup control [5, 6, 31, 39, 44]. However, these language-based makeup image generation methods cannot precisely control makeup details, and often, the same prompt does not produce consistent makeup results.

Our method aims to encode facial makeup to obtain disentangled makeup features. Our makeup encoding can be applied to various applications and expand makeup-related research, enabling new tasks such as enhanced facial image generation with makeup injection and makeup similarity measure.

2.2. StyleGAN-based Stylized Portrait

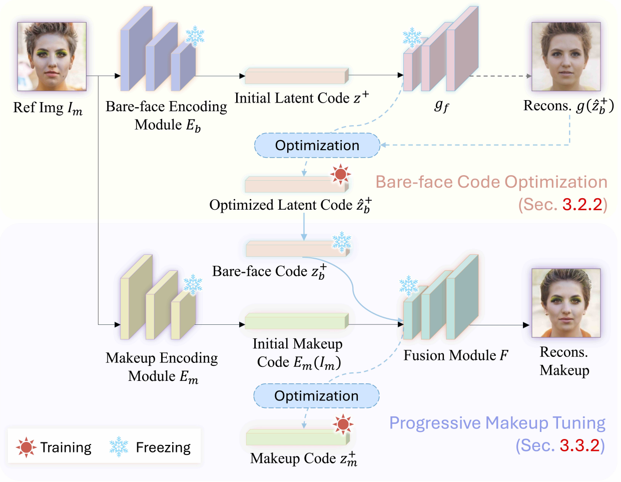

Ref Img I_m
Bare-face Encoding Module E_b
Initial Latent Code z^+
g_f
Recons. $g(\hat{z}_b^+)$
Optimization
Optimized Latent Code $\hat{z}_b^+$
Bare-face Code Optimization (Sec. 3.2.2)
Bare-face Code z_b^+
Makeup Encoding Module E_m
Initial Makeup Code $E_m(I_m)$
Fusion Module F
Recons. Makeup
Optimization
Makeup Code z_m^+
Progressive Makeup Tuning (Sec. 3.3.2)
Training
Freezing

Figure 3. **The workflow of latent code optimization.** We enhance the encoding of identity information to optimize the bare-face code (see Section 3.2.2 for details). Subsequently, based on the encoded bare-face code, we use the specially designed objective function to enhance the encoding of makeup details and avoid encoding features unrelated to the makeup, achieving the final makeup encoding (see Section 3.3.2 for details).

Stylized portrait generation has seen significant advancements [23, 33, 51, 52, 58], particularly through the use of the StyleGAN model [20, 21] for high-resolution image generation and flexible style control. Approaches like Toonify [33] fine-tune a pre-trained StyleGAN on cartoon datasets, combining layers from the fine-tuned and original models to generate cartoon-like faces. The pSp method [36] trains an encoder to project real face images into cartoon faces, while DualStyleGAN [51] adds an extrinsic style path for exemplar-based style transfer. StyleGAN-NADA [12] uses CLIP to guide StyleGAN into new artistic domains without real cartoon datasets, enabling text-driven toonification. StyleGAN inversion techniques [1–3, 11, 36, 37, 42, 43, 46, 47, 53] further enhance these capabilities by projecting real face images into StyleGAN's latent space for editing.

In contrast to the challenges faced by stylized portrait methods, such as misalignment caused by artistic styles, our focus is on realistic face images, specifically within two domains: bare-face and makeup-face. We build upon the StyleGAN-based stylized portrait framework [51] and leverage StyleGAN inversion techniques to capture high-dimensional representations of makeup.

3. Methodology

Our objective is to develop an enhanced model, named BeautyBank (in Section 3.1), which is inspired by DualStyleGAN [51]. It encodes makeup to cater to a broader range of makeup-related applications. Our core idea involves incorporating prior knowledge of identity encoding and makeup as supervision, extracting the bare-face code of makeup portraits (in Section 3.2). Building on the bare-face code, we employ a progressive fine-tuning strategy specifically designed to optimize makeup codes, preserving more detailed makeup features and reducing unrelated information. (in Section 3.3). The workflow is illustrated in Fig. 3.

3.1. BeautyBank

Drawing from the network architecture of DualStyleGAN [51], BeautyBank is designed to extract bare-face and makeup features. It includes two independent style paths—a bare-face style path and a makeup style path—along with a fusion module F.

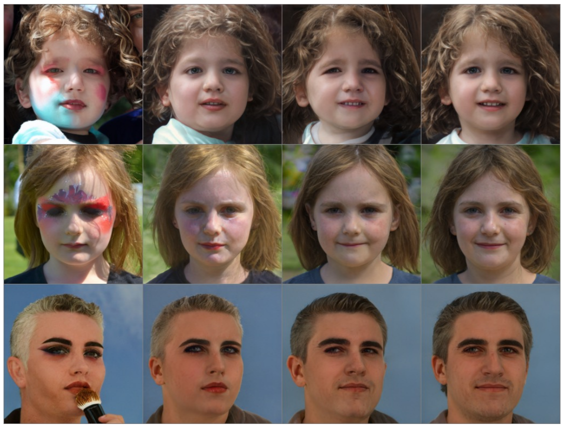
The bare-face style path features a bare-face encoding module E_b, constructed based on the pSp encoder [36], which maps the input facial features to Z_+ space. This initial latent code z^+ ($z^+ = E_b(I)$) is refined to obtain the bare-face code z_b^+ ($z_b^+ \in \mathbb{R}^{18\times512}$), capturing facial identity and structural features. The input image I can be replaced with the reference makeup image I_m if there is no corresponding bare-face image available. Similar to the bare-face style path, the makeup style path incorporates a makeup encoding module, E_m, also constructed based on the pSp encoder, which maps makeup features of I_m to Z_+ space. This results in an initial makeup code, $E_m(I_m)$, that prepares for subsequent makeup encoding of I_m. E_b and E_m are both pretrained on the FFHQ dataset. The fusion module F incorporates two mapping networks for z_b^+ (the bare-face style path) and $E_m(I_m)$ (the makeup style path) separately, and a synthesis network to fuse the two latent codes after mapping. This module generates facial images that merge identity features from z_b^+ with makeup features from $E_m(I_m)$. After refining z_b^+, we further optimizes the initial makeup code to obtain the final makeup code z_m^+ ($z_m^+ \in \mathbb{R}^{18\times512}$), which allows for more flexible control over specific makeup features (color and structural features) of the generated image. The style adjustment parameter w ($w \in \mathbb{R}^{18}$), used in F, serves as a weight vector for the flexible blending of style features from z_b^+ and z_m^+, and is preset to 1. When w is set to 0, F degrades to a standard StyleGAN generator g for face generation.

3.2. Identity-Optimized Bare-face Encoding

Bare-face encoding aims to disentangle bare face features from the reference makeup image I_m to guide the subsequent encoding and reconstruction of makeup features. In this section, we first provide a concise introduction to DualStyleGAN [51] in Section 3.2.1, which outlines the methodology for facial destylization. We then present a detailed explanation of our bare-face code optimization method in Section 3.2.2.

3.2.1 Overview of DualStyleGAN

Our bare-face encoding method is an extension of the facial destylization approach proposed in DualStyleGAN [51]. To balance between face realism and fidelity to the portraits, DualStyleGAN proposes a multi-stage destylization method to obtain an intrinsic style code containing facial structure features.



(a) Ref Img I_m (b) $g(E_b(I_m))$ (c) $g(\hat{z}_b^+)$ (d) $g(z_b^+)$

Figure 4. **Example of bare-face encoding.** Bare-face encoding results from (a) in the preliminary stage (in Section 3.2.1) are shown in (b), while results from bare-face code optimization (in Section 3.2.2) are shown in (c) and (d). Bare-face encoding progressively disentangles the makeup information contained in (a) while maintaining consistent identity features.

Initially, DualStyleGAN performs the latent initialization of an artistic portrait. Due to the robustness of Z_+ space compared to W_+ space in handling background details unrelated to the face and distorted shapes, the encoder E_b is utilized to encode the artistic portrait into the Z_+ space. Then the initial reconstructed facial image is generated using g, which is pretrained on FFHQ.

Subsequently, the latent codes are refined to better match the facial structures. Although the output of g at this stage, as shown in Fig. 4 (b), closely resembles the original face due to g's limitations in fully reconstructing the artistic portrait, certain artistic style features are encoded into Z_+. Therefore, DualStyleGAN performs the latent code optimization, and then applies the latent code of g_f back to g to achieve the transfer from the artistic portrait domain to the original face domain. g_f is obtained by further fine-tuning g using makeup images from the BMS dataset. For more details, please refer to the paper [51].

3.2.2 Bare-face Code Optimization

In the process of latent code optimization (as shown in Fig. 3), although DualStyleGAN incorporates an identity loss, inconsistencies remain between the identity features of the reconstructed images and the reference makeup (as discussed in Section 4.4). This discrepancy is primarily attributed to the facial recognition model used (ArcFace [10]), which does not focus exclusively on the facial region, thereby impacting the accuracy of identity matching. To mitigate the effects of inaccurately encoded bare-face codes on subsequent makeup encoding, we im-

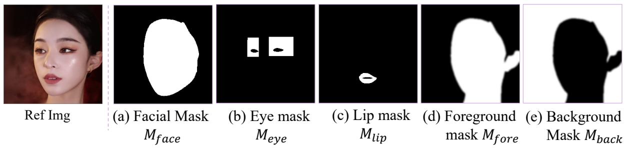

Ref Img
(a) Facial Mask M_{face}
(b) Eye mask M_{eye}
(c) Lip mask M_{lip}
(d) Foreground mask M_{fore}
(e) Background Mask M_{back}

Figure 5. **Example of the Masks Utilized in BeautyBank.** During Bare-face Code Optimization, the objective function employs mask (a) (in Section 3.2.2). Stage 1 of Progressive Makeup Tuning utilizes masks (a), (b), and (c), while Stage 2 employs masks ranging from (a) to (f) (in Section 3.3.2).

prove the focus on identity features within the facial region during the optimization of the bare-face code. This is achieved by employing a facial mask (M_{face}) as shown in Fig. 5 (a) and integrating it into the objective function. Specifically, we introduce a facial enhancement loss $L_{fm}(g_f(z^+), I_m, M_{face}) = \|(I_m - g_f(z^+)) \odot M_{face}\|_1$, where $\odot$ denotes the Hadamard product. This calculates the loss for the facial mask M_{face} region. The full objective function for optimizing the latent encoding is

$$\begin{aligned} L_b = &\lambda_{p_1} L_{perc}(g_f(z^+), I_m) + \lambda_{id} L_{id}(g_f(z^+), I_m) \\ &+ \lambda_{fm_1} L_{fm}(g_f(z^+), I_m, M_{face}) + \|\sigma(z^+)\|_1, \end{aligned}$$

where L_{perc} denotes perceptual loss [19], L_{id} is the identity loss [10], and $\sigma(z^+)$ represents the standard error of 18 different 512-dimension vectors in z^+, to avoid overfitting during training. The parameters λ_{p_1}, λ_{id}, λ_{fm_1} are set to 1, 0.1, and 0.0001, respectively. By minimizing L_b, we obtain the optimized latent $\hat{z}_b^+$.

Since g_f is a model fine-tuned on the BMS dataset and g is pre-trained on FFHQ, they can be regarded as image generators for the makeup domain and bare-face domain, respectively. Therefore, using the optimized $\hat{z}_b^+$, we obtain $g(\hat{z}_b^+)$ as a bare face image that has removed makeup and retains facial features from I_m. The reconstructed facial image by g is shown in Fig. 4 (c). Finally, we use the encoder E_b to encode this bare face image, obtaining the bare-face code, $z_b^+ = E_b(g(\hat{z}_b^+))$. Fig. 4 (d) shows the reconstructed facial image of z_b^+.

Furthermore, as the BMS dataset contains paired data of bare faces I_b and makeup I_m, encoding makeup within the BMS dataset simply requires the use of $z_b^+ = E_b(I_b)$ to obtain the bare-face code. However, for in-the-wild makeup images that lack paired data, the aforementioned bare-face encoding process is still necessary.

3.3. Conditional Fine-Tuning Makeup Encoding

To obtain a high-dimensional makeup code enriched with detailed makeup information, we perform the pre-training and fine-tuning of BeautyBank, as discussed in Section 3.3.1, and implement the Progressive Makeup Tuning (PMT) strategy for makeup encoding optimization, as outlined in Section 3.3.2.

3.3.1 Pre-training and fine-tuning of BeautyBank.

Following DualStyleGAN [51], we conduct pre-training and fine-tuning of the fusion module in BeautyBank to prepare for makeup encoding. To ensure stable and smooth model training, we initially performed the pre-training of the fusion module using the FFHQ dataset. This stage is implemented through color transfer and structural transfer training. Color transfer can stabilize the network parameters without deviating from the original generative space, achieving color migration within the original generative space. Structural transfer involves style mixing operations in the intermediate layers, ensuring the effective capturing and mimicking of detailed structural features while maintaining the color style. To enable the fusion module to utilize the bare-face code and the makeup code to generate facial images in the makeup domain, we fine-tune the fusion module using facial images from the BMS dataset. Specifically, we input paired bare-face code z_b^+ and initial makeup code $E_m(I_m)$ into the fusion module to reconstruct facial makeup. The objective function for this stage is

$$L_{m_1} = \lambda_{adv} L_{adv} + \lambda_{p_2} L_{perc} + \lambda_{sty} L_{sty} + \lambda_{con_1} L_{con},$$

where parameters λ_{adv}, λ_{p_2}, λ_{sty}, and λ_{con_1} are set to 1. L_{adv}, L_{sty}, and L_{con} denote adversarial loss, style loss, and contextual loss [29], respectively. The parameters λ_{adv}, λ_{p_2}, λ_{sty}, and λ_{con_1} are set to 1.

3.3.2 Progressive Makeup Tuning

To better encode essential makeup details and disentangle urelated features, we introduce the Progressive Makeup Tuning (PMT) strategy to optimize the initial makeup code. PMT consists of two stages.

(Stage 1) Detail-Oriented Latent Optimization: To optimize the makeup detail encoding, we fix the parameters of BeautyBank and fine-tune the makeup code. During this fine-tuning stage, the fusion module in BeautyBank receives paired inputs of the bare-face code and the optimized makeup code. It then reconstructs makeup images to calculate the loss necessary for latent optimization. In the objective function, we incorporate prior knowledge of face parsing to enhance feature extraction in makeup-concentrated regions (overall face, eyes, lips) of facial images. We apply the objective function

$$\begin{aligned} L_{m_{2-1}} = &\lambda_{p_3} L_{perc} + \lambda_{con_2} L_{con} + \lambda_{fm_2} L_{fm} \\ &+ \lambda_{pm_1} L_{pm} + \lambda_{em_1} L_{em} + \lambda_{lm_1} L_{lm}, \end{aligned}$$

where L_{pm}, L_{em}, and L_{lm} are the perceptual loss of utilizing the facial mask M_{face} in Fig. 5 (a), eye mask M_{eye} in

Fig. 5 (b), and lip mask M_{lip} in Fig. 5 (c). The parameters λ_{p_3}, λ_{con_2}, λ_{fm_2}, λ_{pm_1}, λ_{em_1}, and λ_{lm_1} are set to 1, 1, 0.0001, 100, 100, 100, respectively.

(Stage 2) Non-Makeup Features Disentanglement: To disentangle makeup-unrelated features (e.g., background, hair color), we further optimize the makeup code. We conduct training using different sources of bare-face code z_b^+ and makeup code z_m^+, and replace L_{perc} with $\lambda_{pf} L_{pf} + \lambda_{pb} L_{pb}$ in the objective function of the previous stage:

$$\begin{aligned} L_{m_{2-2}} = \lambda_{pf} L_{pf} + \lambda_{pb} L_{pb} + \lambda_{fm_3} L_{fm} + \lambda_{con_3} L_{con} \\ + \lambda_{pm_2} L_{pm} + \lambda_{em_2} L_{em} + \lambda_{lm_2} L_{lm}, \end{aligned}$$

where L_{pf}, L_{pb} represent the perceptual loss of utilizing masks for facial areas, M_{fore}, in Fig. 5 (d), and masks for non-facial areas, M_{back}, in Fig. 5 (e). In this stage, the output of BeautyBank is a facial image with face and background features from the bare-face code and makeup features from the makeup code. This avoids the inclusion of makeup-unrelated features in the makeup code. The parameters λ_{pf}, λ_{pb}, λ_{fm_3}, λ_{con_3}, λ_{pm_2}, λ_{em_2}, and λ_{lm_2} are set to 100, 100, 0.0001, 1, 100, 100, 100, respectively.

Through PMT, BeautyBank achieves bare-face and makeup encoding for 1412 makeup styles. This makeup encoding can be widely applied to various makeup tasks, such as generating faces with specific makeup, makeup transfer, and makeup similarity measure, discussed in Section 5.

4. Experiments

4.1. Bare-Makeup Synthesis Dataset

We utilized a pretrained diffusion method LEDITS++ [6] to create a large-scale bare-makeup synthesis dataset, Bare-Makeup Synthesis Dataset (BMS). The construction process primarily involves two steps:

First, inspired by Stable-Makeup [59], we employed GPT-4 to generate 400 style prompts using the template “make it {} makeup”. However, upon testing these prompts, we found that the generated makeup samples lacked diversity in patterns and colors. Therefore, we used the template “{} makeup with {} on the face” to generate 410 style prompts with GPT-4. To further enhance the diversity of prompts, we constructed 20 color prompts (e.g., Red, Blue, etc.). Ultimately, we created 16,200 prompt pairs by combining the 810 style prompts with the 20 color prompts, which were used to guide the LEDITS++ model in synthesizing makeup data.

Second, we used the FFHQ dataset as the bare skin data to synthesize the corresponding makeup data. For each prompt, we randomly selected 20 facial images from the FFHQ dataset as source images for makeup rendering.

Consequently, we constructed the BMS dataset, comprising 324,000 pairs of 512x512 pixel bare-makeup facial images. It should be noted that even when using identical prompts, LEDITS++ cannot produce consistent makeup results. As shown in Fig. 6, using the same style prompt “make it fairy makeup” and the color prompt “Blue”, the generated makeup looks are significantly different. This demonstrates that the prompt code cannot be used as the makeup embedding.

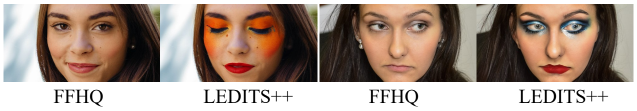


Figure 6. **Examples of generated makeup images using LEDITS++ with text prompt.** Despite using the same style prompt ’make it fairy makeup’ and the color prompt ’Blue’, the generated images exhibit markedly different colors and pattern details.

4.2. Experimental Setup

We conducted the training of BeautyBank using the PMT strategy. The training was performed on 4 NVIDIA Tesla T4 GPUs, with a batch size of 2 per GPU. For bare-face encoding, the number of training iterations for g_f was 600, and the number of iterations for optimizing the encoding was 300. In makeup encoding, the number of iterations for each stage of PMT was 300 and 300, respectively. The bare-face images used for training were sourced from the FFHQ dataset, and the makeup images were sourced from the BMS dataset and BeautyFace dataset [49].

Our developed BeautyBank can encode a wide variety of makeup styles. Currently, we have encoded 1412 makeup codes using BeautyBank, all of which are derived from the BMS dataset and BeautyFace dataset. Utilizing these makeup codes, we can perform various makeup-related tasks (in Section 5), demonstrating the versatility and flexibility of BeautyBank in practical applications. To further expand the application scope of BeautyBank, we plan to encode additional makeup codes in future work to support more diverse makeup image tasks.

4.3. Comparison with SOTA

We performed comprehensive comparisons with the most representative makeup transfer algorithms, including PSGAN [17] SCGAN [9], EleGANt [50], BeautyRec [49], CSD-MT [41], and Stable-Makeup [59]. As shown in Fig. 7, our results demonstrate more stable performance across various makeup references.

Besides, we conducted a user study to quantitatively evaluate the generation quality and transfer accuracy of different models. We randomly selected 20 pairs of bare-face images from the FFHQ dataset and makeup images from the BMS dataset and BeautyFace dataset, producing 20 makeup transfer result images. A total of 15 participants were asked to evaluate these samples in three aspects: “visual quality”,

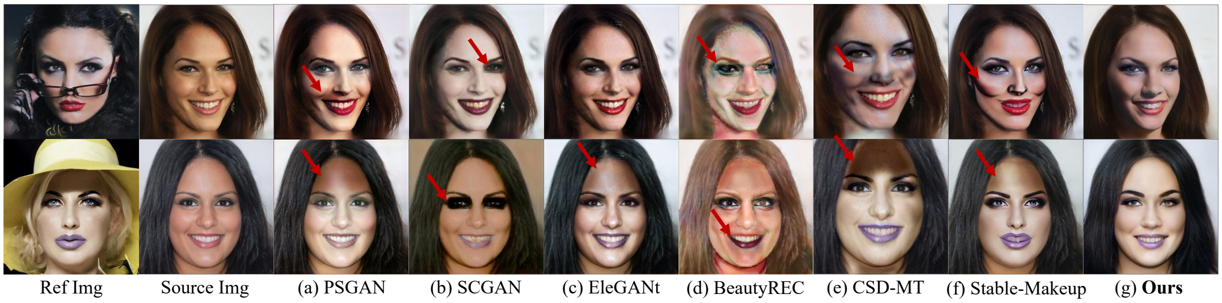


Figure 7. **Qualitative comparison of different methods.** Our results outperform other methods in terms of color and detail.

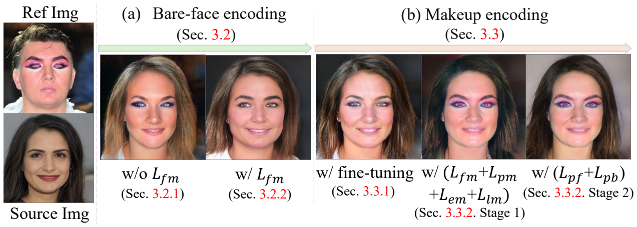


Figure 8. **Ablation study.** Figure (a) illustrates the ablation study of each stage in bare-face encoding (in Section 3.2), while Figure (b) shows the ablation study of each stage in makeup encoding (in Section 3.3).

Table 1. **Comparison of different methods based on Quality, Detail, and Overall performance.** Our method received the highest (best) scores across all criteria.

Criteria	PSGAN [17]	SCGAN [9]	EleGANt [50]	BeautyRec [49]	CSD-MT [41]	Stable-Makeup [59]	**BeautyBank (Ours)**
Quality↑	0.00%	0.00%	20.00%	0.00%	0.00%	6.67%	**73.33%**
Detail↑	0.00%	0.00%	40.00%	0.00%	0.00%	13.33%	**46.67%**
Overall↑	0.00%	0.00%	26.67%	0.00%	0.00%	6.67%	**66.67%**

"detail processing" (the precision of transferred details), and "overall performance" (the visual quality, the fidelity of transferred makeup, etc.). Participants were requested to select the best set of results for each aspect. Table 1 shows the results of the user study (ratio (%) selected as the best). Our BeautyBank outperformed other methods in all aspects. It should be noted that our evaluation data includes reference makeup images with extensive occlusions and shadows, as we aim to evaluate the stability of performance under various conditions.

4.4. Ablation Study

This section demonstrates the effectiveness of bare-face encoding and makeup encoding by showcasing results on makeup image generation and makeup transfer tasks. As shown in Fig. 8, our results demonstrate more stable performance across various makeup references.

Bare-face encoding: Fig. 8 (a) shows the performance in the makeup transfer task before and after adding L_{fm} during the optimization stage. Without L_{fm}, the loss of identity features is more pronounced under the same number of iterations. Additionally, it is worth noting that the makeup transfer results shown in this section are all generated by BeautyBank after completing the stage 1 of PMT.

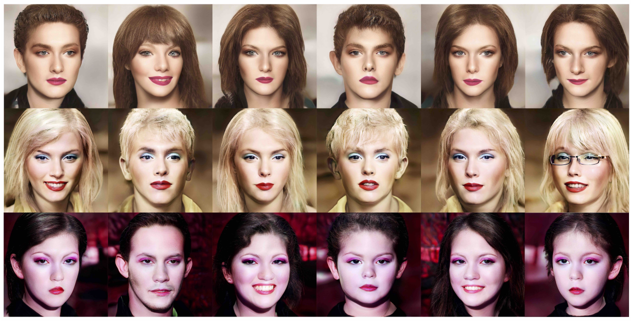


Figure 9. **Examples of makeup facial generation with makeup injection.** We replace the bare-face code with random Gaussian noise as input to BeautyBank, generating facial images with the same makeup but varying in gender, expressions, hairstyles, and face shapes.

Makeup encoding: Fig. 8 (b) presents the results from BeautyBank training along with stages 1 and 2 of PMT in the makeup transfer task. With the addition of the detail-enhanced objective function, BeautyBank can fully transfer the color and pattern of the makeup. After further latent optimization, as the makeup code contains fewer makeup-unrelated features, BeautyBank can better preserve the hair and background color of the source image.

5. Applications

To explore the effectiveness of our method, we evaluated our makeup encoding on several makeup-related applications.

Makeup facial generation with makeup injection: We randomly selected several sets of encoded makeup codes, and for each makeup code, we generated random Gaussian noises to replace the bare-face code. Subsequently, we used the fusion module of BeautyBank for facial image genera-

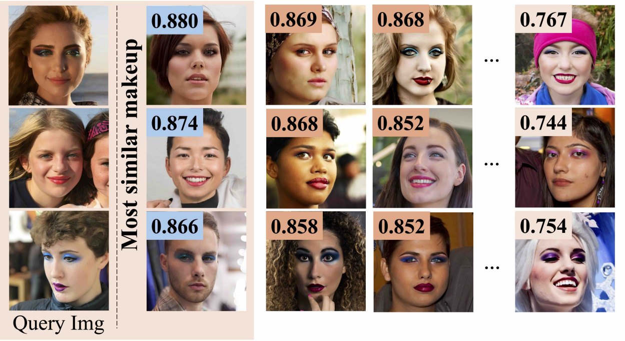


Figure 10. **Examples of makeup similarity measure with reference makeup.** By searching the encoded makeup database and calculating the cosine similarity with the makeup code of the query image, we can identify the makeup style most similar to the query image.

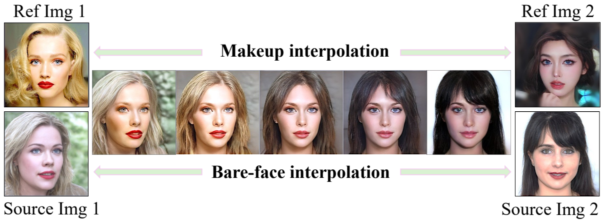


Figure 11. **Makeup interpolation application.** BeautyBank can separately encode the bare-face code (Source Img 1 and 2) and the makeup code (Ref Img 1 and 2), supporting interpolation between different sets of bare-face and makeup images.

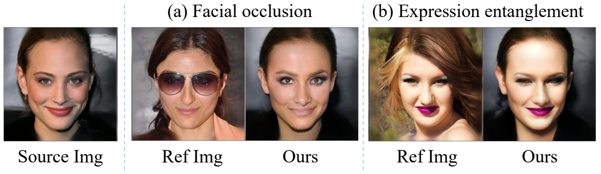


Figure 12. **Limitations of BeautyBank.** The makeup images generated by Beauty perform poorly in cases of extensive facial occlusion, or exhibit entangled expression information due to the limitations of the image encoder.

tion. Fig. 9 illustrates the results of our facial image generation. The figure indicates that by altering the input random noise, we can generate faces with various expressions, poses, genders, and hairstyles, while retaining the specified makeup.

Makeup similarity measure: As shown in Fig. 10, by calculating and ranking the cosine similarity between makeup codes, we can retrieve similar makeup styles from the encoded makeup database. The examples shown are from the 1412 encoded makeup styles. With more makeup encoded, more accurate and similar results can be obtained.

Makeup transfer: As shown in Fig. 7, BeautyBank can perform makeup transfer by utilizing the bare-face code from the source image and the makeup code from the reference makeup image. The generated images using BeautyBank are overall more natural and realistic, with rich colors and detailed features in the makeup.

Makeup removal: As shown in Fig. 4 (d), BeautyBank can generate bare-skin facial images with preserved identity features by performing bare-face encoding of the input makeup image I_m.

Makeup interpolation: Demonstrated in Fig. 1 (f) and 11, since BeautyBank includes two style paths, we can achieve seamless interpolation between different source images and reference makeup styles by interpolating between the bare-face codes or between the makeup codes.

6. Conclusion

In this study, we introduced BeautyBank, a novel makeup encoding approach that significantly expands the application possibilities in the field of makeup. We also developed the Bare-Makeup Synthesis Dataset (BMS) and the Progressive Makeup Tuning (PMT) strategy, which enhance the extraction and refinement of makeup codes. Extensive empirical testing confirms that our approach not only improves the adaptability of makeup tasks but also opens up new avenues for innovative applications such as makeup injection and similarity measure. We believe these advancements set a new standard for future research and applications in makeup-related technologies.

As illustrated in Fig. 12, although our model demonstrates robustness in accurately encoding makeup from reference images with partial facial occlusions, significant occlusions can lead to incorrect encoding in these areas. Moreover, accurately estimating natural skin tone from images with makeup presents challenges, primarily because most makeup applications include a foundation layer. Consequently, our methodology assumes that the input facial images already have foundation applied. Additionally, due to the variability in iris color—which may be natural or altered by cosmetic lenses—we do not categorize it as unrelated to makeup. Therefore, both the foundation color and iris color in our generated results are closely aligned with the reference makeup. Furthermore, the accuracy of our makeup encoding process, which utilizes the pSp encoder [36], is constrained by the capabilities of this model. Challenges such as effectively disentangling facial expressions or avoiding identity shifts during the encoding process may occur. Moving forward, we plan to explore the use of higher-quality facial encoders and develop specialized methods aimed at more effectively disentangling expressions while preserving identity features to overcome these limitations.

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space. In *ICCV*, pages 4431–4440. IEEE, 2019. 3

[2] Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. Restyle: A residual-based stylegan encoder via iterative refinement. In *ICCV*, pages 6691–6700. IEEE, 2021. 3

[3] Yuval Alaluf, Omer Tov, Ron Mokady, Rinon Gal, and Amit Bermano. Hyperstyle: Stylegan inversion with hypernetworks for real image editing. In *CVPR*, pages 18490–18500. IEEE, 2022. 3

[4] Taleb Alashkar, Songyao Jiang, Shuyang Wang, and Yun Fu. Examples-rules guided deep neural network for makeup recommendation. In *AAAI*, pages 941–947, 2017. 3

[5] Shivangi Aneja, Justus Thies, Angela Dai, and Matthias Nießner. ClipFace: text-guided editing of textured 3d morphable models. In Erik Brunvand, Alla Sheffer, and Michael Wimmer, editors, *SIGGRAPH 2023*, pages 70:1–70:11. ACM, 2023. 3

[6] Manuel Brack, Felix Friedrich, Katharina Kornmeier, Linoy Tsaban, Patrick Schramowski, Kristian Kersting, and Apolinário Passos. LEDITS++: limitless image editing using text-to-image models. 2024. 2, 3, 6

[7] Huiwen Chang, Jingwan Lu, Fisher Yu, and Adam Finkelstein. PairedCycleGAN: Asymmetric style transfer for applying and removing makeup. In *CVPR*, pages 40–48. Computer Viion Foundation / IEEE Computer Society, 2018. 3

[8] Hung-Jen Chen, Ka-Ming Hui, Szu-Yu Wang, Li-Wu Tsao, Hong-Han Shuai, and Wen-Huang Cheng. BeautyGlow: Ondemand makeup transfer framework with reversible generative network. In *CVPR*, pages 10042–10050, 2019. 3

[9] Han Deng, Chu Han, Hongmin Cai, Guoqiang Han, and Shengfeng He. Spatially-invariant style-codes controlled makeup transfer. In *CVPR*, pages 6549–6557, 2021. 3, 6, 7

[10] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. ArcFace: Additive angular margin loss for deep face recognition. In *CVPR*, pages 4690–4699. IEEE, 2019. 4, 5

[11] Tan M. Dinh, Anh Tuan Tran, Rang Nguyen, and Binh-Son Hua. Hyperinverter: Improving stylegan inversion via hypernetwork. In *CVPR*, pages 11379–11388. IEEE, 2022. 3

[12] Rinon Gal, Or Patashnik, Haggai Maron, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. StyleGAN-NADA: clip-guided domain adaptation of image generators. *ACM Trans. Graph.*, 41(4):141:1–141:13, 2022. 3

[13] 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 *ICCV*, pages 10480–10489, 2019. 3

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 3

[15] Shengshan Hu, Xiaogeng Liu, Yechao Zhang, Minghui Li, Leo Yu Zhang, Hai Jin, and Libing Wu. Protecting facial privacy: Generating adversarial identity masks via style-robust makeup transfer. In *CVPR*, pages 14994–15003, 2022. 3

[16] Cheng-Guo Huang, Wen-Chieh Lin, Tsung-Shian Huang, and Jung-Hong Chuang. Physically-Based Cosmetic Rendering. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, page 190, 2013. 3

[17] 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 *CVPR*, pages 5193–5201, 2020. 1, 3, 6, 7

[18] Qiaoqiao Jin, Xuanhong Chen, Meiguang Jin, Ying Cheng, Rui Shi, Yucheng Zheng, Yupeng Zhu, and Bingbing Ni. Toward tiny and high-quality facial makeup with data amplify learning. 2024. 3

[19] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *ECCV*, volume 9906, pages 694–711. Springer, 2016. 5

[20] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, pages 4401–4410, 2019. 1, 2, 3

[21] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *CVPR*, pages 8107–8116, 2020. 1, 3

[22] Robin Kips, Pietro Gori, Matthieu Perrot, and Isabelle Bloch. CA-GAN: Weakly supervised color aware GAN for controllable makeup transfer. In *ECCV*, volume 12537, pages 280–296, 2020. 3

[23] Dongyeun Lee, Jae Young Lee, Doyeon Kim, Jaehyun Choi, Jaejun Yoo, and Junmo Kim. Fix the noise: Disentangling source feature for controllable domain translation. In *CVPR*, pages 14224–14234. IEEE, 2023. 3

[24] Chen Li, Kun Zhou, and Stephen Lin. Simulating makeup through physics-based manipulation of intrinsic image layers. In *CVPR*, pages 4621–4629, 2015. 3

[25] 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 International Conference on Multimedia*, pages 645–653, 2018. 1, 2, 3

[26] Si Liu, Xinyu Ou, Ruihe Qian, Wei Wang, and Xiaochun Cao. Makeup like a superstar: Deep localized makeup transfer network. In *IJCAI*, pages 2568–2575, 2016. 3

[27] Xudong Liu, Ruizhe Wang, Hao Peng, Minglei Yin, Chih-Fan Chen, and Xin Li. Face beautification: Beyond makeup transfer. In *Frontiers in Computer Science*, volume 4. Frontiers, 2022. 1

[28] Yueming Lyu, Jing Dong, Bo Peng, Wei Wang, and Tieniu Tan. SOGAN: 3D-aware shadow and occlusion robust GAN for makeup transfer. In *Proceedings of International Conference on Multimedia*, pages 3601–3609, 2021. 3

[29] Roey Mechrez, Itamar Talmi, and Lihi Zelnik-Manor. The contextual loss for image transformation with non-aligned data. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *ECCV*, volume 11218, pages 800–815. Springer, 2018. 5

[30] Mohit Mendiratta, Xingang Pan, Mohamed Elgharib, Kartik Teotia, Mallikarjun B. R., Ayush Tewari, Vladislav

Golyanik, Adam Kortylewski, and Christian Theobalt. AvatarStudio: text-driven editing of 3d dynamic human head avatars. *ACM Trans. Graph.*, 42(6):226:1–226:18, 2023. 3

[31] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *CVPR*, pages 6038–6047. IEEE, 2023. 3

[32] Thao Nguyen, Anh Tuan Tran, and Minh Hoai. Lipstick ain't enough: Beyond color matching for in-the-wild makeup transfer. In *CVPR*, pages 13305–13314, 2021. 2, 3

[33] Justin N. M. Pinkney and Doron Adler. Resolution dependent GAN interpolation for controllable image synthesis between domains. abs/2010.05334, 2020. 2, 3

[34] Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, and Supasorn Suwajanakorn. Diffusion autoencoders: Toward a meaningful and decodable representation. In *CVPR*, pages 10609–10619. IEEE, 2022. 3

[35] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *ICML*, volume 139, pages 8748–8763. PMLR, 2021. 3

[36] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: A stylegan encoder for image-to-image translation. In *CVPR*, pages 2287–2296, 2021. 3, 4, 8

[37] Daniel Roich, Ron Mokady, Amit H. Bermano, and Daniel Cohen-Or. Pivotal tuning for latent-based editing of real images. *ACM Trans. Graph.*, 42(1):6:1–6:13, 2023. 3

[38] Kristina Scherbaum, Tobias Ritschel, Matthias Hullin, Thorsten Thormählen, Volker Blanz, and Hans-Peter Seidel. Computer-suggested Facial Makeup. *Computer Graphics Forum*, 30(2), 2011. 3

[39] Fahad Shamshad, Muzammal Naseer, and Karthik Nandakumar. Clip2protect: Protecting facial privacy using text-guided makeup via adversarial latent search. In *CVPR*, pages 20595–20605, 2023. 3

[40] Zhaoyang Sun, Yaxiong Chen, and Shengwu Xiong. SSAT: A symmetric semantic-aware transformer network for makeup transfer and removal. In *AAAI*, pages 2325–2334, 2022. 3

[41] Zhaoyang Sun, Shengwu Xiong, Yaxiong Chen, and Yi Rong. Content-style decoupling for unsupervised makeup transfer without generating pseudo ground truth. 2024. 3, 6, 7

[42] Ayush Tewari, Mohamed Elgharib, Mallikarjun B. R., Florian Bernard, Hans-Peter Seidel, Patrick Pérez, Michael Zollhöfer, and Christian Theobalt. PIE: portrait image embedding for semantic control. *ACM Trans. Graph.*, 39(6):223:1–223:14, 2020. 3

[43] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *ACM Trans. Graph.*, 40(4):133:1–133:14, 2021. 3

[44] Linoy Tsaban and Apolinário Passos. LEDITS: real image editing with DDPM inversion and semantic guidance. abs/2307.00522, 2023. 3

[45] Tengfei Wang, Yong Zhang, Yanbo Fan, Jue Wang, and Qifeng Chen. High-fidelity GAN inversion for image attribute editing. In *CVPR*, pages 11369–11378. IEEE, 2022. 3

[46] Tengfei Wang, Yong Zhang, Yanbo Fan, Jue Wang, and Qifeng Chen. High-fidelity GAN inversion for image attribute editing. In *CVPR*, pages 11369–11378. IEEE, 2022. 3

[47] Weihao Xia, Yulun Zhang, Yujiu Yang, Jing-Hao Xue, Bolei Zhou, and Ming-Hsuan Yang. GAN inversion: A survey. *TPAMI*, 45(3):3121–3138, 2023. 3

[48] Jianfeng Xiang, Junliang Chen, Wenshuang Liu, Xianxu Hou, and Linlin Shen. RamGAN: Region attentive morphing GAN for region-level makeup transfer. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *ECCV*, volume 13682, pages 719–735, 2022. 3

[49] Qixin Yan, Chunle Guo, Jixin Zhao, Yuekun Dai, Chen Change Loy, and Chongyi Li. BeautyREC: Robust, efficient, and component-specific makeup transfer. In *CVPRW*, pages 1102–1110, 2023. 3, 6, 7

[50] Chenyu Yang, Wanrong He, Yingqing Xu, and Yang Gao. EleGANt: Exquisite and locally editable GAN for makeup transfer. In *ECCV*, 2022. 1, 2, 3, 6, 7

[51] Shuai Yang, Liming Jiang, Ziwei Liu, and Chen Change Loy. Pastiche master: Exemplar-based high-resolution portrait style transfer. In *CVPR*, pages 7683–7692. IEEE, 2022. 2, 3, 4, 5

[52] Shuai Yang, Liming Jiang, Ziwei Liu, and Chen Change Loy. Vtoonify: Controllable high-resolution portrait video style transfer. *ACM Trans. Graph.*, 41(6):203:1–203:15, 2022. 2, 3

[53] Shuai Yang, Liming Jiang, Ziwei Liu, and Chen Change Loy. Styleganex: Stylegan-based manipulation beyond cropped aligned faces. In *ICCV*, pages 20943–20953. IEEE, 2023. 3

[54] Xingchao Yang and Takafumi Taketomi. BareSkinNet: Demakeup and De-lighting via 3D Face Reconstruction. *Computer Graphics Forum*, 41(7):623–634, 2022. 3

[55] Xingchao Yang, Takafumi Taketomi, Yuki Endo, and Yoshihiro Kanamori. Makeup prior models for 3D facial makeup estimation and applications. In *CVPR*, 2024. 3

[56] Xingchao Yang, Takafumi Taketomi, and Yoshihiro Kanamori. Makeup extraction of 3D representation via illumination-aware image decomposition. *Computer Graphics Forum*, 42(2):293–307, 2023. 1, 3

[57] Xu Yao, Alasdair Newson, Yann Gousseau, and Pierre Hellier. A latent transformer for disentangled face editing in images and videos. In *ICCV*, pages 13769–13778, 2021. 3

[58] Junzhe Zhang, Yushi Lan, Shuai Yang, Fangzhou Hong, Quan Wang, Chai Kiat Yeo, Ziwei Liu, and Chen Change Loy. Deformtoon3d: Deformable neural radiance fields for 3d toonification. In *ICCV*, pages 9110–9120. IEEE, 2023. 3

[59] Yuxuan Zhang, Lifu Wei, Qing Zhang, Yiren Song, Jiaming Liu, Huaxia Li, Xu Tang, Yao Hu, and Haibo Zhao. Stable-makeup: When real-world makeup transfer meets diffusion model. abs/2403.07764, 2024. 1, 3, 6, 7