HairCLIP: Design Your Hair by Text and Reference Image

Tianyi Wei[1], Dongdong Chen[2,†], Wenbo Zhou[1], Jing Liao[3],
Zhentao Tan[1], Lu Yuan[2], Weiming Zhang[1], Nenghai Yu[1]
[1]University of Science and Technology of China [2]Microsoft Cloud AI [3]City University of Hong Kong
{bestwty@mail., welbeckz@, tzt@mail., zhangwm@, ynh@}ustc.edu.cn
cddlyf@gmail.com, jingliao@cityu.edu.hk, luyuan@microsoft.com

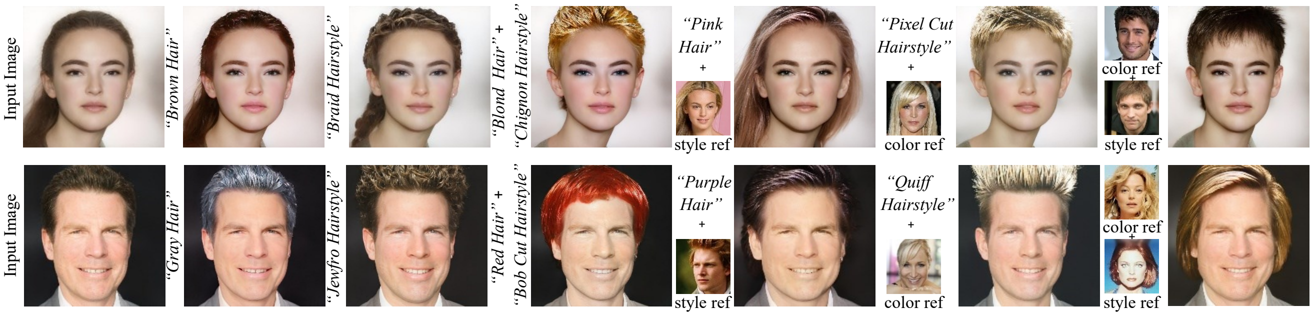

Figure 1. Our framework supports hairstyle and color editing individually or jointly, and conditions may be either image or text.

Abstract

Hair editing is an interesting and challenging problem in computer vision and graphics. Many existing methods require well-drawn sketches or masks as conditional inputs for editing, however these interactions are neither straightforward nor efficient. In order to free users from the tedious interaction process, this paper proposes a new hair editing interaction mode, which enables manipulating hair attributes individually or jointly based on the texts or reference images provided by users. For this purpose, we encode the image and text conditions in a shared embedding space and propose a unified hair editing framework by leveraging the powerful image text representation capability of the Contrastive Language-Image Pre-Training (CLIP) model. With the carefully designed network structures and loss functions, our framework can perform high-quality hair editing in a disentangled manner. Extensive experiments demonstrate the superiority of our approach in terms of manipulation accuracy, visual realism of editing results, and irrelevant attribute preservation.

1. Introduction

Human hair, as the critical yet challenging component of the face, has long attracted the interest of researchers. In recent years, with the development of deep learning, many conditional GAN-based hair editing methods [26, 40, 50] can produce satisfactory editing results. Most of these methods use well-drawn sketches [20, 40, 50] or masks [26, 40] as the input of image-to-image translation networks to produce the manipulated results.

However, we think that these interaction types are not intuitive or user-friendly enough. For example, in order to edit the hairstyle of one image, users often need to spend several minutes to draw a good sketch, which greatly limits the large-scale, automated use of these methods. We therefore wonder "*Can we provide another more intuitive and convenient interaction way, just like human communication behaviors?*". And the language (or"text") naturally meets our requirements.

Benefiting from the development of cross-modal vision and language representations [28, 37, 38], text-guided image manipulation has become possible. Recently, StyleCLIP [31] has achieved amazing image manipulation results by leveraging the powerful image text representation capabilities of CLIP [32]. CLIP has an image encoder and a text encoder, by joint training on 400 million image text pairs, they can measure the semantic similarity between an input image and a text description. Based on this observation, StyleCLIP proposes to use them as the loss supervision to make the manipulated results match the text condition.

Although StyleCLIP inherently supports text description based hair editing, they are not exactly suitable for our task. It suffers from the following drawbacks: 1) For each spe-

[†] Dongdong Chen is the corresponding author. Our code is available at https://github.com/wty-ustc/HairCLIP

cific hair editing description, it needs to train a separate mapper, which is not flexible in real applications; 2) The lack of tailored network structure and loss design makes the method poorly disentangled for hairstyle, hair color, and other unrelated attributes; 3) In practical applications, some hairstyles or colors are difficult to describe in text. At this time, users may prefer to use reference images, but StyleCLIP does not support reference image based hair editing.

To overcome the aforementioned limitations, we propose a hair editing framework that simultaneously supports different texts or reference images as the hairstyle/color conditions within one model. Generally, we follow StyleCLIP and utilize the StyleGAN [24] pre-trained on a large-scale face dataset as our generator, and then the key is to learn a mapper network to map the input conditions into corresponding latent code changes. But different from StyleCLIP, we explore the potential of CLIP to go beyond measuring image text similarity, along with some new designs: 1) *Shared Condition Embedding*. To unify the text and image conditions into the same domain, we leverage the text encoder and image encoder of CLIP to extract their embedding as the conditions for the mapper network respectively. 2) *Disentangled Information Injection*. We explicitly separate hairstyle and hair color information and feed them into different sub hair mappers corresponding to their semantic levels. This helps our method achieve disentangled hair editing; 3) *Modulation Module*. We design a conditional modulation module to accomplish the direct control of input conditions on latent codes, which improves the manipulation ability of our method.

Since our goal is to achieve the hair editing based on the text or reference image condition while ensuring other irrelevant attributes unchanged, three types of losses are introduced: 1) Text manipulation loss is used to guarantee the similarity between the editing result and the given text description; 2) Image manipulation loss is used to guide hairstyle or hair color transfer from the reference image to the target image; 3) Attribute preservation loss is used to keep irrelevant attributes (e.g., identity and background) unchanged before and after editing.

Quantitative and qualitative comparisons and user study demonstrate the superiority of our method in terms of manipulation accuracy, manipulation fidelity, and irrelevant attribute preservation. And some example editing results are shown in Figure 1. We also conduct extensive ablation analysis and well justify the designs of our network structure and loss function.

To summarize, our contributions are three-fold as below:

- We push the frontiers of interactive hair editing, i.e., unifying text and reference image conditions within one framework. It supports a wide range of text and image conditions in one single model without the need of training many independent models, which has never been achieved before.
- In order to perform various hairstyle and hair color manipulation in a disentangled manner, we propose some new network structure designs and loss functions tailored for our task.
- Extensive experiments and analysis are conducted to show the better manipulation quality of our method and the necessity of each new design.

2. Related Work

Generative Adversarial Networks. Since being proposed by Goodfellow *et al*. [11], GANs have made great progress in terms of loss functions [3, 4], network structure design [12,35,39], and training strategies [13,42]. As a representative GAN in the field of image synthesis, StyleGAN [23,24] can synthesize very high-fidelity human faces with realistic facial details and hair. As the typical unconditional GANs, StyleGAN itself is difficult to achieve controllable image synthesis effects. But fortunately, its latent space demonstrates promising disentanglement properties [8, 10, 18, 36], and many works utilize StyleGAN to perform image manipulation tasks [2, 29, 31, 45, 48]. In this paper, we convert the unconditional StyleGAN into our conditional hair editing network with the help of CLIP's powerful image text representation capability. Moreover, we unify the text and reference image condition in one framework and achieve disentangled editing effects.

Image-based Hair Manipulation. As an important part of the human face, hair has attracted many works dedicated to hair modeling [5, 6, 15] and synthesis [21, 26, 47, 52]. Some works [26, 52] use mask which explicitly decouples facial attributes including hair as the conditional input for image-to-image translation networks to accomplish hair manipulation. There are also several works [40, 50] that use sketches as input to depict the structure and shape of the desired hairstyle. However, such interactions are still relatively costly for users. To enable easier interaction, MichiGAN [40] supports hair transfer by extracting the orientation map of one hairstyle reference image as well as the appearance from another hair color reference image. However, MichiGAN is easy to fail for arbitrary shape changes during hair transfer. Recently, LOHO [34] performs a two-stage optimization in the $\mathcal{W}+$ space and noise space of StyleGANv2 [24] to complete the hair transfer for a given reference image. However, the area optimized by this method is limited to the foreground, which requires blending the reconstructed foreground with the original background and often brings obvious artifacts. Besides, it is very time-consuming, e.g., several minutes to optimize an image.

Text-based Hair Manipulation. Along with the booming development of cross-modal visual and language representations [28, 37, 38, 51], especially the powerful CLIP [32], many recent efforts [7,19,31,44,49] start to study text based manipulation. However, there is no existing method specifically tailored for hair editing. Among these works, the most relevant ones are StyleCLIP [31] and TediGAN [49]. But StyleCLIP needs to train a separate mapper network for each specific hair editing description, which is not flexible for real applications. For TediGAN, it proposes two approaches: TediGAN-A encodes text and image separately into the latent space of StyleGAN and completes manipulation with style-mixing, which is less decoupled and difficult to complete hair editing; TediGAN-B conducts the manipulation with optimization using CLIP to provide text-image similarity, but the lack of knowledge learned from a large dataset makes the process unstable and time-consuming.

Different from existing works, this paper presents the first unified framework that enables the text and image conditions simultaneously. This provides a more intuitive and convenient interaction mode, and enables diverse text and image conditions within one single model. Besides, benefiting from the new designs tailored for this task, our method also shows much better hair manipulation quality.

3. Proposed Method

3.1. Overview

Imagine we are in a barbershop and if someone wants to design his hair, the common interaction would be to name the desired hairstyle or provide the hairstylist with a corresponding picture. Inspired by this, we think empowering the AI algorithms to enable such an intuitive and efficient interaction mode is really needed. Thanks to the great image synthesis quality of StyleGAN [23, 24] and the excellent image/text representation ability of CLIP [32], we are finally able to design such a unified hair editing framework to achieve this goal. Before diving into the framework details, we briefly introduce StyleGAN and CLIP respectively.
StyleGAN [23, 24] can synthesize high-resolution, high-fidelity realistic images with a progressive upsample network from noises. Its synthesis process involves multiple latent spaces. $\mathcal{Z} \in \mathbb{R}^{512}$ is the original noise space of StyleGAN. A randomly sampled noise vector $z \in \mathcal{Z}$ is transformed to the $\mathcal{W} \in \mathbb{R}^{512}$ latent space after 8 fully connected layers. Several studies [8,10,18,36] have demonstrated that StyleGAN spontaneously learns to encode rich semantics within its $\mathcal{W}$ space during training, and thus $\mathcal{W}$ exhibits good semantic decoupling properties. In addition, some recent StyleGAN inversion works [1,33,48] extend $\mathcal{W}$ space to $\mathcal{W}+$ space for better reconstruction. For a StyleGAN with 18 layers, it is defined by the cascade of 18 different 512-dimensional vectors $[w_1, ..., w_{18}], w_i \in \mathcal{W}$.
CLIP [32] is a multi-modality model pretrained from 400 million image-text pairs collected from the Internet. It consists of one image encoder and one text encoder that will encode the image and text into the 512-dimensional embedding vector, respectively. It adopts the typical contrastive learning framework, which minimizes the cosine distance between the encoded vectors of the correct image text pairs and maximizes the cosine distance of the incorrect pairs. Benefiting from large-scale pretraining, CLIP can well measure the semantic similarity between an image and a text, via learning one shared image-text embedding space.

3.2. HairCLIP

Inspired by the pioneering work StyleCLIP [31], we utilize the powerful synthesis ability of the pretrained StyleGAN, and aim to learn an extra mapper network to achieve the hair editing function. More specifically, given the real image to edit, we first use the StyleGAN inversion method "e4e" [43] to get its latent code w in the $\mathcal{W}+$ space, then use the mapper network to predict the latent code change Δw based on w and editing conditions (including hairstyle condition e_s and hair color condition e_c). Finally, the modified latent code $w' = w + \Delta w$ will be fed back into the pretrained StyleGAN to get the target editing result. The overall pipeline is illustrated in Figure 2, and each component will be elaborated below.
Shared Condition Embedding. To unify the conditions from the text and image domains under one framework, we naturally choose to represent them by embedding them in the joint latent space of CLIP. For the user-supplied text hairstyle prompt and text hair color prompt, we use CLIP's text encoder to encode them into 512-dimensional conditional embedding, which are denoted as e_s^t and e_c^t respectively. Similarly, the hairstyle reference image and hair color reference image are encoded by the image encoder of CLIP and denoted as $e_s^{I_r}$ and $e_c^{I_r}$ respectively. Because CLIP is well trained on large-scale image-text pairs, $e_s^t, e_c^t, e_s^{I_r}, e_c^{I_r}$ all reside in the shared latent space, thus can be fed into one mapper network and flexibly switched.
Disentangled Information Injection. As demonstrated in many works [23, 49], different layers of StyleGAN correspond to different semantic levels of information in the generated images, with the more preceding layers corresponding to higher semantic levels of information. Following the StyleCLIP [31], we adopt three sub hair mappers M_c, M_m, M_f with the same network structure, which are responsible for predicting Δw of hair editing corresponding to different parts (coarse, medium and fine) of the latent code $w = (w_c, w_m, w_f)$. More specifically, w_c, w_m, w_f correspond to the high semantic level, the middle semantic level, and the low semantic level respectively.

Noticing this semantic layering phenomenon in StyleGAN, we propose disentangled information injection, which aims to improve the decoupling ability of the network for hairstyle and hair color editing. In detail, we use the embedding of hairstyle information $e_s \in \{e_s^t, e_s^{I_r}\}$ from

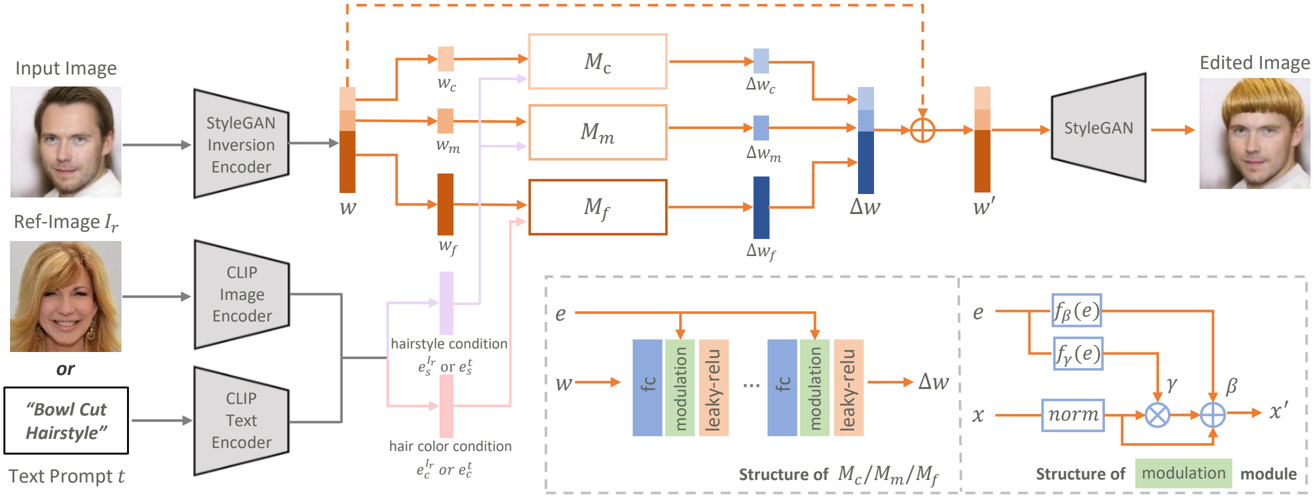

Figure 2. The overview of our framework, here we show an example with hairstyle description text and hair color reference image as conditional inputs. We achieve the corresponding hair editing according to the given reference images and texts, where images, texts are encoded by CLIP's image encoder, text encoder to 512-dimensional vectors as conditional inputs for the hair mapper. Only three sub hair mappers are trainable, where M_c and M_m take the hairstyle conditional input e_s and M_f takes the hair color conditional input e_c.

CLIP as the conditional input for M_c and M_m, and the embedding of hair color information $e_c \in \{e_c^t, e_c^{I_r}\}$ from CLIP as the conditional input for M_f. This is based on the empirical observation that hairstyle often corresponds to middle and high level semantic information in StyleGAN while hair color corresponds to low level semantic information. Therefore, the hair mapper M can be formulated as:

$$M(w, e_s, e_c) = (M_c(w_c, e_s), M_m(w_m, e_s), M_f(w_f, e_c)). \tag{1}$$

Modulation Module. As shown in Figure 2, each sub hair mapper network follows a simple design and consists of five blocks, and each block consists of one fully connected (fc) layer, one newly designed modulation module, and one non-linear activation layer (leakly relu). Rather than simply concatenating the condition embedding with the input latent code, the modulation module uses the condition embedding e to modulate the intermediate output x of the preceding fc layer. Mathematically, it follows the below formulation:

$$x' = (1 + f_\gamma(e))\frac{x - \mu_x}{\sigma_x} + f_\beta(e), \tag{2}$$

where μ_x and σ_x denote the mean and standard deviation of x respectively. And f_γ and f_β are implemented with simple fully connected networks (two fc layers with one intermediate layernorm and leaky relu layer). This design is motivated by recent conditional image translation works [16, 30, 41]. During testing, if no conditional input is provided for hairstyle or hair color, then all modulation modules in the corresponding sub hair mapper will be implemented as identity functions, which is denoted as $e_s = 0$ or $e_c = 0$. In this way, we flexibly support users to edit only hairstyle, only hair color, or both hairstyle and hair color.

3.3. Loss Functions

Our goal is to manipulate the hair in a decoupled manner based on the conditional input, while requiring other irrelevant attributes (e.g., background, identity) well preserved. Therefore, we specifically design three types of loss functions to train the mapper networks: text manipulation loss, image manipulation loss, and attribute preservation loss.

Text Manipulation Loss. In order to perform the corresponding hair manipulation based on the text prompt of the hairstyle or color, we design the text manipulation loss $\mathcal{L}_t$ with the help of CLIP as follows:

$$\mathcal{L}_t = \mathcal{L}_{st}^{clip} + \mathcal{L}_{ct}^{clip}. \tag{3}$$

For the hairstyle text manipulation loss, we measure the cosine distance between the manipulated image and the given text in the CLIP's latent space:

$$\mathcal{L}_{st}^{clip} = 1 - cos(E_i(G(w + M(w, e_s^t, e_c))), e_s^t), \tag{4}$$

where $cos(\cdot)$ means cosine similarity, E_i represents the image encoder of CLIP, G represents the pretrained StyleGAN generator, $e_s^t = E_t(st)$ denotes the embedding of a given hairstyle description text st which is encoded by the text encoder E_t of CLIP, and $e_c \in \{e_c^t, e_c^{I_r}, 0\}$. Similarly, color text manipulation loss is defined as follows:

$$\mathcal{L}_{ct}^{clip} = 1 - cos(E_i(G(w + M(w, e_s, e_c^t))), e_c^t), \tag{5}$$

where e_c^t denotes the embedding of a given color description text which is encoded by the text encoder of CLIP, and $e_s \in \{e_s^t, e_s^{I_r}, 0\}$.

Image Manipulation Loss. Given a reference image, we want the manipulated image to possess the same hairstyle as that of the reference image. However characterizing the similarity between two hairstyles is a challenging task. Exploiting the powerful potential of CLIP again, we encode them separately using CLIP's image encoder to measure their similarity in CLIP's latent space:

$$\mathcal{L}_{si} = 1 - cos(E_i(\mathbf{x}_M * P_h(\mathbf{x}_M)), E_i(\mathbf{x} * P_h(\mathbf{x}))), \quad (6)$$

where the manipulated image $\mathbf{x}_M = G(w + M(w, e_s^{I_r}, e_c))$, $e_s^{I_r} = E_i(\mathbf{x} * P_h(\mathbf{x}))$, $e_c \in \{e_c^t, e_c^{I_r}, 0\}$, P denotes the pre-trained facial parsing network [27], $P_h(\mathbf{x}_M)$ represents the mask of the hair region of $\mathbf{x}_M$, and $\mathbf{x}$ means the given reference image. Thanks to this supervision we propose, our method can yield plausible editing results for cases where the reference image and the input image are seriously misaligned, which is currently unavailable for other hairstyle transfer methods. Also, for reference image based hair color manipulation, we calculate the average color difference in the hair area between reference image and manipulated image as the loss:

$$\mathcal{L}_{ci} = ||avg(\mathbf{x}_M * P_h(\mathbf{x}_M)) - avg(\mathbf{x} * P_h(\mathbf{x}))||_1, \quad (7)$$

where $\mathbf{x}_M = G(w + M(w, e_s, e_c^{I_r}))$, $e_c^{I_r} = E_i(\mathbf{x} * P_h(\mathbf{x}))$, and $e_s \in \{e_s^t, e_s^{I_r}, 0\}$. In summary, the image manipulation loss $\mathcal{L}_i$ is defined as:

$$\mathcal{L}_i = \lambda_{si}\mathcal{L}_{si} + \lambda_{ci}\mathcal{L}_{ci}, \quad (8)$$

where λ_{si}, λ_{ci} are set to 5, 0.02 respectively by default.

Attribute Preservation Loss. To ensure identity consistency before and after hair editing, the identity loss is applied as follows:

$$\mathcal{L}_{id} = 1 - cos(R(G(w + M(w, e_s, e_c))), R(G(w))), \quad (9)$$

where $e_s \in \{e_s^t, e_s^{I_r}, 0\}$, $e_c \in \{e_c^t, e_c^{I_r}, 0\}$, R is a pre-trained ArcFace [9] network for face recognition and $G(w)$ denotes the reconstructed real image. In addition, we designed $\mathcal{L}_{s\_mc}$ in the same way as $\mathcal{L}_{ci}$ in order to maintain the hair color when only manipulating the hairstyle:

$$\mathcal{L}_{s\_mc} = ||avg(\mathbf{x}_M * P_h(\mathbf{x}_M)) - avg(\mathbf{x}_w * P_h(\mathbf{x}_w))||_1, \quad (10)$$

where $\mathbf{x}_M = G(w + M(w, e_s, e_c))$, $e_s \in \{e_s^t, e_s^{I_r}\}$, $e_c = 0$, and $\mathbf{x}_w = G(w)$. Empirically, we find the hairstyle can be well preserved when only changing the color, so we do not add corresponding preservation loss.

Moreover, we introduced background loss with the help of facial parsing network [27] :

$$\mathcal{L}_{bg} = ||(\mathbf{x}_M - \mathbf{x}_w) * (P_{nh}(\mathbf{x}_M) \cap P_{nh}(\mathbf{x}_w))||_2, \quad (11)$$

where $\mathbf{x}_M = G(w + M(w, e_s, e_c))$, $P_{nh}(\mathbf{x}_M)$ represents the mask of the non-hair region of $\mathbf{x}_M$. In this way, we largely ensure that the non-relevant attribute regions remain unchanged. For the same purpose, the L_2 norm of the manipulation step in the latent space is utilized:

$$\mathcal{L}_{norm} = ||M(w, e_s, e_c)||_2. \quad (12)$$

The overall attribute preservation loss $\mathcal{L}_{ap}$ is defined as:

$$\mathcal{L}_{ap} = \lambda_{id}\mathcal{L}_{id} + \lambda_{s\_mc}\mathcal{L}_{s\_mc} + \lambda_{bg}\mathcal{L}_{bg} + \lambda_{norm}\mathcal{L}_{norm}, \quad (13)$$

where λ_{id}, λ_{s_mc}, λ_{bg}, λ_{norm} are set to 0.3, 0.02, 1, 0.8 respectively by default.

Finally, the overall loss function is defined as:

$$\mathcal{L} = \lambda_t\mathcal{L}_t + \lambda_i\mathcal{L}_i + \lambda_{ap}\mathcal{L}_{ap}, \quad (14)$$

where λ_t, λ_i, λ_{ap} are set to 2, 1, 1 respectively by default.

4. Experiments

Implementation Details. We train and evaluate our hair mapper on the CelebA-HQ dataset [22]. Since we use e4e [43] as our inversion encoder, we follow its division of the training set and test set. The StyleGAN2 [24] pre-trained on the FFHQ dataset [23] is used as our generator. For the text input, we collected 44 hairstyle text descriptions and 12 hair color text descriptions; The CelebA-HQ dataset is used to provide reference images of hairstyles or hair colors, and we also generated several edited images using our text-guided hair editing method to augment the diversity of the reference image set. During training, the hair mapper is randomly tasked to edit only the hairstyle or only the hair color or both hairstyle and hair color depending on the provided conditional input. The conditioned input is randomly set as text or reference image. Regarding the training strategy, the base learning rate is 0.0005 with batch size of 1. The number of training iterations is $500,000$, and the Adam [25] optimizer is used, with β_1 and β_2 set to 0.9 and 0.999, respectively. For all compared methods, we use the official training codes or pre-trained models.

To quantitatively evaluate irrelevant attributes preservation, four metrics are used: IDS denotes identity similarity before and after editing calculated by Curricularface [17]. PSNR and SSIM are calculated in the region of intersection of non-hair regions before and after editing. ACD represents the average color difference of the hair region.

4.1. Quantitative and Qualitative Comparison

Comparison to Text-Driven Image Manipulation Methods. We compare our approach with current state-of-the-art text-driven image manipulation methods TediGAN [49] and StyleCLIP [31] on ten text descriptions. The optimization iteration number of TediGAN is set to 200 according to their official recommendations. The visual comparison is shown in Figure 3. TediGAN fails in all hairstyle editing related tasks, only the hair color editing is barely successful but the

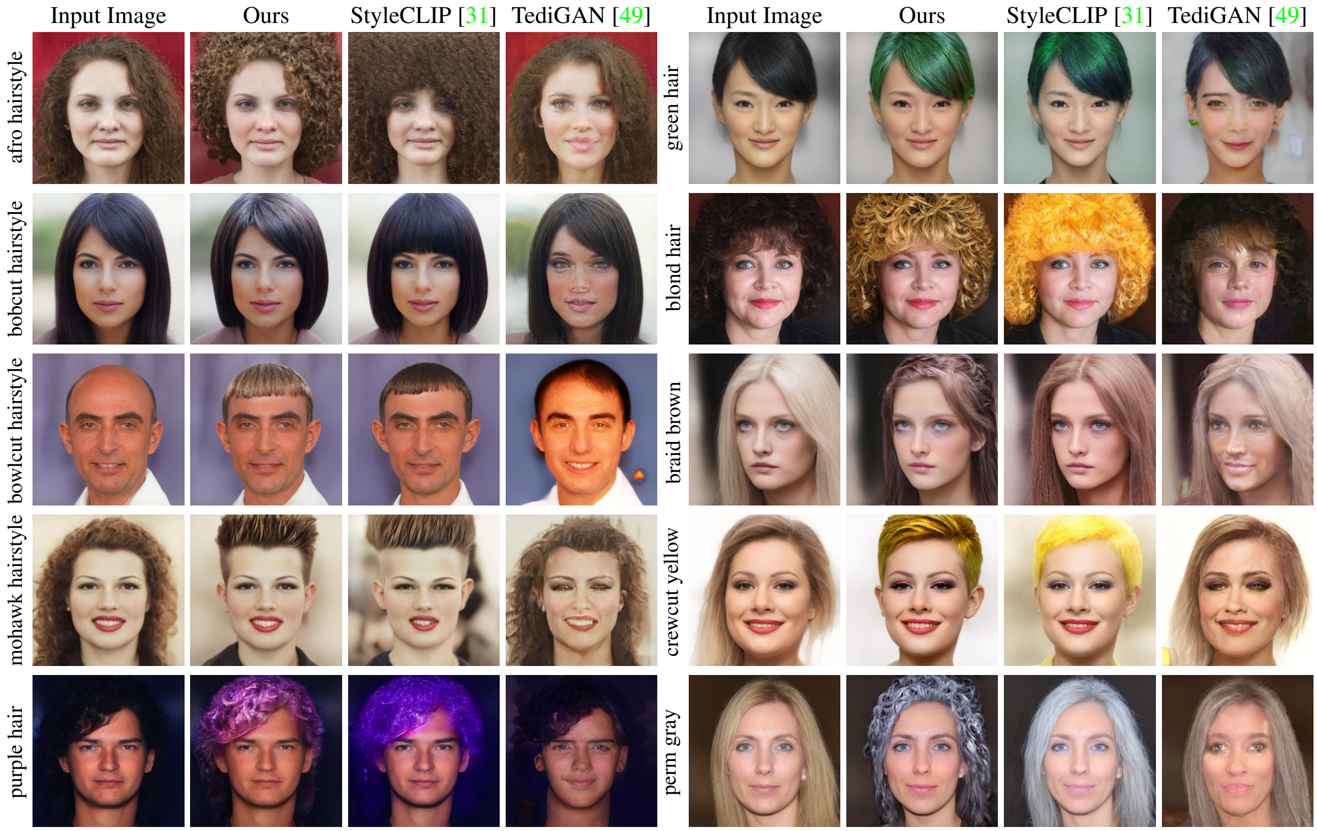

Figure 3. Visual comparison with StyleCLIP [31] and TediGAN [49]. The corresponding simplified text descriptions (editing hairstyle, hair color, or both of them) are listed on the leftmost side of each row, and all input images are the inversions of the real images. Our approach demonstrates better visual photorealism and irrelevant attributes preservation ability while completing the specified hair editing.

results are still unsatisfactory. This phenomenon is consistent with the findings given in the StyleCLIP: the optimization method using CLIP similarity loss is very unstable due to the lack of knowledge learned from a large dataset.

StyleCLIP trains a separate mapper for each description and thus demonstrates stronger manipulation ability on the task of editing only the hairstyle, but excessive manipulation ability instead affects the image realism (see afro hairstyle). Thanks to our *shared condition embedding*, our method finds a balance between the degree of manipulation and realism by fully learning over many hair editing description inputs. On the task of editing both hairstyle and hair color, our method exhibits better manipulation ability. This is due to proposed *disentangled information injection* and *modulation module*, whereas StyleCLIP leaves this information in one description making it poorly decoupled and difficult to perform hairstyle and hair color editing tasks at the same time. In addition, benefiting from attribute preservation loss, our method exhibits better retention of irrelevant attributes (see mohawk hairstyle, purple hair).

In Table 1, we give the average quantitative comparison results in terms of irrelevant attributes preservation on these written text descriptions. And the quantitative results lead to the same conclusions as the visual comparison. We do not compare the FID [14] used in TediGAN here since it can not reflect the manipulation capability. More quantitative results and analysis in terms of the FID metric are given in the supplementary material.

Methods	IDS	PSNR	SSIM
Ours	**0.83**	**27.8**	**0.92**
StyleCLIP [31]	0.79	23.2	0.87
TediGAN [49]	0.17	24.1	0.79

Table 1. Quantitative comparison regarding the preservation of irrelevant attributes. Our approach exhibits the best irrelevant attributes preservation ability.

Comparison to Hair Transfer Methods. Given a hairstyle reference image and a hair color reference image, the purpose of hair transfer is to transfer their corresponding hairstyle and hair color attributes to the input image. We compare our method with the current state-of-the-art LOHO [34] and MichiGAN [40] in Figure 4. Both of these methods perform hairstyle transfer by direct replication in the spatial domain to generate more accurate details of the

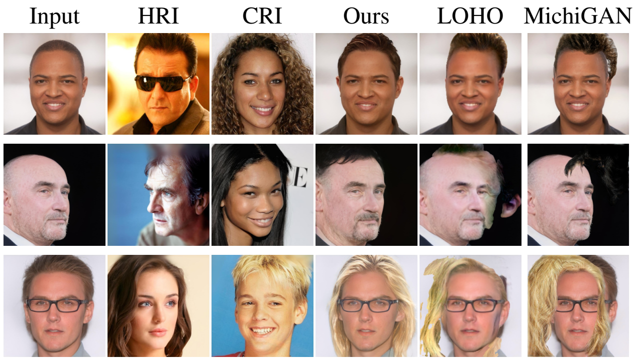

Figure 4. Comparison of our approach with LOHO [34] and MichiGAN [40] on hair transfer. HRI means hairstyle reference image and CRI means hair color reference image.

hair structure, although suffer from obvious artifacts in the boundary areas in some cases (see the results in the first row). However, as shown in the last two rows, they are sensitive to the pose of hairstyle reference images and cannot complete plausible hairstyle transfer when the hairstyle and pose are not well aligned between the hairstyle reference image and the input image. Unlike these two approaches, we transform the measure space of similarity into the latent space of CLIP during training and use the embedding of the hair region of the reference image from CLIP as the conditional input. As a result, our method provides a solution for the unaligned hairstyle transfer and shows its superiority compared to other existing methods.

User Study. To further evaluate the manipulation ability and the visual realism of the edited results of different methods in two types of hair editing tasks, we recruited 20 participants for our user study. For the text-driven image manipulation methods, we provided 20 groups of results from three methods at a time, which were randomly selected from two of each of ten hair editing descriptions. For the hair transfer methods, participants were also provided with 20 groups of results, half of which were aligned hairstyle transfer cases and the other half were non-aligned. Participants were asked to rank three methods for each task with respect to manipulation accuracy and visual realism, where 1 represents the best and 3 represents the worst. The average ranking values are listed in Table 2, where our method outperforms the competitive approaches in both metrics.

4.2. Ablation Analysis

To verify the effectiveness of our proposed network structure and loss functions, we alternately ablate one of these key components to retrain variants of our method, by keeping all but the selected component unchanged.

Importance of Attribute Preservation Loss. To verify the role of each component in the attribute preservation loss, we randomly selected $4,400$ images for qualitative and quantitative ablation studies across the task of editing only hairstyles. Consistent conclusions can be drawn from Table 3 and Figure 5: $\mathcal{L}_{bg}$, $\mathcal{L}_{id}$, and $\mathcal{L}_{norm}$ all contribute to the maintenance of irrelevant attributes, and $\mathcal{L}_{s\_mc}$ helps keep hair color unchanged when only editing the hairstyle.

	Text-Driven Methods			Hair Transfer Methods		
Metrics	Ours	StyleCLIP	TediGAN	Ours	LOHO	MichiGAN
Acc.	**1.39**	1.66	2.95	**1.79**	2.26	1.95
Real.	**1.42**	1.63	2.95	**1.09**	2.48	2.43

Table 2. User study on text-driven image manipulation methods and hair transfer methods. Acc. denotes the manipulation accuracy for given conditional inputs and Real. denotes the visual realism of the manipulated image. The numbers in the table are average rankings, the lower the better.

Methods	IDS	PSNR	SSIM	ACD
Ours	**0.85**	**27.0**	**0.91**	**0.02**
w/o $\mathcal{L}_{bg}$	0.82	19.9	0.82	**0.02**
w/o $\mathcal{L}_{id}$	0.25	22.8	0.80	0.03
w/o $\mathcal{L}_{s\_mc}$	0.82	26.6	0.90	0.09
w/o $\mathcal{L}_{norm}$	0.75	24.9	0.87	0.03

Table 3. Quantitative ablation on attribute preservation loss.

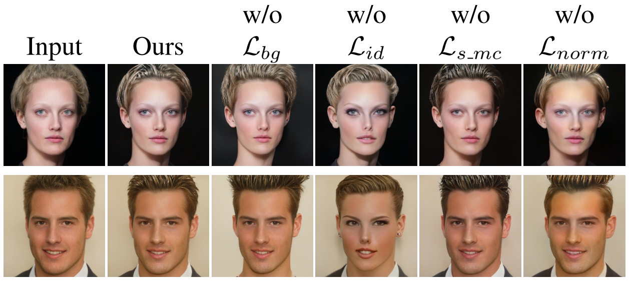

Figure 5. The effect of attribute preservation loss. The text description is "slicked back hairstyle".

Superiority of Network Structure Design. We compare our model with three variants. (a) replace the modulation module with vanilla layernorm layer, and concatenate conditional inputs with the latent code and then feed them into the network. (b) replace the conditional inputs of the coarse and medium sub hair mappers with hair color embedding, and the fine sub hair mapper with hairstyle embedding. (c) replace the conditional input of the medium sub hair mapper with the hair color embedding and leave the rest unchanged. As shown in Figure 6, only our model completes both hairstyle and hair color manipulation. The unsatisfactory result of (a) proves that our *modulation module* can better fuse the condition information into latent space and improve manipulation capability. (b) and (c) confirm the correctness of our *disentangled semantic-matching-based information injection*.

Hair Interpolation. Given two edited latent codes $W_A, W_B \in \mathcal{W}+$, we can achieve fine-grained hair edit-

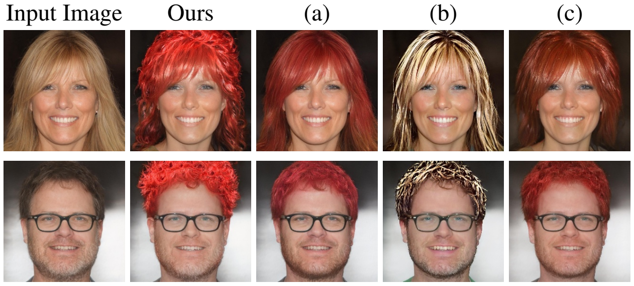

Figure 6. Visual comparison between our method and variants. The text condition is "perm hairstyle and red hair". (a) concatenate conditional inputs with the latent code. (b) replace the conditional inputs of coarse and medium sub hair mappers with hair color embedding, and fine sub hair mapper with hairstyle embedding. (c) replace the conditional input of medium sub hair mapper with the hair color embedding and leave the rest unchanged.

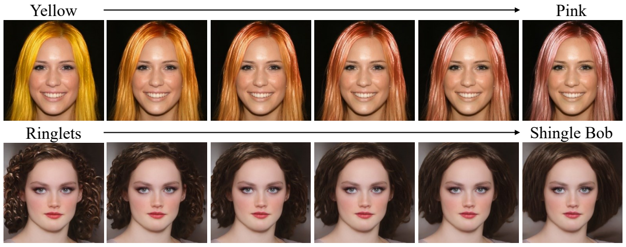

Figure 7. Hair interpolation results. By gradually increasing the blending parameter λ from 0 to 1, we can manage hair editing at a fine-grained level, such as changing from yellow hair to pink hair, from ringlets hairstyle to shingle bob hairstyle.

ing by interpolation. In detail, we combine the two latent codes by linear weighting to generate the intermediate latent code $W_I = \lambda W_B + (1-\lambda)W_A$. Finally, the image corresponding to the intermediate latent code is generated. By gradually increasing the blending parameter λ from 0 to 1, we can manage hair editing at a fine-grained level, as shown in Figure 7.

Generalization Ability. In Figure 8, we demonstrate the generalization ability of our method to unseen text descriptions. Thanks to our strategy of *shared condition embedding*, our method possesses some extrapolation ability after training with only a limited number of hair editing descriptions, which yields reasonable editing results for texts that never appear in the training descriptions.

Cross-Modal Conditional Inputs. Our method supports conditional inputs from the image and text domains individually or jointly, which is not feasible with current existing hair editing methods, and the results are shown in Figure 1. More results will be given in the supplementary materials.

5. Limitations and Negative Impact

Since our editing is done in the latent space of pretrained StyleGAN, we can not complete the editing for some rare hairstyle descriptions or reference images that are not within the domain of StyleGAN. But it can be potentially solved by adding corresponding images to StyleGAN pre-training. For hairstyle transfer, we use the embedding of the reference image in the CLIP latent space as the condition for our hair mapper, which sometimes loses the fine-grained structure and thus cannot achieve a perfect transfer for structural details. The hair-edited images may be used to spread malicious information, which can be evaded by using GAN-generated image detectors [46].

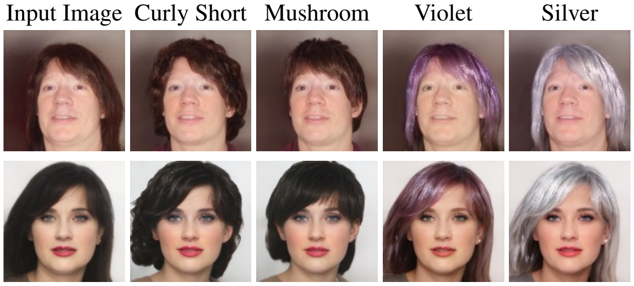

Figure 8. Generalization ability to unseen descriptions. Despite never being trained on these descriptions of "curly short hairstyle", "mushroom hairstyle", "violet hair", and "silver hair", our method can still yield plausible manipulation results.

6. Conclusions

In this paper, we propose a new hair editing interaction mode that unifies conditional inputs from text and image domains in a unified framework. In our framework, users can individually or jointly provide textual descriptions and reference images to complete the hair editing. This multi-modal interaction greatly increases the flexibility of hair editing and reduces the interaction cost for users. By maximizing the great potential of CLIP, tailored network structure designs and loss functions, our framework supports high-quality hair editing in a decoupled manner. Extensive qualitative and quantitative comparisons and user study demonstrate the superiority of our method compared to competing methods in terms of manipulation capability, irrelevant attributes preservation, and image realism.

Acknowledgement This work was supported in part by NSFC under Grant U20B2047, U1636201, 62002334, 62121002, by the Anhui Science Foundation of China under Grant 2008085QF296, by the Exploration Fund Project of USTC under Grant YD3480002001 and the Fundamental Research Funds for the Central Universities under Grant WK2100000011. Jing Liao is supported by an ECS grant from the Research Grants Council of the Hong Kong (Project No. CityU 21209119) and an DON_RMG grant from CityU, Hong Kong (Project No. 9229064). Thank Yi Yin for her help in this work.

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4431–4440, 2019. 3

[2] Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. Only a matter of style: age transformation using a style-based regression model. *ACM Transactions on Graphics (TOG)*, 40:1 – 12, 2021. 2

[3] Abdul Fatir Ansari, J. Scarlett, and Harold Soh. A characteristic function approach to deep implicit generative modeling. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7476–7484, 2020. 2

[4] Martín Arjovsky, Soumith Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *ICML*, 2017. 2

[5] Menglei Chai, Linjie Luo, Kalyan Sunkavalli, Nathan A. Carr, Sunil Hadap, and Kun Zhou. High-quality hair modeling from a single portrait photo. *ACM Transactions on Graphics (TOG)*, 34:1 – 10, 2015. 2

[6] Menglei Chai, Tianjia Shao, Hongzhi Wu, Yanlin Weng, and Kun Zhou. Autohair: fully automatic hair modeling from a single image. *ACM Trans. Graph.*, 35:116:1–116:12, 2016. 2

[7] Jianbo Chen, Yelong Shen, Jianfeng Gao, Jingjing Liu, and Xiaodong Liu. Language-based image editing with recurrent attentive models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8721–8729, 2018. 3

[8] Edo Collins, R. Bala, B. Price, and S. Süsstrunk. Editing in style: Uncovering the local semantics of gans. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5770–5779, 2020. 2, 3

[9] Jiankang Deng, J. Guo, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4685–4694, 2019. 5

[10] Lore Goetschalckx, Alex Andonian, A. Oliva, and Phillip Isola. Ganalyze: Toward visual definitions of cognitive image properties. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5743–5752, 2019. 2, 3

[11] Ian J. Goodfellow, Jean Pouget-Abadie, M. Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014. 2

[12] Ishaan Gulrajani, F. Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of wasserstein gans. In *NIPS*, 2017. 2

[13] Tianyu Guo, C. Xu, Jiajun Huang, Yunhe Wang, Boxin Shi, Chao Xu, and Dacheng Tao. On positive-unlabeled classification in gan. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8382–8390, 2020. 2

[14] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, 2017. 6

[15] Liwen Hu, Chongyang Ma, Linjie Luo, and Hao Li. Robust hair capture using simulated examples. *ACM Transactions on Graphics (TOG)*, 33:1 – 10, 2014. 2

[16] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017. 4

[17] Y. Huang, Yuhan Wang, Ying Tai, Xiaoming Liu, Pengcheng Shen, Shao xin Li, Jilin Li, and Feiyue Huang. Curricularface: Adaptive curriculum learning loss for deep face recognition. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5900–5909, 2020. 5

[18] A. Jahanian, L. Chai, and Phillip Isola. On the ”steerability” of generative adversarial networks. *ArXiv*, abs/1907.07171, 2020. 2, 3

[19] Yuming Jiang, Ziqi Huang, Xingang Pan, Chen Change Loy, and Ziwei Liu. Talk-to-edit: Fine-grained facial editing via dialog. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13799–13808, 2021. 3

[20] Youngjoo Jo and Jongyoul Park. Sc-fegan: Face editing generative adversarial network with user's sketch and color. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1745–1753, 2019. 1

[21] Youngjoo Jo and Jongyoul Park. Sc-fegan: Face editing generative adversarial network with user's sketch and color. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1745–1753, 2019. 2

[22] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018. 5

[23] Tero Karras, S. Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4396–4405, 2019. 2, 3, 5

[24] Tero Karras, S. Laine, Miika Aittala, Janne Hellsten, J. Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8107–8116, 2020. 2, 3, 5

[25] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 5

[26] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5548–5557, 2020. 1, 2

[27] Ziwei Liu. https://github.com/switchablenorms/CelebAMask-HQ/tree/master/face_parsing. Accessed: Nov. 2021. [Online]. 5

[28] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pages 13–23, 2019. 1, 2

[29] Yotam Nitzan, Amit H. Bermano, Yangyan Li, and Daniel Cohen-Or. Face identity disentanglement via latent space

mapping. *ACM Transactions on Graphics (TOG)*, 39:1 – 14, 2020. 2

[30] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2337–2346, 2019. 4

[31] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2085–2094, 2021. 1, 2, 3, 5, 6

[32] 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 *ICML*, 2021. 1, 2, 3

[33] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2287–2296, 2021. 3

[34] Rohit Saha, Brendan Duke, Florian Shkurti, Graham W. Taylor, and Parham Aarabi. Loho: Latent optimization of hairstyles via orthogonalization. In *CVPR*, 2021. 2, 6, 7

[35] Edgar Schönfeld, B. Schiele, and A. Khoreva. A u-net based discriminator for generative adversarial networks. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8204–8213, 2020. 2

[36] Yujun Shen, Jinjin Gu, X. Tang, and B. Zhou. Interpreting the latent space of gans for semantic face editing. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9240–9249, 2020. 2, 3

[37] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. In *International Conference on Learning Representations*, 2019. 1, 2

[38] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111, 2019. 1, 2

[39] Zhentao Tan, Menglei Chai, Dongdong Chen, Jing Liao, Qi Chu, Bin Liu, Gang Hua, and Nenghai Yu. Diverse semantic image synthesis via probability distribution modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 2

[40] Zhentao Tan, Menglei Chai, Dongdong Chen, Jing Liao, Qi Chu, Lu Yuan, Sergey Tulyakov, and Nenghai Yu. Michigan: multi-input-conditioned hair image generation for portrait editing. *ACM Transactions on Graphics (TOG)*, 39(4):95–1, 2020. 1, 2, 6, 7

[41] Zhentao Tan, Dongdong Chen, Qi Chu, Menglei Chai, Jing Liao, Mingming He, Lu Yuan, Gang Hua, and Nenghai Yu. Efficient semantic image synthesis via class-adaptive normalization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 4

[42] Song Tao and J. Wang. Alleviation of gradient exploding in gans: Fake can be real. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1188–1197, 2020. 2

[43] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *ACM Transactions on Graphics (TOG)*, 40:1 – 14, 2021. 3, 5

[44] Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. *arXiv preprint arXiv:2112.05139*, 2021. 3

[45] Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Cross-domain and disentangled face manipulation with 3d guidance. *arXiv preprint arXiv:2104.11228*, 2021. 2

[46] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. Cnn-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8695–8704, 2020. 8

[47] Lingyu Wei, Liwen Hu, Vladimir G. Kim, Ersin Yumer, and Hao Li. Real-time hair rendering using sequential adversarial networks. In *ECCV*, 2018. 2

[48] Tianyi Wei, Dongdong Chen, Wenbo Zhou, Jing Liao, Weiming Zhang, Lu Yuan, Gang Hua, and Nenghai Yu. A simple baseline for stylegan inversion. *ArXiv*, abs/2104.07661, 2021. 2, 3

[49] Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. Tedigan: Text-guided diverse face image generation and manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2256–2265, 2021. 3, 5, 6

[50] Chufeng Xiao, Deng Yu, Xiaoguang Han, Youyi Zheng, and Hongbo Fu. Sketchhairsalon: Deep sketch-based hair image synthesis. *arXiv preprint arXiv:2109.07874*, 2021. 1, 2

[51] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021. 2

[52] Peihao Zhu, Rameen Abdal, Yipeng Qin, and Peter Wonka. Sean: Image synthesis with semantic region-adaptive normalization. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5103–5112, 2020. 2