Supervised Attribute Information Removal and Reconstruction for Image Manipulation

Nannan Li and Bryan A. Plummer

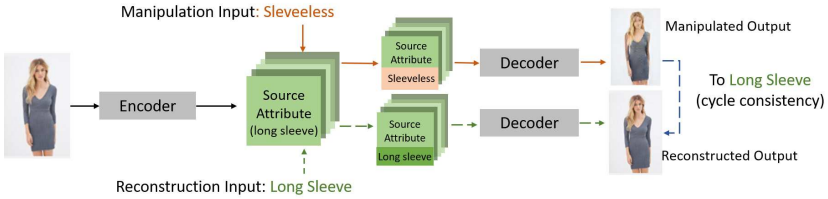
Boston University
{nnli,bplum}@bu.edu

Abstract. The goal of attribute manipulation is to control specified attribute(s) in given images. Prior work approaches this problem by learning disentangled representations for each attribute that enables it to manipulate the encoded source attributes to the target attributes. However, encoded attributes are often correlated with relevant image content. Thus, the source attribute information can often be hidden in the disentangled features, leading to unwanted image editing effects. In this paper, we propose an Attribute Information Removal and Reconstruction (AIRR) network that prevents such information hiding by learning how to remove the attribute information entirely, creating attribute excluded features, and then learns to directly inject the desired attributes in a reconstructed image. We evaluate our approach on four diverse datasets with a variety of attributes including DeepFashion Synthesis, DeepFashion Fine-grained Attribute, CelebA and CelebA-HQ, where our model improves attribute manipulation accuracy and top-k retrieval rate by 10% on average over prior work. A user study also reports that AIRR manipulated images are preferred over prior work in up to 76% of cases[1].

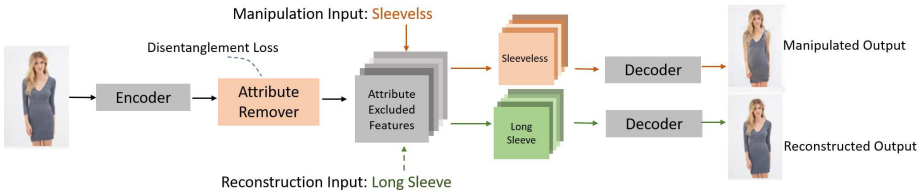
1 Introduction

Attribute manipulation translates images based on desired attributes, which has applications to face editing [26,32,31], image retrieval [27,12,34], and image synthesis [3,16], among others. In these tasks, the goal is to be able to control a specified attribute without affecting other information in the source image. While Generative Adversarial Networks (GANs) have achieved impressive performance on attribute manipulation, a major challenge is that the generator tends to take a shortcut by utilizing the preserved source attribute information instead of the target attribute for manipulation [13,29,18], thus causing improper image editing effects in manipulated images. Prior work has tried to address this by adding random noise during the reconstruction [4,30] or learning disentangled attribute representations which are used to manipulate the images [26,33]. However, low-magnitude random noise is not targeted to the source attribute, which could be intentionally ignored by the model or inadvertently suppress key source features. On the other hand, even with a disentangled image representation, the correlation between attributes and relevant image content could cause source attribute

[1] Code and models are available at https://github.com/NannanLi999/AIRR



(a) Framework of previous methods [3,12,34,35,11]. Dashed arrows mean two different ways of obtaining the reconstructed image



(b) Pipeline of the proposed method

Fig. 1: In (a), the generator like those used by [11,3,12,34,35] incorrectly utilizes the hidden source attribute information instead of the target attribute *sleeveless* for image manipulation. As a result, the manipulated image still contains the source attribute *long sleeve*, causing improper image editing effects. To avoid this issue, the proposed method in (b) erases the source attribute information in the encoded features through an attribute remover with a disentanglement loss, conditioning the manipulated output only on the input target attribute *sleeveless* and the attribute excluded features

information to be hidden in the rest of the image features. For example, attribute *formal* in a dress is often correlated with the dress's *long length*.

To address these issues, we propose a supervised Attribute Information Removal and Reconstruction (AIRR) model that learns an attribute excluded representation and reconstructs the image with desired attributes. The key challenge is in identifying the preserved source attribute information and decorrelating it from the image representation [15,31,26,33]. Prior research on feature disentanglement either doesn't consider the decorrelation [15,31,35], or has limitations on the number of attributes it can decorrelate in a forward pass [26,33], unlike our approach which can disentangle any number of attributes. In addition, as illustrated in Figure 1a, these methods often rely on the full image information for both manipulation and reconstruction. As mentioned earlier, this can lead to information hiding in the manipulated image. In contrast, as shown in Figure 1b, we use our remover to erase attribute information to obtain attribute excluded features, which are then used to directly generate both the reconstructed and manipulated images. Since this should eliminate the information that could potentially be hidden in the disentangled representation, we avoid the information hiding issues in prior work.

One challenge in our approach is our reliance on being able to identify and remove attribute information in real images. For example, although the color *white* appears in the background of the input images in Figure 1, a good attribute classifier would predict that the clothing item is *gray* and not *white*. This means that the background information could mislead the attribute recognition. To address this issue, we segment the object of interest (*e.g.*, using [19,36]) to split the image encoder into two branches for the object of interest and the background, respectively. This helps AIRR to concentrate the manipulation on the object of interest without influencing the background information.

Our main contributions are:

- We propose the Attribute Information Removal and Reconstruction method (AIRR), a controllable disentangled attribute manipulation framework that produces high quality images. The key insight in AIRR is the attribute information removal and reconstruction module that produces an attribute excluded representation, eliminating sources of information hiding that degrades performance in prior work.
- Extensive experiments across DeepFashion Synthesis [21], DeepFashion Fine-grained Attribute [21], CelebA [22] and CelebA-HQ [17] report that AIRR improves the attribute manipulation accuracy and top-k retrieval rate by 10% on average over the state-of-the-art. Moreover, we show that AIRR can effectively control attribute strength as well as efficiently manipulating multiple attributes in a single forward pass.
- A user study further validates the effectiveness of our approach, where our methods are shown to produce high quality images that more accurately achieve the target attribute manipulation by up to 76% over prior work.

2 Related Work

Early research in attribute manipulation [7,11] combined the target attribute label directly with the image or image features, and decoded them into manipulated output. However, the decoder could incorrectly use the preserved source attribute information for image manipulation. Thus, more recent work (including this paper), has focused on learning disentangled attribute representations, which we will discuss in more detailed below.

Unsupervised disentanglement. Several studies explored disentanglement in the latent space of GANs in an unsupervised manner [25,23,9,28,32]. These methods aim to manipulate the attributes on synthetic data, where the image content is randomly generated. In [9], the authors found that the principle components of features on pretrained GANs represent high-level semantic concepts. In [32], the authors introduced channel-wise disentanglement of StyleGAN [14]. Shoshan *et al.* [28] utilized contrastive learning to disentangle the latent space, achieving explicit control over synthetic facial images. However, without manual examinations on the feature space, it's difficult to locate the exact attribute representation that we want to manipulate, especially for attributes with high-

level semantics. Thus, in our work we focus on cases where attributes we wish to manipulate are known, enabling us to directly target our feature learning.

Supervised disentanglement. Supervised disentanglement methods edit real images based on attribute annotations. Prior work on this task can be categorized in two types: spatial disentanglement and feature disentanglement. In methods that focus on spatial disentanglement [15,31], attributes are located spatially and thus disentangled in the feature map. These methods can find attribute-specific features by an attention map, whereas attribute-relevant information is implicitly kept and thus influences the image manipulation. On the other hand, feature disentanglement identifies certain features corresponding to the manipulated attribute. [35] presents a method that learns a linear transformation function that maps StyleGAN's latent code. Although StyleGAN's latent space is disentangled [1], without orthogonal constraints, such linear combination could result in correlation between different image attributes and content. To address this, Yang *et al*. [33] learned attribute relevant and irrelevant features, but each manipulated attribute requires training its own model, which is computationally costly. Instead, Shen *et al*. [26] manipulated attributes with a conditional subspace projection via Support Vector Machines (SVM), whereas the manipulation accuracy depends on the capability of SVM and each forward pass can control only a single attribute. In contrast, our proposed approach can manipulate multiple source attributes in a single forward pass by utilizing the injected attribute embedding.

Attribute manipulation in fashion. Apart from the above mentioned methods that are mainly applied to facial attribute editing, attribute manipulation in fashion images has also gained a lot of attention. Recent work on this topic mainly aim to improve the image retrieval accuracy for item recommendation. For example, researches have leveraged spatial information when manipulating attributes [3,2], learned a dictionary of attribute transformations [27], or used the attribute probability distribution as an disentangled representation for image retrieval[12]. Kwon *et al*. [16] predicted changes to an item's shape as a result of changing an attribute, enabling them to make more significant alterations to the clothing in images. However, many of these methods also suffered from issues with disentangling attributes, often due to misinformation hiding, which our work minimizes.

3 Attribute Information Removal and Reconstruction

Given image I and its attributes $\mathbf{A} = \{\mathbf{a_1}, \mathbf{a_2}, ..., \mathbf{a_n}\}$, where a_i denotes the ith attribute, we aim to manipulate any number of attributes in $\mathbf{A}$. To achieve this goal, the generator first takes a real image as input, and uses our attribute remover to decorrelate the image attributes from the image features. The resulting attribute excluded representation is then combined with the target attribute embeddings to produce the manipulated output I_{map}. In the following, we introduce the four components of our model: image encoder (Section 3.1), attribute

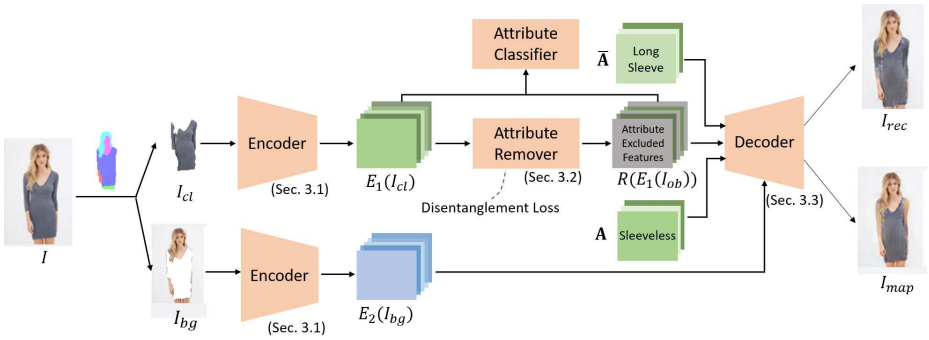


Fig. 2: **AIRR framework**. In the AIRR generator, a given image is parsed into an object of interest I_{cl} and background I_{bg} through an offline parser [19,36]. I_{cl} and I_{bg} are encoded in separate branches (Sec. 3.1). In the I_{cl} branch, the source attribute information in the encoded features are erased by an attribute remover using a disentanglement loss (Sec. 3.2). Subsequently, the source attributes in $\mathbf{A}$ and the target attributes in $\bar{\mathbf{A}}$ are embedded into the attribute excluded features for image reconstruction and image manipulation, respectively (Sec. 3.3)

remover (Section 3.2), decoder (Section 3.3), and learning objectives (Section 3.4). Figure 2 shows an overview of our approach.

3.1 Image Encoder

To concentrate the manipulation on the object that we want to manipulate, the image encoder in AIRR is split into two branches for the object of interest I_{cl} and the background I_{bg}, respectively. Prior work often achieves the segmentation of I_{cl} by learning an attention map [15,3,2], while we found empirically that using an offline parser [19,36] is more accurate in segmenting instances. This segmentation is especially helpful for images with multiple objects, *e.g.*, an image of a fashion model wearing top, leggings and boots. As shown in Figure 2, after obtaining I_{cl} and I_{bg}, the image encoder encodes I_{cl} into $E_1(I_{cl})$ in the first branch, and I_{bg} into $E_2(I_{bg})$ in the second branch. Later on, AIRR only manipulates $E_1(I_{cl})$ without influencing the background information $E_2(I_{bg})$.

3.2 Attribute Remover

While prior work directly used the image features $E_1(I_{cl})$ to generate the image [11,3,12,34,35], in AIRR, image features $E_1(I_{cl})$ from our base encoder are fed into an attribute remover to learn an attribute-excluded representation $R(E_1(I_{cl}))$. The attribute remover is an n-layer convolutional block that is used to decorrelate the source attribute information from the image representation. A design requirement for the attribute remover is that it does not have skip connections since we aim to erase the attribute information from the encoded features, whereas skip connections would preserve this information.

To disentangle all the source attribute information from $E_1(I_{cl})$, we would need an attribute classifier to first identify these attributes. This can be achieved by Maximum Likelihood Estimation (MLE):

$$L_d\big(E_1(I_{cl})\big) = -\sum\nolimits_{\mathbf{a_i}\in\mathbf{A}} \mathbf{y_i}^T \log p_c(\mathbf{a_i}|E_1(I_{cl})) \tag{1}$$

where $p_c(\mathbf{a_i}|E_1(I_{cl}))$ is the probability distribution of $\mathbf{a_i}$, and $\mathbf{y_i}$ is the corresponding one-hot attribute label. We use one residual block as the attribute classifier to predict $p_c(\cdot)$.

After identifying the source attributes, we can then eliminate the attribute information in $R(E_1(I_{cl}))$ by minimizing their mutual information. Alternatively, it's easier to minimize the upper bound of this mutual information, which is the maximum log probability in the attribute class distribution added by a constant c (See the Supplementary for proof of this upper bound):

$$\mathrm{MI}\big(\mathbf{a}_i, R(E_1(I_{cl}))\big) \leq c + \max\nolimits_{\mathbf{a}_i} \log p_c\big(\mathbf{a}_i|R(E_1(I_{cl}))\big) \tag{2}$$

Intuitively, minimizing this upper bound gives a uniform distribution over the attributes, meaning that the uncertainty for a specific attribute is maximized. Therefore, the generated features would have little knowledge of what the original attributes are. This loss function is thus defined as a margin loss:

$$L_d\big(R(E_1(I_{cl}))\big) = \sum_{\mathbf{a_i}\in\mathbf{A}} \max\{\max\nolimits_{\mathbf{a}_i} \log p_c\big(\mathbf{a_i}|R(E_1(I_{cl}))\big) - \log\frac{1}{|\mathbf{a_i}|}, c\}, \tag{3}$$

where $|\mathbf{a_i}|$ is the number of attribute values in $\mathbf{a_i}$, and c indicates the proximity to a uniform distribution, which is set to 0.01 in our experiments as we found it performed well. We refer to $L_d(R(E_1(I_{cl})))$ as a Mutual Information Minimization (MIM) loss. Note that our attribute classifier and remover are trained end-to-end with our other generator and decoder components.

3.3 Decoder

After the attribute remover, a new learned attribute representation is integrated with the disentangled features $R(E_1(I_{cl}))$ to generate the output image. Assuming a scale embedding vector $\boldsymbol{\beta}_{\boldsymbol{a_i}}$ and a bias embedding vector $\boldsymbol{\gamma}_{\boldsymbol{a_i}}$ for attribute $\mathbf{a_i}$, the original image I thus can be reconstructed by combining $R\big(E_1(I_{cl})\big)$, $E_2(I_{bg})$ and the attribute embeddings, *i.e.*,

$$I_{rec} = G\Big(\mathrm{concat}\big[\sum\nolimits_{\mathbf{a_i}\in\mathbf{A}} \boldsymbol{\beta}_{\mathbf{a_i}} \cdot R\big(E_1(I_{cl})\big) + \boldsymbol{\gamma}_{\mathbf{a_i}}, E_2(I_{bg})\big]\Big), \tag{4}$$

where G is the decoder. Similarly, given target attributes of our desired output $\bar{\mathbf{A}}$, $R\big(E_1(I_{cl})\big)$ produces the manipulated image I_{map} by

$$I_{map} = G\Big(\mathrm{concat}\big[\sum\nolimits_{\bar{\mathbf{a}}_\mathbf{i}\in\bar{\mathbf{A}}} \boldsymbol{\beta}_{\bar{\mathbf{a}}_\mathbf{i}} \cdot R\big(E_1(I_{cl})\big) + \boldsymbol{\gamma}_{\bar{\mathbf{a}}_\mathbf{i}}, E_2(I_{bg})\big]\Big) \tag{5}$$

In contrast to some prior work [3,7,15], where the attribute embedding vector and the encoded features are combined by concatenation, we multiply these two features such that linear interpolation between different attribute embeddings can better control the strength of these attributes in the output [8,33]. Further discussion can be found in Section 4.5.

3.4 Learning Objectives

Disentanglement loss. The disentanglement loss combines the MLE loss in Eq. (1) and the MIM loss in Eq. (3) as

$$L_d = L_d\big(E_1(I_{cl})\big) + L_d\big(R\big(E_1(I_{cl})\big)\big) \tag{6}$$

By first identifying the source attributes in $E_1(I_{cl})$ and then minimizing the attribute information in $R(E_1(I_{cl}))$, this disentanglement loss enables the decoder to condition the output on the new attribute that is injected to the generator. In Section 4.5, we also show empirically that with the disentanglement loss, the attribute remover indeed gets rid of all the source attribute information.

Reconstruction loss. The reconstructed image I_{rec} is evaluated by its l_1 distance to the original image I:

$$L_{rec} = ||I_{rec} - I||_1 \tag{7}$$

Adversarial loss. The manipulated image I_{map} doesn't have a paired ground truth of how it should look like, for which its plausibility is evaluated by the discriminator D. Using LSGAN [24], the adversarial loss of the generator can be written as

$$L^g_{adv} = (1 - D(I_{map}))^2 + (1 - D(I_{rec}))^2 \tag{8}$$

In the discriminator, this adversarial loss includes both the reconstructed image I_{rec} and the manipulated image I_{map} since they are both fake samples:

$$L^d_{adv} = (1 - D(I))^2 + \frac{1}{2}\big((D(I_{map})^2 + D(I_{rec})^2\big) \tag{9}$$

Image attribute classification loss [6]. This loss maximizes the mutual information between the injected attribute embedding and the generated image:

$$L^g_{attr} = -\sum\nolimits_{\mathbf{a_i} \in \mathbf{A}} \mathbf{y_i}^T \log p_d(\mathbf{a_i}|I_{rec}) - \sum\nolimits_{\bar{\mathbf{a}}_\mathbf{i} \in \bar{\mathbf{A}}} \bar{\mathbf{y}}_\mathbf{i}^T \log p_d(\bar{\mathbf{a}}_\mathbf{i}|I_{map}) \tag{10}$$

where $p_d(\cdot)$ is the probability distribution of attributes predicted by a classification branch in the discriminator. For the discriminator, this loss is defined on the real image I.

$$L^d_{attr} = L^d_{attr}(I) = -\sum\nolimits_{\mathbf{a_i} \in \mathbf{A}} \mathbf{y_i}^T \log p_d(\mathbf{a_i}|I) \tag{11}$$

Perceptual loss. To further improve the quality of the generated images, a perceptual loss is introduced in the generator as in [3]. It is based on the distance of paired real and fake images in the CNN feature space

$$L_p = ||\mathrm{CNN}(I) - \mathrm{CNN}(I_{rec})||_1 + ||\mathrm{CNN}(I_{ref}) - \mathrm{CNN}(I_{map})||_1 \tag{12}$$

Table 1: Statistics of the datasets used in our experiments in Section 4

Dataset	image size	#training images	#test images	#attributes	#attribute values
DeepFashion Synthesis [21]	128x128	76,979	2000	2	21
DeepFashion Fine-grained Attribute [21]	256x256	19,000	1000	6	26
CelebA [22]	128x128	200,599	2000	8	21
CelebA-HQ [17]	1024x1024	29,000	1000	8	21

where I_{ref} is selected from the real images in the dataset to have exactly the same attributes $\bar{\mathbf{A}}$ as I_{map}.

Full objective. Including all the above loss functions, the full objectives for the generator and discriminator are

$$L_{gen} = L^g_{adv} + \lambda_1 L_d + \lambda_2 L^g_{attr} + \lambda_3 L_{rec} + \lambda_4 L_p \tag{13}$$

$$L_{dis} = L^d_{adv} + 2\lambda_2 L^d_{attr} \tag{14}$$

where $\lambda_1, \lambda_2, \lambda_3$ are trade-off parameters. As in prior work [20,5], these parameters must be set carefully to control the degree of disentanglement. Note that except for L_{rec}, these loss functions are all symmetrical with respect to $\mathbf{A}$ and $\bar{\mathbf{A}}$ to enforce that the manipulated image is a plausible reconstructed image.

4 Experiments

To prove the efficiency of the proposed method, we evaluate our model on four publicly available datasets: DeepFashion Synthesis [21], DeepFashion Fine-grained Attribute [21], CelebA [22] and CelebA-HQ [17]. On CelebA and CelebA-HQ, we group the attributes into 8 attribute categories and 21 attribute values following [2]. See Table 1 for detailed statics of each dataset.

4.1 Implementation Details

We use the model architecture in [37] as our backbone on DeepFashion and CelebA datasets. Following [35], on CelebA-HQ we adopt another backbone: StyleGAN2 [14], which is better suited to high resolution images. To improve training stability, we froze the weights of StyleGAN2's encoder and generator. Except for CelebA-HQ, the CNN used in Eq. (12) is a ResNet-50 model [10] pretrained on image attribute classification task. On CelebA-HQ, we use StyleGAN2's encoder as the CNN in order to match the identity loss defined in [35].

For DeepFashion the target attributes are uniformly and randomly sampled from items in the same clothing category, *e.g.*, dress and leggings. Here we did not sample the target attributes from the whole dataset because some annotated attributes can only appear in certain clothing categories. For example, leggings can't have V-neckline, and skirts can't have long sleeve. On CelebA-HQ, we traverse the values of the 8 attributes for each image during test for fair comparison with existing methods that use binary attribute values [33,35,26].

4.2 Experimental Settings

Baselines. We compare our model with related approaches on attribute manipulation: StarGAN [7], AMNet [3], FLAM [27], VPTNet [16], AttGAN [11], Student [18], FSNet-v2 [2], CAFE-GAN [15], InterfaceGAN [26], L2M-GAN [33] and LatentTransformer [35]. Among these approaches, FLAM, Student, InterfaceGAN and L2M-GAN also aim to achieve feature-level disentanglement, while AMNet, FSNet-v2 and CAFE-GAN aim to learn spatially disentangled representations. For StarGAN, AttGAN, Student, InterfaceGAN, L2M-GAN and LatentTransformer, we used the official implementations at author-provided links. AMNet and FLAM are reproduced by us following the configurations provided in the corresponding papers. Results of FashionSearchNet-v2, VPTNet and CAFE-GAN are directly copied from the original papers. For fair comparison on CelebA-HQ, we used StyleGAN2 encoded image features in InterfaceGAN.

Evaluation metrics. Following [3,2], we use human evaluation and two standard metrics to evaluate the model's performance on attribute manipulation: attribute manipulation accuracy and top-k retrieval. Attribute manipulation accuracy, which is the classification accuracy of the target attribute on the manipulated images, measures the extent to which a model can modify the target attribute. We use a ResNet-50 model [10] pretrained on attribute classification to evaluate the attribute manipulation accuracy on DeepFashion and CelebA. On CelebA-HQ, the accuracy is computed using the same facial attribute classifier as [35] for a fair comparison. Top-k retrieval, on the other hand, evaluates both the attribute changing and preservation capability. It is defined as the number of hits divided by the total number of queries. A query is called a hit if any of the manipulated image's top-k matches has exactly the target attributes in $\bar{\mathbf{A}}$. The top-k retrieval rate is averaged across all attributes. In all experiments, we use the deep features in the last fully-connected layer of the attribute classifier for image retrieval. All retrieval galleries have 20,000 images.

4.3 Quantitative Results

As shown in Table 2-5, AIRR outperforms the state-of-the-art by a significant margin on most evaluation metrics. For example, Table 5 shows the average attribute manipulation accuracy and top-k retrieval rates on CelebA-HQ, which boost performance by more than 20% compared to existing methods. The improvements are more obvious on additive attributes, such as *wearing hat*, which reports gains over prior work by more than 40% in Table 4 and 5. Further discussion on ablations of our model can be found in Section 4.5.

Attribute preservation analysis. To analyze what influence that changing a specific attribute has on preserving others, we gradually increase the ratio of manipulated images (*i.e.*, the number of manipulated images divided by the number of all test images), and observe the ratio of successfully preserved attributes. Figure 3 provides the attribute changing rate (*i.e.*, attribute manipulation accuracy) vs. attribute preservation rate for each attribute in the DeepFashion dataset. In each graph, the preservation rates are averaged over all attributes excluding the

Table 2: Results on DeepFahison Synthesis. In our models, $\lambda_1 = 0.25, \lambda_2 = 0.125, \lambda_3 = 1.0, \lambda_4 = 1.0$

Method	Manipulation Accuracy			Top-K Retrieval	
	Color	Sleeve	Avg.	R@5	R@20
StarGAN [7]	70.4	77.2	73.8	71.1	82.9
AMNet [3]	74.4	82.1	78.3	85.1	90.5
AttGAN [11]	80.2	91.0	85.6	90.6	95.2
FLAM [27]	-	-	-	26.7	41.3
VPTNet [16]	-	85.7	-	-	-
AIRR (w/o mask)	88.8	92.2	90.5	94.4	97.1
AIRR (w/o L_d)	93.9	89.5	91.7	95.0	97.3
AIRR (w L_h)	89.4	90.5	90.0	90.8	92.4
AIRR	**94.1**	**96.5**	**95.3**	**97.6**	**98.8**

Table 3: Results on DeepFahison Fine-grained Attributes. In our models, $\lambda_1 = 0.05, \lambda_2 = 0.125, \lambda_3 = 2.0, \lambda_4 = 1.0$

Method	Manipulation Accuracy							Top-K Retrieval	
	Pattern	Sleeve	Length	Neckline	Material	Style	Avg.	R@5	R@20
StarGAN [7]	54.0	38.7	22.4	44.3	47.2	24.6	38.5	25.1	39.3
AttGAN [11]	47.4	31.5	19.0	33.7	40.3	23.5	32.6	28.3	39.1
AMNet [3]	53.6	56.5	24.4	68.3	47.9	27.4	46.4	44.1	47.5
FLAM [27]	-	-	-	-	-	-	-	17.6	29.8
AIRR (w/o mask)	70.7	44.3	19.0	57.3	55.0	25.9	45.4	38.2	52.4
AIRR (w/o L_d)	**89.1**	60.6	31.8	73.5	74.9	34.2	60.8	56.4	68.7
AIRR (w L_h)	85.1	59.2	25.7	72.4	72.6	34.7	58.2	51.1	60.2
AIRR	87.8	**65.9**	**32.2**	**74.5**	**76.3**	**35.8**	**62.1**	**57.7**	**70.3**

Table 4: Results on CelebA. In our models, $\lambda_1 = 0.5, \lambda_2 = 0.5, \lambda_3 = 1.0, \lambda_4 = 1.0$

Method	Manipulation Accuracy									Top-K Retrieval	
	Hair Color	Beard	Hair Type	Smiling	Eyeglasses	Gender	Hat	Age	Avg.	R@5	R@20
StarGAN [7]	55.2	51.6	35.6	64.0	86.1	36.5	6.4	44.9	38.8	39.9	55.2
Student [18]	47.7	43.5	37.3	60.0	12.1	42.7	11.5	39.8	36.8	38.2	53.9
AMNet [3]	58.6	34.4	26.7	43.7	10.0	21.0	13.0	22.4	28.7	33.1	46.0
AttGAN [11]	72.6	88.5	48.1	79.8	94.7	89.7	21.7	60.6	69.5	72.4	86.5
CAFE-GAN [15]	83.6	40.1	-	-	-	**95.2**	-	**88.6**	-	-	-
FSNet-v2 [2]	-	-	-	-	-	-	-	-	-	68.0	77.5
AIRR(w/o mask)	**86.1**	96.0	**58.5**	92.7	98.9	91.3	75.6	64.9	83.0	86.5	94.3
AIRR (w/o L_d)	74.3	93.9	51.4	92.4	99.4	89.5	83.4	69.7	81.8	87.1	94.9
AIRR	75.9	**96.4**	58.4	**94.8**	**99.1**	91.6	**93.1**	80.6	**86.2**	**89.1**	**95.6**

target attribute. In Figure 3, our method achieves the highest preservation rate under the same attribute changing rate, proving its capability of controllable attribute manipulation as well as preservation.

User study. We also conducted human evaluation experiments on DeepFashion Fine-grained Attribute and CelebA-HQ using Amazon Mechanical Turk service to verify the quality of manipulated images. We tested on 50 images in each dataset, and different 5 worker were assigned per image. Each worker was presented 3 pictures: the original image, the manipulated image produced by AIRR, and the manipulated image generated by a randomly chosen baseline approaches in Table 6 or 7. The worker was asked to pick an image that better converts the specified attribute in the given image. Table 6 shows that 54-65% of workers

Table 5: Results on CelebA-HQ. In our models, $\lambda_1 = 0.25, \lambda_2 = 0.125, \lambda_3 = 20.0, \lambda_4 = 10.0$

Method	Manipulation Accuracy									Top-K Retrieval	
	Hair Color	Beard	Hair Type	Smiling	Eyeglasses	Gender	Hat	Age	Avg.	R@5	R@20
InterfaceGAN [26]	38.4	**80.8**	36.4	**97.7**	29.7	55.8	2.9	42.0	48.0	19.1	38.8
LatentTrans [35]	37.0	78.8	48.9	85.3	49.6	62.2	5.6	47.4	51.9	20.7	41.8
L2M-GAN [33]	-	-	-	89.7	-	-	-	-	-	-	-
AIRR(w/o mask+L_d)	40.8	73.1	50.7	93.2	63.1	71.4	40.1	83.3	64.5	51.3	67.9
AIRR(w/o mask)	**54.8**	76.9	**58.4**	95.4	**88.2**	**79.0**	**49.2**	**88.0**	**73.7**	**60.9**	**75.5**

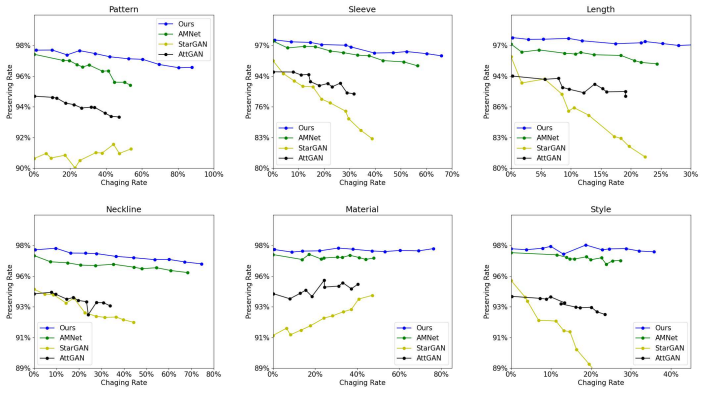


Fig. 3: Attribute changing rate vs. attribute preservation rate. The interval of y axis is made to be unequal for better visualization purposes

Table 6: A/B user judgements for attribute manipulation correctness on the DeepFashion Fine-grained Attribute dataset

AIRR/StarGAN	AIRR/AttGAN	AIRR/AMNet
65%/35%	61%/39%	54%/46%

Table 7: A/B user judgements for attribute manipulation correctness on the CelebA-HQ dataset

AIRR/InterfaceGAN	AIRR/LatentTrans
76%/24%	70%/30%

think our method achieves better attribute manipulation on the DeepFashion Fine-grained Attribute dataset. On CelebA-HQ, reported in Table 7, 70-76% workers voted for AIRR, verifying its improved capability of manipulating facial attributes compared to prior work.

4.4 Qualitative Results

Figure 4-7 presents some qualitative examples on each dataset that we used. For attributes that are relatively shallow and easy to learn, such as *color* in Figure 5, all methods perform well in transforming the source attribute into the target attribute. Whereas for attributes with relatively more complicated semantics, *e.g.*, *lattice* in Figure 5, our method better represents the target

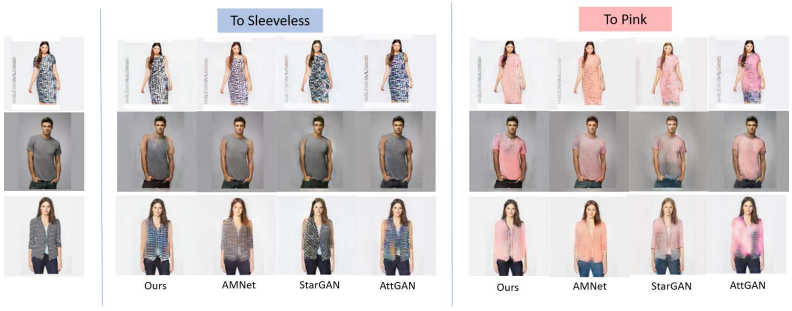


Fig. 4: Qualitative examples on DeepFashion Synthesis. The first column shows original images

Fig. 5: Qualitative examples on DeepFashion Fine-grained Attribute

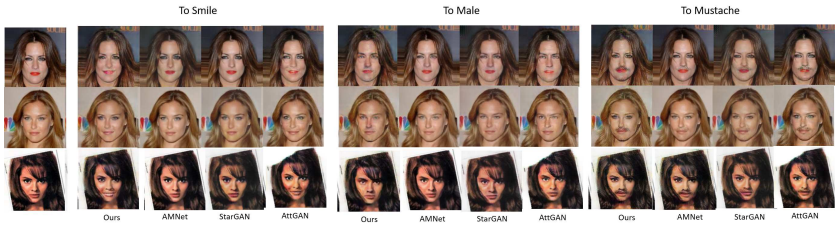


Fig. 6: Qualitative examples on CelebA

attribute in the generated images. In addition, it can also be observed that due to the information hiding problem, several failures of previous methods exhibit as visually not altering the original image, such as AMGAN's *To Denim* in Figure 5 and LatentTrans's *To Hat* in Figure 7. On the other hand, although our method avoids using the source attribute information for manipulation, it's failures can be more significant due to excessive manipulation. For example, the last two rows of *To Floral* in Figure 5 alter the dresses' appearance completely.

4.5 Model Analysis

Ablations of model components. To demonstrate the contribution of the components of our proposed framework, we evaluate the performance of our model without the disentanglement loss and the parsing mask. Tables 2-5 report quantitative results of ablations of our model. AIRR (w/o L_d), which disables the attribute remover, causes losses on average attribute manipulation accuracy and top-5 retrieval on all four datasets. Especially in CelebA-HQ, AIRR (w/o L_d) losses 9% accuracy compared to AIRR. This suggests that disentangling attribute information by decorrelation is effective in image manipulation. We also explored what the generated images would look like when injecting no attribute, *i.e.*, setting the target attribute's scale embedding vector to be $\mathbf{1}$ and the bias embedding vector to be $\mathbf{0}$. This way we can check if the model is hiding source attribute information in the encoded features. If it suffers from information hiding, then the source attributes should been seen in generated images. In Figure 8a, without the proposed disentanglement loss, most source attributes, including material and pattern, indeed appear in the generated images. With the proposed attribute excluded representation, all these attribute information is successfully removed, avoiding the information hiding problem suffered by prior work.

In the meanwhile, AIRR (w/o mask), which removes the parsing mask along with the second encoder, also degrades the evaluation metrics as seen in TABles 2-5. This indicates the importance of concentrating manipulation on the object of interest. However, we note that even without the parsing mask, our approach still outperforms prior work on most metrics. Note that in the two ablations of CelebA-HQ, we didn't add the parsing mask in order to reduce the computational costs for generating high resolution images.

We also tried replacing the proposed disentanglement loss with the honesty loss in [4], which was introduced to avoid the general information hidden problem when using cycle consistency. In Table 2 and 3, AIRR still outperforms AIRR (w L_h) that adopts the honesty loss, suggesting that the proposed disentanglement loss is more targeted to the attribute manipulation task. See the Supplementary for more ablation results for hyperparameters used by our model.

Interpolation of attribute values. Linear interpolating between different attribute embedding vectors $\boldsymbol{\beta}_{\boldsymbol{a}_i}$ and $\boldsymbol{\gamma}_{\boldsymbol{a}_i}$ corresponds to an interpolation between different values of the target attribute. Take "smile" for example, Figure 8b gives the outputs of interpolating from the *not smiling* embedding vector to the *smiling* embedding vector. Let c be the weight (*i.e.*, interpolation coefficient) of the target attribute *smiling*. The smile in generated images gradually builds up as c increases, showing a continuous control over the attribute strength.

Controlling multiple attributes in one forward pass. In some prior work, *e.g.*, [35,31,33], multi-attribute editing is often accomplished by sequential manipulation, *i.e.*, edit one attribute at a time. In contrast, AIRR is capable of changing multiple attributes in a single forward-pass by directly specifying the input target attributes in $\bar{\mathbf{A}}$. Figure 8c gives some examples on CelebA-HQ. Even manipulating 3 or 4 attributes at the same time, our model is able to edit only the specified attributes without influencing other information in the image.

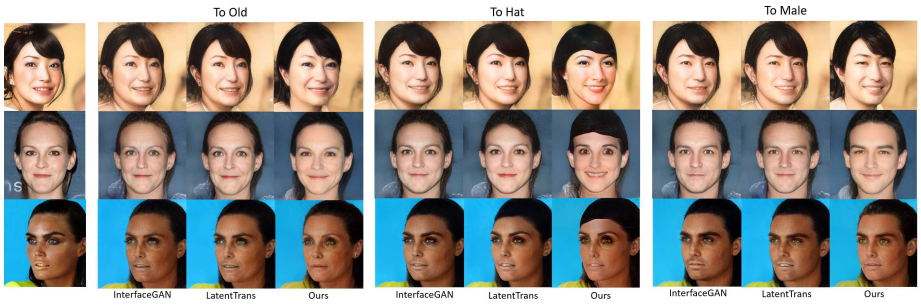
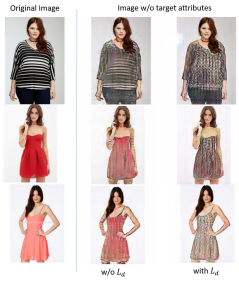
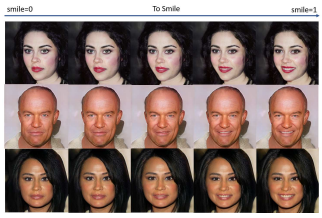


Fig. 7: Qualitative examples on CelebA-HQ



(a) Examples on images without target attributes



(b) Examples on controlling attribute strength

(c) Examples on controlling multiple attributes

Fig. 8: Visualizations used for model analysis in Sec. 4.5

5 Conclusion

In this paper, we propose Attribute Information Removal and Reconstruction (AIRR) network for image editing. The attribute information removal and reconstruction module in AIRR produces an attribute excluded representation, eliminating sources of information hiding suffered by prior work. Results on four diverse datasets including DeepFashion Synthesis, DeepFashion Fine-grained Attribute, CelebA and CelebA-HQ, report that our model improves attribute manipulation accuracy and top-k retrieval rate by 10% on average over prior work. A user study also demonstrates that images with attributes manipulated with our approach are preferred in up to 76% of cases. One direction for future work is to explore controllable attribute manipulation in unsupervised setting.

Acknowledgements. This material is based upon work supported, in part, by DARPA under agreement number HR00112020054 and the National Science Foundation under Grant No. DBI-2134696. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the supporting agencies.

References

1. Abdal, R., Qin, Y., Wonka, P.: Image2stylegan++: How to edit the embedded images? In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2020)
2. Ak, K.E., Lim, J.H., Sun, Y., Tham, J.Y., Kassim, A.A.: Fashionsearchnet-v2: Learning attribute representations with localization for image retrieval with attribute manipulation. arXiv preprint arXiv:2111.14145 (2021)
3. Ak, K.E., Lim, J.H., Tham, J.Y., Kassim, A.A.: Attribute manipulation generative adversarial networks for fashion images. In: Proceedings of the IEEE International Conference on Computer Vision (2019)
4. Bashkirova, D., Usman, B., Saenko, K.: Adversarial self-defense for cycle-consistent GANs. arXiv preprint arXiv:1908.01517 (2019)
5. Burns, A., Sarna, A., Krishnan, D., Maschinot, A.: Unsupervised disentanglement without autoencoding: Pitfalls and future directions. arXiv preprint arXiv:2108.06613 (2021)
6. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: Proceedings of the International Conference on Neural Information Processing Systems (2016)
7. Choi, Y., Choi, M., Kim, M., Ha, J., Kim, S., Choo, J.: StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018)
8. Choi, Y., Uh, Y., Yoo, J., Ha, J.W.: StarGAN v2: Diverse image synthesis for multiple domains. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2020)
9. Härkönen, E., Hertzman, A., Lehtinen, J., Paris, S.: GANSpace: Discovering interpretable GAN controls. In: Proceedings of the IEEE Conference on Neural Information Processing Systems; (2020)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016)
11. He, Z., Zuo, W., Kan, M., Shan, S., Chen, X.: AttGAN: Facial attribute editing by only changing what you want. IEEE transactions on image processing **28**(11), 5464–5478 (2019)
12. Hou, Y., Vig, E., Donoser, M., Bazzani, L.: Learning attribute-driven disentangled representations for interactive fashion retrieval. In: Proceedings of the IEEE International Conference on Computer Vision (2021)
13. Hu, Q., Szabó, A., Portenier, T., Favaro, P., Zwicker, M.: Disentangling factors of variation by mixing them. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2018)
14. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2019)
15. Kwak, J.g., Han, D.K., Ko, H.: CAFE-GAN: arbitrary face attribute editing with complementary attention feature. In: Proceedings of the European Conference on Computer Vision (2020)
16. Kwon, Y., Petrangeli, S., Kim, D., Wang, H., Swaminathan, V., Fuchs, H.: Tailor me: An editing network for fashion attribute shape manipulation. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision (2022)

17. Lee, C.H., Liu, Z., Wu, L., Luo, P.: MaskGAN: Towards diverse and interactive facial image manipulation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2020)
18. Lezama, J.: Overcoming the disentanglement vs reconstruction trade-off via jacobian supervision. In: International Conference on Learning Representations (2018)
19. Li, P., Xu, Y., Wei, Y., Yang, Y.: Self-correction for human parsing. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020). https://doi.org/10.1109/TPAMI.2020.3048039
20. Liu, X., Thermos, S., Valvano, G., Chartsias, A., O'Neil, A., Tsaftaris, S.A.: Measuring the biases and effectiveness of content-style disentanglement. In: Proceedings of the British Machine Vison Conference (2021)
21. Liu, Z., Luo, P., Qiu, S., Wang, X., Tang, X.: DeepFashion: Powering robust clothes recognition and retrieval with rich annotations. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (2016)
22. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: Proceedings of International Conference on Computer Vision (2015)
23. Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., Bachem, O.: Challenging common assumptions in the unsupervised learning of disentangled representations. In: International Conference on Machine Learning (2019)
24. Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Paul Smolley, S.: Least squares generative adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision (2017)
25. Ramesh, A., Choi, Y., LeCun, Y.: A spectral regularizer for unsupervised disentanglement. In: International Conference on Machine Learning (2018)
26. Shen, Y., Gu, J., Tang, X., Zhou, B.: Interpreting the latent space of GANs for semantic face editing. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2020)
27. Shin, M., Park, S., Kim, T.: Semi-supervised feature-level attribute manipulation for fashion image retrieval. In: Proceedings of the British Machine Vison Conference (2019)
28. Shoshan, A., Bhonker, N., Kviatkovsky, I., Medioni, G.: GAN-control: Explicitly controllable GANs. In: Proceedings of the IEEE International Conference on Computer Vision (2021)
29. Szabo, A., Hu, Q., Portenier, T., Zwicker, M., Favaro, P.: Understanding degeneracies and ambiguities in attribute transfer. In: Proceedings of the European Conference on Computer Vision (2018)
30. Usman, B., Bashkirova, D., Saenko, K.: Disentangled unsupervised image translation via restricted information flow (2021)
31. Wang, R., Chen, J., Yu, G., Sun, L., Yu, C., Gao, C., Sang, N.: Attribute-specific control units in StyleGAN for fine-grained image manipulation. In: Proceedings of the ACM International Conference on Multimedia (2021)
32. Wu, Z., Lischinski, D., Shechtman, E.: StyleSpace analysis: Disentangled controls for StyleGAN image generation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2021)
33. Yang, G., Fei, N., Ding, M., Liu, G., Lu, Z., Xiang, T.: L2M-GAN: Learning to manipulate latent space semantics for facial attribute editing. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2021)
34. Yang, X., Song, X., Han, X., Wen, H., Nie, J., Nie, L.: Generative attribute manipulation scheme for flexible fashion search. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval (2020)

35. Yao, X., Newson, A., Gousseau, Y., Hellier, P.: A latent transformer for disentangled face editing in images and videos. In: Proceedings of the IEEE International Conference on Computer Vision (2021)
36. Yu, C., Wang, J., Peng, C., Gao, C., Yu, G., Sang, N.: Bisenet: Bilateral segmentation network for real-time semantic segmentation. In: Proceedings of the European Conference on Computer Vision (2018)
37. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision (2017)

Supplementary

1 Proof of the Upper Bound for Mutual Information

In Eq. (2) of the paper, we claimed that the mutual information between the source attributes $\mathbf{a}_i$ and the attribute excluded features $R(E_1(I_{cl}))$ is upper bounded by the maximum log probability in the attribute distribution. We prove this claim in the following.

Let $\mathrm{MI}(\mathbf{a}_i, R(E_1(I_{cl})))$ denote the mutual information. Replacing $R(E_1(I_{cl}))$ with r for convenience gives

$$\begin{aligned}\mathrm{MI}(\mathbf{a}_i, r) &= \sum\nolimits_r \sum\nolimits_{\mathbf{a}_i} p(\mathbf{a}_i, r) \log \frac{p(\mathbf{a}_i, r)}{p(\mathbf{a}_i)p(r)} \\ &= \sum\nolimits_r \sum\nolimits_{\mathbf{a}_i} p(\mathbf{a}_i, r) \log \frac{p(\mathbf{a}_i|r)}{p(\mathbf{a}_i)} \\ &= \sum\nolimits_r \sum\nolimits_{\mathbf{a}_i} p(\mathbf{a}_i, r)[\log p(\mathbf{a}_i|r) - \log p(\mathbf{a}_i)]\end{aligned} \tag{15}$$

Since the number of attribute values in $\mathbf{a}_i$ is finite, $-\log p(\mathbf{a}_i)$ can be upper bounded by a constant $c, c > 0$:

$$\begin{aligned}\mathrm{MI}(\mathbf{a}_i, r) &\leq \sum\nolimits_r \sum\nolimits_{\mathbf{a}_i} p(\mathbf{a}_i, r) \log p(\mathbf{a}_i|r) + c \sum\nolimits_r \sum\nolimits_{\mathbf{a}_i} p(\mathbf{a}_i, r) \\ &= \sum\nolimits_r \sum\nolimits_{\mathbf{a}_i} p(\mathbf{a}_i, r) \log p(\mathbf{a}_i|r) + c\end{aligned} \tag{16}$$

In the r.h.s., we can continue upper bounding $p(\mathbf{a}_i|r)$ with the maximum probability in the distribution to make it independent of $\mathbf{a}_i$:

$$\begin{aligned}\mathrm{MI}(\mathbf{a}_i, r) &\leq \sum\nolimits_r \max_{\mathbf{a}_i} \log p(\mathbf{a}_i|r) \sum\nolimits_{a_i} p(\mathbf{a}_i, r) + c \\ &= \sum\nolimits_r p(r) \max_{\mathbf{a}_i} \log p(\mathbf{a}_i|r) + c \\ &= \mathbb{E}_{r\sim p(r)}[\max_{\mathbf{a}_i} \log p(a_i|r)] + c,\end{aligned} \tag{17}$$

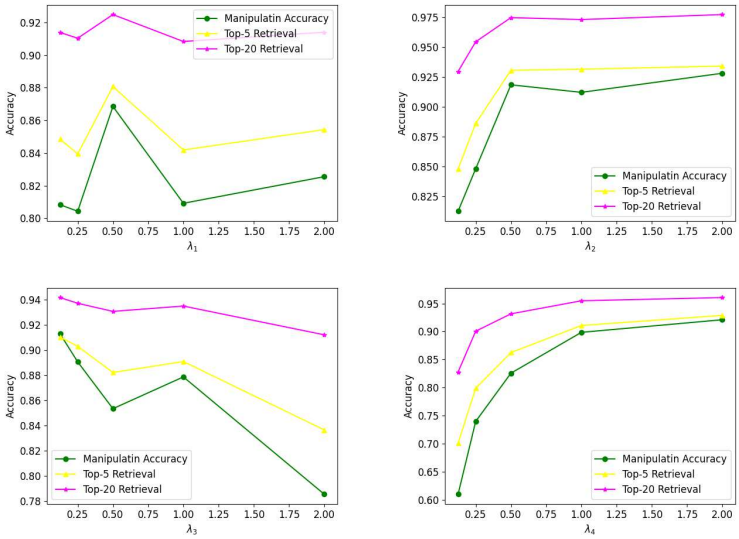
where c is a constant. Note that we can not minimize the mutual information itself because the joint distribution $p(\mathbf{a}_i, r)$ is intractable. The tightness of this upper bound depends on the distribution $p(\mathbf{a}_i)$ and $p(\mathbf{a}_i|r)$. More specifically, larger $\min_{\mathbf{a_i}} p(\mathbf{a_i})$ gives smaller constant c, and smaller $\max_{\mathbf{a_i}} p(\mathbf{a_i}|r)$ reduces the gap. The equality is reached when $p(\mathbf{a_i}|r)$ is an uniform distribution.

To conclude, using an attribute classifier to estimate the above conditional probability $p(a_i|r)$, we prove that the upper bound is the maximum log probability in the attribute distribution as in Eq. (2).

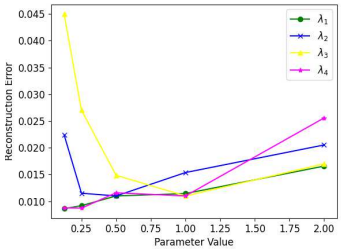
2 Ablations on Hyperparameters

In Figure 9, we provide the experimental results for setting different values of the hyperparameters in Eq. (13) and (14) on CelebA. λ_1 to λ_4 denotes the trade-off

parameter for disentanglement, image attribute prediction, image reconstruction and perceptual loss, respectively. Figure 9a shows the manipulation accuracy, top-5 retrieval and top-20 retrieval rates for each parameter. The reconstruction error has a different unit of measurement, for which we show its corresponding graph in Figure 9b. It can be noticed that increasing the weight (*i.e.*, λ_2) for the image attribute loss improves the manipulation accuracy, whereas it can hurt the reconstruction performance. This indicates a trade-off between successful manipulation and qualitative reconstruction. In the paper, we chose the values of each trade-off parameter for a balance between these two aspects.



(a) Parameter value v.s. Accuracy. Higher is better



(b) Parameter Value v.s. Reconstruction Error. Lower is better

Fig. 9: Results on using different values of the hyperparameters. λ_1 to λ_4 denotes the trade-off parameters for disentanglement, image attribute prediction, image reconstruction and perceptual loss, respectively

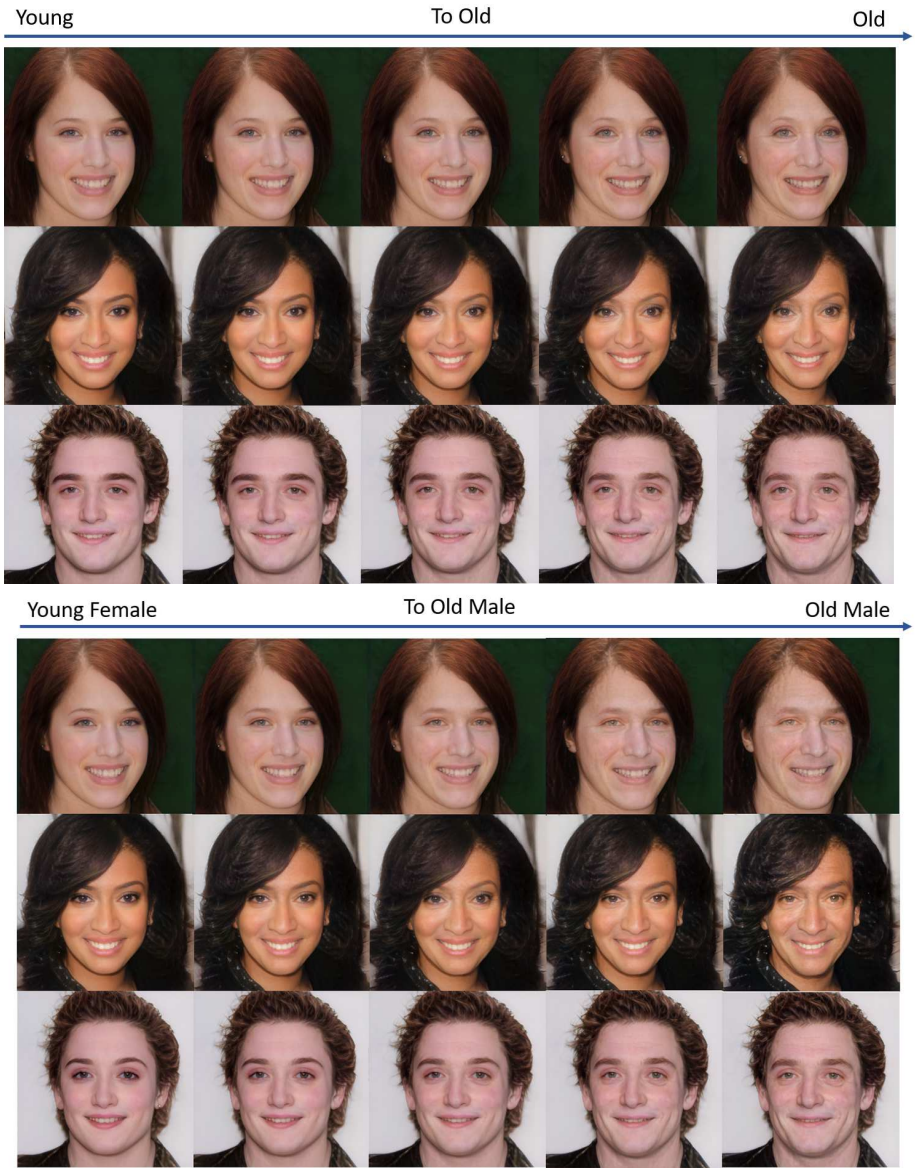


Fig. 10: Additional examples on manipulating the attribute strength