

MagGAN: High-Resolution Face Attribute Editing with Mask-Guided Generative Adversarial Network

Yi Wei¹, Zhe Gan², Wenbo Li³, Siwei Lyu⁴, Ming-Ching Chang¹, Lei Zhang², Jianfeng Gao², and Pengchuan Zhang²

¹ University at Albany, State University of New York, USA

² Microsoft Corporation, Redmond, USA

³ Samsung Research America AI Center, USA

⁴ University at Buffalo, State University of New York, USA

This supplementary material has 5 sections. Section 1 describes the network architecture of MagGAN. Section 2 shows some facial editing results on resolution 256×256 . Section 3 demonstrates some visual results on high resolution 512×512 and 1024×1024 . Section 4 defines the attribute and facial part relationship matrix \mathbf{AR}^+ , \mathbf{AR}^- . Section 5 provides formal definition of evaluation metrics - PSNR and SSIM. We use § to refer the section in our submitted paper.

1 Network Architecture of MagGAN

We present MagGAN network architecture for image generators in Table 2 and the network architectures for discriminators in Table 3. They are built with basic blocks defined in Table 1. We reached this architecture design by extensive architecture search based on STGAN, as we present below.

Architecture optimization based on STGAN We first conduct hyper-parameter tuning for STGAN [3] on resolution 256×256 , and compare the attribute editing accuracy and FID to select the best architecture. First, we apply both cycle-consistency loss [1] \mathcal{L}^{cycle} and the reconstruction loss used in AttGAN [2] \mathcal{L}^{rec} to train generator, but combine the two losses with a weight $C \in [0, 1]$. Then the total reconstruction loss \mathcal{L}_G^R is defined as:

$$\mathcal{L}_G^R = C \cdot \mathcal{L}^{rec} + (1 - C) \cdot \mathcal{L}^{cycle}$$

From Figure 3, we find that only applying the reconstruction loss achieves the best accuracy and FID. From Figure 4, we also find that increasing the layer of discriminator and generator from 5 to 6 improves the attribute editing accuracy and FID. Also, in the original STGAN discriminator, images are fed into a shared convolution layer, and the feature maps are then used by two separate branches for adversarial prediction and attribute classification. We observe that applying average pooling after the shared convolution layer improves the attribute editing accuracy.

Learning rate optimization to stabilize training In our experiment setting, we set encoding/decoding layer of generator to 6. The shared convolution backbone layer of the vanilla discriminator or PatchGAN discriminators is also set to 6. The illustration of network architecture for generator and discriminators are shown in Figure 1 and Figure 2. To make generator training stable, the learning rate of generator is set to 0.0001 according to Figure 5, while learning rate of discriminator is set to 0.0002.

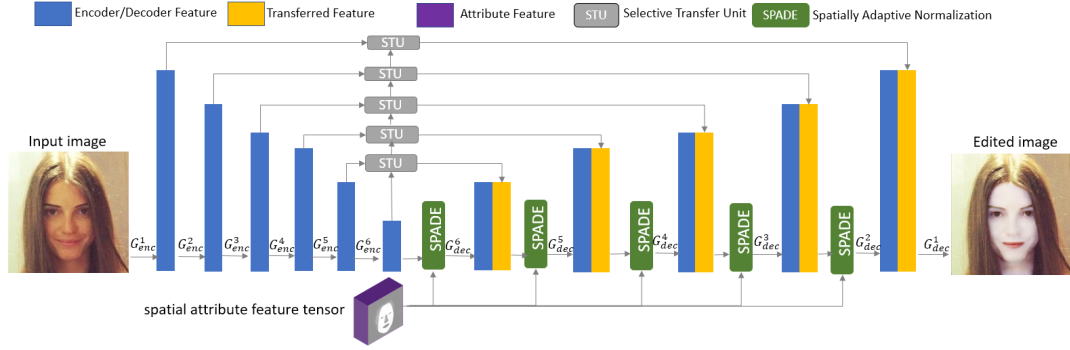


Fig. 1: Network architecture of MagGAN generator

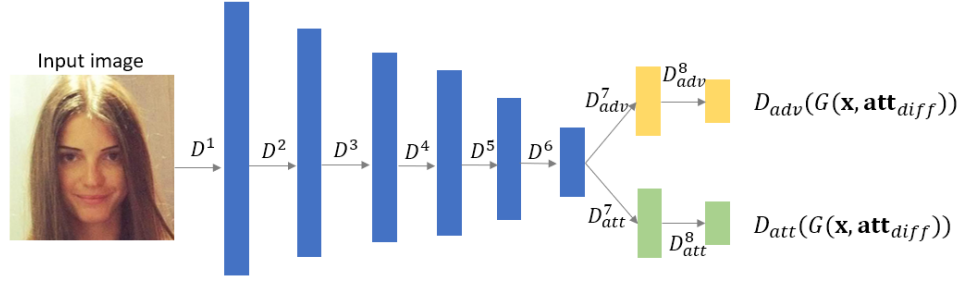


Fig. 2: Network architecture of MagGAN discriminator

Hyper-parameter for mask-guided reconstruction weight With extensive experiments, we find that to achieve reasonable visual effects for synthesized images, both mask-guided attribute conditioning and mask-guided reconstruction loss should be applied. We present the effects of mask-guided reconstruction weight λ_3 in Table 4. To achieve a better balance between editing accuracy and preserving quality, we choose $\lambda_3 = 200$ in practice.

Unified architecture for a single discriminator and multi-level patch-wise discriminators We unify the architecture of vanilla STGAN discriminator and PatchGAN discriminator in Table 3. The difference is that level- i PatchGAN discriminator works on different resolution, from 256 to 1024, the adversarial outputs of PatchGAN discriminators ($i = 0, 1, 2$) are of size (1, 1), (2, 2), (4, 4) respectively. Each output entry represents a real/fake output corresponding to a 256×256 patch.

Table 1: The basic blocks for architecture design. (“-” connects two consecutive layers; “+” means element-wise addition between two layers, * means element-wise multiplication between two layers.) F_{dec} , F_{attr} , F_{trans} denotes decoding feature, spatial attribute feature, transferred feature as shown in Figure 1

Name	Operations / Layers
Concat	Concatenate input tensors along the channel dimension.
Downsample	Nearest neighbor Downsampling layer
BN	Batch normalization layer
IN	Instance normalization layer
LN wo/ affine	Layer normalization layer without apply affine transformation
Conv(dim , k , s)	Convolutional layer with output dimension dim , kernel size k , stride s
DeConv(dim , k , s)	Transposed convolutional layer with output dimension dim , kernel size k , stride s
STU	Selective transfer unit proposed by STGAN [3]
SPADE	Spatially adaptive normalization layer [4]: $\beta = \text{Concat}(F_{dec}, \text{Downsample}(F_{attr})) - \text{Conv}(d,3,1) - \text{Conv}(d,3,1)$ $\gamma = \text{Concat}(F_{dec}, \text{Downsample}(F_{attr})) - \text{Conv}(d,3,1) - \text{Conv}(d,3,1)$ LN wo/ affine (F_{dec}) - Conv(d , 3, 1) * $\gamma + \beta$
Avgpool (os)	Average pooling with output size os .

Table 2: Network architecture of MagGAN generator. G_{enc}^l and G_{dec}^l denotes the encoding layer and decoding layer of generator at layer l respectively. The input feature of DeConv layer is the concatenation of decoding feature and selective feature. SPADE is applied as normalization layer for transposed convolution feature

l	G_{enc}^l	G_{dec}^l
1	Conv(64, 4, 2), BN, Leaky ReLU	DeConv(3, 4, 2), Tanh
2	Conv(128, 4, 2), BN, Leaky ReLU	DeConv(128, 4, 2), SPADE, ReLU
3	Conv(256, 4, 2), BN, Leaky ReLU	DeConv(256, 4, 2), SPADE, ReLU
4	Conv(512, 4, 2), BN, Leaky ReLU	DeConv(512, 4, 2), SPADE, ReLU
5	Conv(1024, 4, 2), BN, Leaky ReLU	DeConv(1024, 4, 2), SPADE, ReLU
6	Conv(1024, 4, 2), BN, Leaky ReLU	DeConv(1024, 4, 2), SPADE, ReLU

Table 3: Network architecture of MagGAN discriminator/PatchGAN discriminator. i denote the level of PatchGAN discriminator, $i = \{0, 1, 2\}$ corresponds to resolution 256, 512, 1024 respectively. When $i = 0$, PatchGAN discriminator is equal to single vanilla discriminator applied on resolution 256. c denotes the attribute class numbers. D_{adv}^l and D_{att}^l denotes the adversarial learning branch and attribute classification branch respectively, they share the same convolution backbone

l	D_{adv}^l	D_{att}^l
1	Conv(64, 4, 2), IN, Leaky ReLU	
2	Conv(128, 4, 2), IN, Leaky ReLU	
3	Conv(256, 4, 2), IN, Leaky ReLU	
4	Conv(1024, 4, 2), IN, Leaky ReLU	
5	Conv(1024, 4, 2), IN, Leaky ReLU	
6	Conv(1024, 4, 2), IN, Leaky ReLU	
7	Avgpool(2^i)	
8	Conv(1024, 1, 1), Leaky ReLU	Conv(1024, 1, 1), Leaky ReLU
9	Conv(1, 1, 1)	Conv(c , 1, 1)

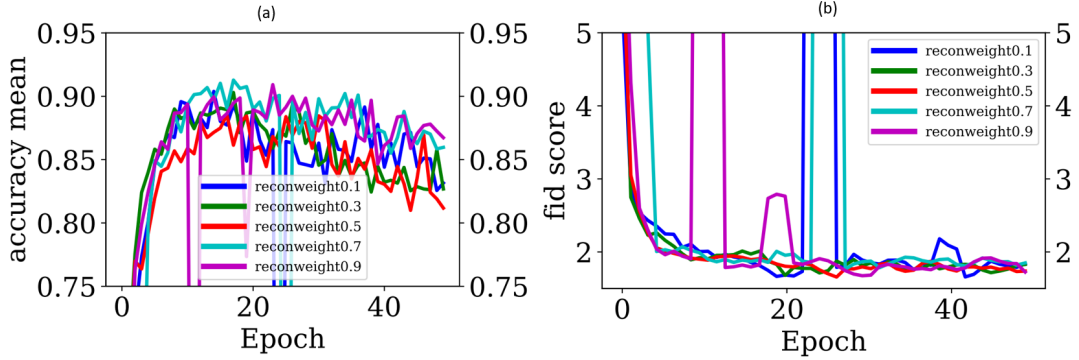


Fig. 3: Attribute editing accuracy and FID comparison for reconstruction weight tuning. 'reconweight' ranges from 0 to 1, 'reconweight' = 0 means that only cycle-consistency loss is applied, 'reconweight' = 1 means that only reconstruction loss is applied

Table 4: Comparison of different mask-guided reconstruction weight λ_3 for MagGAN

Methods	MRE ↓	FID ↓	Avg Acc	PSNR	SSIM
$\lambda_3 = 0$	0.0397	1.22	89.6%	39.35	0.980
$\lambda_3 = 100$	0.0232	1.39	85.6%	38.57	0.976
$\lambda_3 = 200$	0.0163	1.10	90.0%	40.25	0.984
$\lambda_3 = 400$	0.0157	1.33	88.2%	39.34	0.984

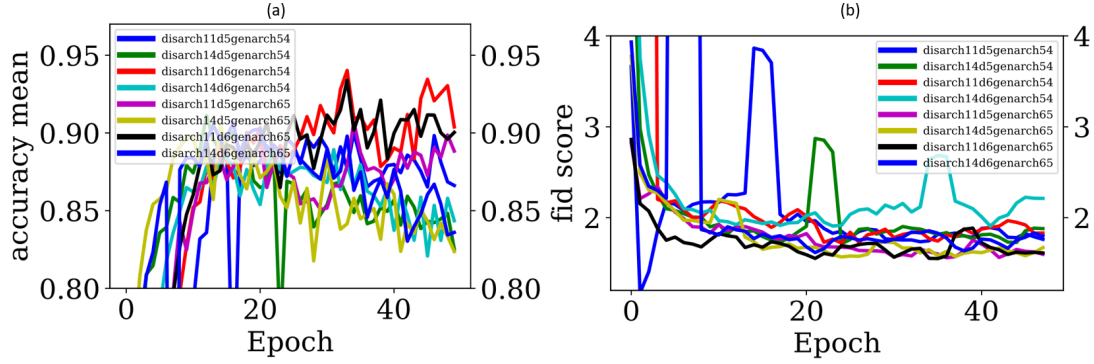


Fig. 4: Attribute editing accuracy and FID comparison for architecture search. 'disarch11' means that average pooling is applied after the last shared convolution layer, output size is 1. 'disarch14' means no average pooling after the final convolution layer. 'd5', 'd6' means that the discriminator has 5 or 6 shared convolution layers. 'genarch54', 'genarch65' means that generator has 5 encoding-decoding layers, 4 STU layers or 6 encoding-decoding layers, 5 STU layers

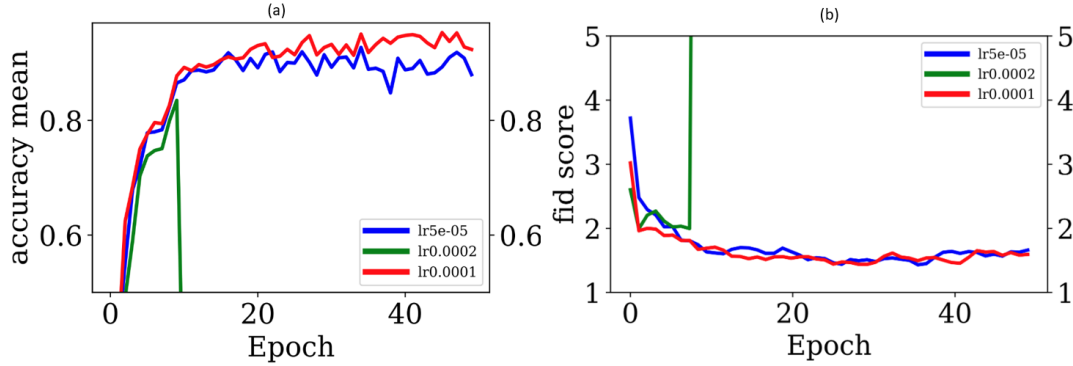


Fig. 5: Attribute editing accuracy and FID comparison for generator learning rate tuning. We test 3 learning rate: 5×10^{-5} , 1×10^{-4} , 2×10^{-4}

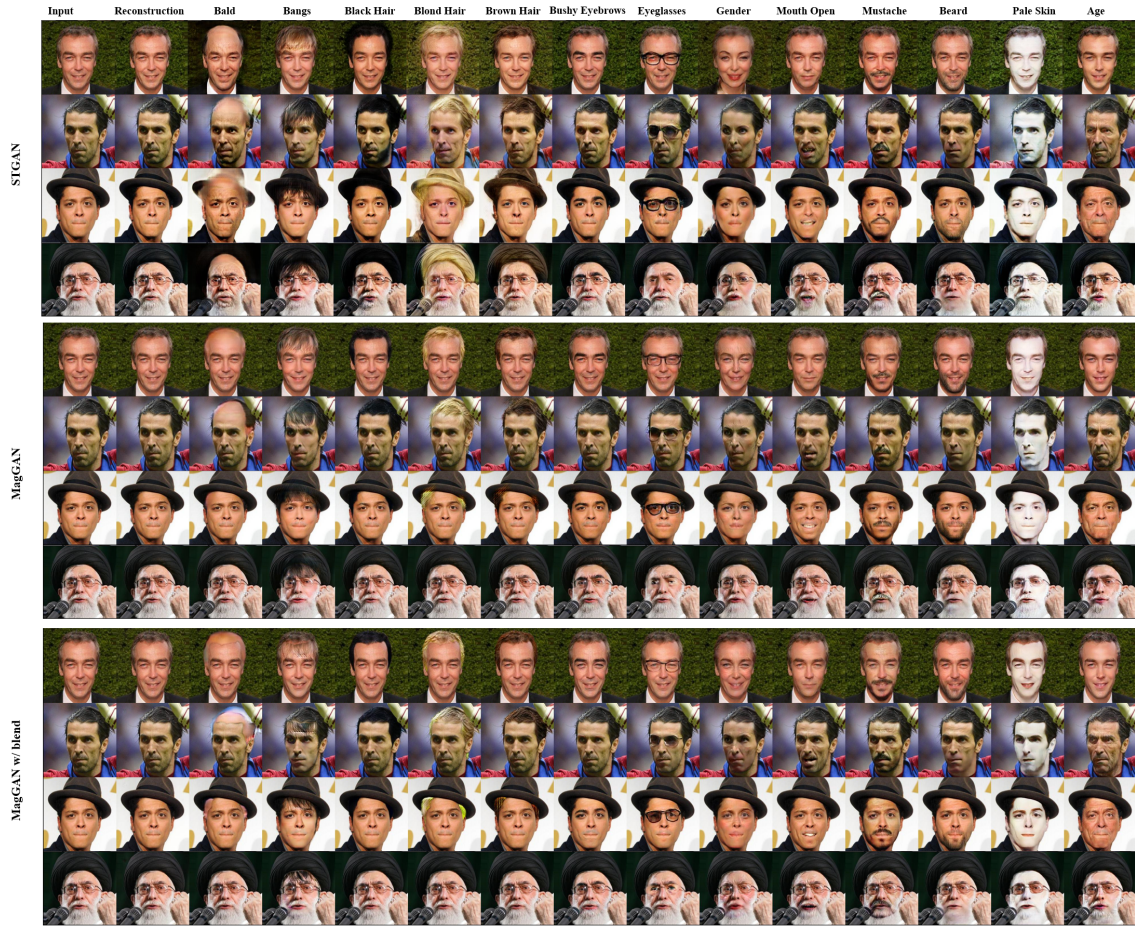


Fig. 6: Visual results of STGAN, MagGAN, MagGAN w/ blend on resolution 256×256

2 Face Attribute Editing Results on Resolution 256

In this section, we show more visual results of MagGAN on resolution 256×256 . Figure 6 shows single-attribute reverse editing, Figure 9 shows multiple-attribute reverse editing, and Figure 10 shows the editing results when attribute intensity varies continuously from 0 to 1.

The blending trick to preserve attribute irrelevant regions In §3.3 of our submission, we propose a blending trick to help preserve the attribute-irrelevant regions with alpha composition [5]. We adopt this blending trick to MagGAN(SP) and report the quantitative results in Table 5. Results show that except MRE reduces significantly, the other metrics are worse than MagGAN when applying the blending trick. From Figure 6, we can also observe that the blending trick generates sharp images, but the visual quality decreases as artifacts are obvious at the boundary of attribute-irrelevant regions.

Table 5: Comparison of MagGAN with blending trick on resolution 256×256 . The blending trick does decrease the mask-aware reconstruction error, but incorporates artifacts at borders, which diminish the visual quality

Methods	MRE ↓	FID ↓	Avg Acc	PSNR	SSIM
MagGAN	0.0163	1.10	90.0%	40.25	0.984
MagGAN w/ blend	0.0015	1.14	83.3%	37.70	0.976

User study We conduct user study on Amazon Mechanical Turk to compare the generation quality of STGAN and MagGAN. Figure 7 shows the web interface of our user study experiment. 100 input samples are randomly chosen from test set, 50 samples with hat or scarf and 50 samples without. For each sample, 5 attribute editing tasks are performed by STGAN and MagGAN (500 comparison pairs in total). All 5 tasks are randomly chosen from 13 attributes, for subjects with hat, we increase the chance to select hair related attributes, *e.g.*, “Blonde Hair”, “Bald”. The users are instructed to choose the best result which changes the attribute more successfully considering image quality and identity preservation. The user interface also provides a neutral option, which can be selected if the turker thinks both outputs are equally good. To avoid human bias, each sample pair is evaluated by 3 volunteers, thus we have 1500 comparison pairs in total. Only workers with a task approval rate greater than 95% can participate the study.

Figure 8 shows some example visual results for MagGAN and STGAN. Top 3 rows are samples wearing hat or scarf, the last 3 rows are samples without hat. From our observation, MagGAN works better on preserving hat or background regions for with-hat samples, the editing quality also improves for without-hat samples due to the help of mask information. In Table 2 of our submission, we find the gap between MagGAN and STGAN on with-hat samples is not significantly large. We made a meticulous investigation on the collected user study results. We find that the user may misunderstand our instructions by choosing the model with more obvious editing results. For example, the 3rd row in Figure 8, STGAN achieve more obvious change on “To Bald” and “To Blonde Hair” attributes, but in fact, STGAN changes the hat regions which should stay intact. In that situation, MagGAN should be considered as the better model. Thus, such “failure cases” decrease the votes to MagGAN.

Human Evaluation Instructions

We are interested in face image editing. Please evaluate the quality of the edited face images according to the instructions below.

For each group, there are one source face image and one desired face attribute. The system will generate one output image with the desired face attribute, while keeping the attribute-irrelevant face regions unchanged. The task is to evaluate which edited face image is better. We consider 13 attributes in total: ['Bald', 'Bangs', 'Black_Hair', 'Blond_Hair', 'Brown_Hair', 'Bushy_Eyebrows', 'Eyeglasses', 'Male', 'Mouth_Slightly_Open', 'Mustache', 'No_Beard', 'Pale_Skin', 'Young']. Below, we show two examples.















The edited images in the 1st row are better than the edited images in the 2nd row, because although the edited images in both rows successfully contain the desired attributes (Bald is not visible for the 2nd image due to the hat), however, in the 2nd row, some attribute-irrelevant regions (e.g., the scarf and the hat) are also edited, thus, images in the 1st row are better.

Your Job:

Given an input face image, please evaluate which edited face image is better aligned with the desired attribute. A good edited face image should have good image quality, contain the desired face attribute, while also preserving other attribute-irrelevant face regions unchanged.

Try not to select the "Cannot tell" button unless you really cannot tell which one is better.

Task 1: Input Face Image: 	Desired Face Attribute: Bald	Edited Face Image A: 	Edited Face Image B: 	<input type="radio"/> Image A is better <input type="radio"/> Image B is better <input type="radio"/> Cannot tell
Task 2: Input Face Image: 	Desired Face Attribute: No Beard	Edited Face Image A: 	Edited Face Image B: 	<input type="radio"/> Image A is better <input type="radio"/> Image B is better <input type="radio"/> Cannot tell
Task 3: Input Face Image: 	Desired Face Attribute: Bald	Edited Face Image A: 	Edited Face Image B: 	<input type="radio"/> Image A is better <input type="radio"/> Image B is better <input type="radio"/> Cannot tell
Task 4: Input Face Image: 	Desired Face Attribute: Black Hair	Edited Face Image A: 	Edited Face Image B: 	<input type="radio"/> Image A is better <input type="radio"/> Image B is better <input type="radio"/> Cannot tell

You must ACCEPT the HIT before you can submit the results.

Fig. 7: Amazon Mechanical Turk interface of user study. Users are asked to choose the better edited image considering desired attribute

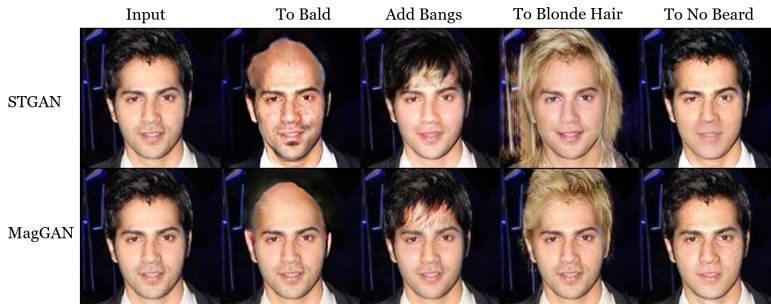




Fig. 9: Visual results of MagGAN for multiple facial attribute editing on resolution 256×256



Fig. 10: Illustration of attribute intensity control of MagGAN on resolution 256×256 . The first column is the input image

3 Face Attribute Editing on High Resolution

We provide more visual results of high-resolution image editing in Figure 11 and Figure 12, for resolution 1024×1024 and 512×512 respectively. Fine details of hair and skin can be well reconstructed with the help of PatchGAN discriminator.

4 Definition of attribute-facial part relationship matrix

We find that facial attributes have strong semantic relationship with specific facial parts. For example, the attribute of "blonde hair" is highly related to the hair regions which should be modified in the edited image if this attribute changes. That leads to a pre-defined attribute-facial part relationship matrix \mathbf{AR} that denotes the relevant regions of each attribute changes. With the help of \mathbf{AR} , the preserved mask M to the attribute difference $\mathbf{att}_{\text{diff}}$ can be obtained to computed the mask-guided reconstruction loss (in §3.2) and mask-guided condition attribute feature (in §3.3).

We define two binary attribute-part relation matrices $\mathbf{AR}^+, \mathbf{AR}^- \in [0, 1]^{13 \times 19}$ in our setting (13 modified attributes and 19 facial parts). We separate the attribute changes to two scenarios: attribute strengthen ($0 \rightarrow 1$) or attribute weaken ($1 \rightarrow 0$). The i -th row of matrix \mathbf{AR}^+ or \mathbf{AR}^- indicates which facial parts should be modified when the i -th attribute is strengthened, *i.e.*, $\mathbf{att}_{\text{diff},i} > 0$, or weakened, *i.e.*, $\mathbf{att}_{\text{diff},i} < 0$. The detailed definition is in Figure 13.

5 Quantitative Evaluation Metric

In §4, we apply PSNR (Peak signal-to-noise ratio) and SSIM (Structural Similarity Index) to evaluate the quality of reconstructed images.

PSNR is most commonly used to measure the quality of reconstruction of lossy compression codecs. In our experiment, we denote the original image as I , the reconstructed image as R , which takes its original attribute as target attribute. In theory, the input image I and reconstructed image R should be as similar as possible. PSNR (in dB) is defined as:

$$\begin{aligned} \text{PSNR} &= 10 \cdot \log_{10} \left(\frac{\text{MAX}_I^2}{\text{MSE}} \right), \\ \text{MSE} &= \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - R(i, j)]^2 \end{aligned} \quad (1)$$

MAX_I is the maximum pixel value of input image I . In general, the larger PSNR value, the better quality the reconstructed image is.

SSIM (Structural Similarity Index) [6] is another metric to measure the similarity of image I and image R . The SSIM is defined as:

$$\text{SSIM}(I, R) = \frac{(2\mu_I\mu_R + c_1)(2\sigma_{IR} + c_2)}{(\mu_I^2 + \mu_R^2 + c_1)(\sigma_I^2 + \sigma_R^2 + c_2)} \quad (2)$$

where μ_I, μ_R denotes the average of I and R , σ_I^2, σ_R^2 are the variance of I and R , σ_{IR}^2 denotes the covariance of I and R , c_1, c_2 are small constants to avoid division instability. Also the larger SSIM value denotes better image quality for reconstructed image.

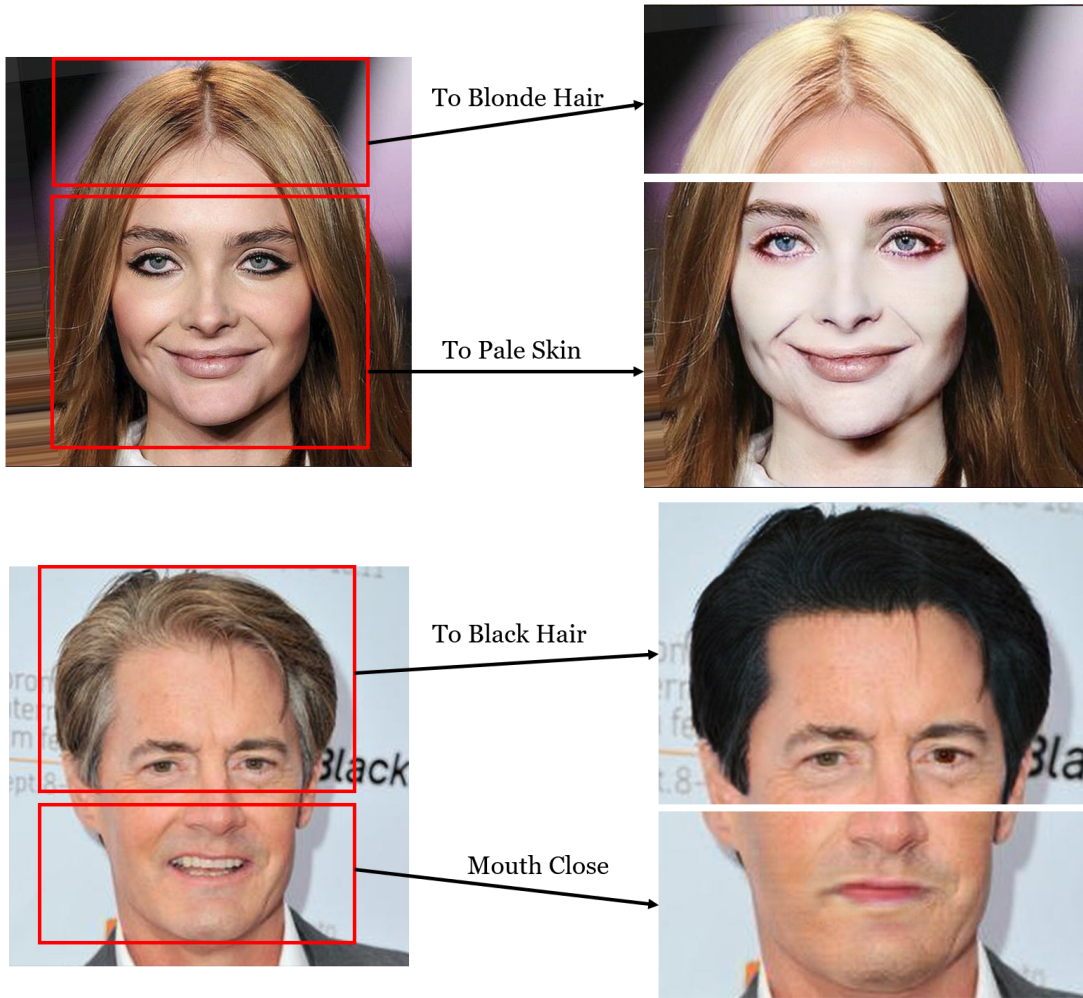


Fig. 11: Visual results of MagGAN (using PatchGAN discriminator) on resolution 1024×1024 . We show the specific sub-regions for better visualization



Fig. 12: Visual results of MagGAN (using PatchGAN discriminator) on resolution 512×512

	Background	Skin	Left Eyebrow	Right Eyebrow	Left Eye	Right Eye	Eyeglasses	Left Ear	Right Ear	Ear Ring	Nose	Mouse	Upper Lip	Lower Lip	Neck	Necklace	Clothes	Hair	Hat		
\mathbf{AR}^+ =	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	Bald	
	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bangs	
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	Black Hair
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	Blond Hair
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	Brown Hair
	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bushy Eyebrows
	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Eyeglasses
	0	1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	0	Male
	0	1	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	Mouse Slightly Open
	0	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	Mustache
	0	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	No Beard
	0	1	0	0	0	0	0	1	1	0	1	0	0	0	1	0	0	0	0	0	Pale Skin
	0	1	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	1	0	0	Young
	\mathbf{AR}^- =	1	1	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	Bald
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	Bangs
0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	Black Hair	
0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	Blond Hair	
0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	Brown Hair	
0		0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Bushy Eyebrows
0		0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	Eyeglasses
0		1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	0	Male
0		1	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	Mouse Slightly Open
0		1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	Mustache
0		1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	No Beard
0		1	0	0	0	0	0	1	1	0	1	0	0	0	1	0	0	0	0	0	Pale Skin
0		1	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	1	0	0	Young

Fig. 13: Definition of attribute-part relationship matrices \mathbf{AR}^+ , \mathbf{AR}^- . Value 1 represents the attribute and facial part are related, 0 represents that they are irrelevant

References

1. Yunjey Choi, Min-Je Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*, pages 8789–8797, 2018. [1](#)
2. Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. Attgan: Facial attribute editing by only changing what you want. *CoRR*, abs/1711.10678, 2017. [1](#)
3. Ming Liu, Yukang Ding, Min Xia, Xiao Liu, Errui Ding, Wangmeng Zuo, and Shilei Wen. STGAN: A unified selective transfer network for arbitrary image attribute editing. *CoRR*, abs/1904.09709, 2019. [1](#), [3](#)
4. Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*, 2019. [3](#)
5. Thomas Porter and Tom Duff. Compositing digital images. In *ACM Siggraph Computer Graphics*, volume 18, pages 253–259. ACM, 1984. [6](#)
6. Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. [12](#)