Semi-Supervised Spatial Attention Method for Facial Attribute Editing

Hyeon Seok Yang[1], Jeong Hoon Han[1], and Young Shik Moon[1*]
[1]Department of Computer Science and Engineering, Hanyang University
Ansan 15588, Korea
[e-mail: hsyang@visionlab.or.kr, bghan@visionlab.or.kr, ysmoon@hanyang.ac.kr]
[*]Corresponding author: Young Shik Moon

Received June 9, 2021; revised September 3, 2021; accepted Septebmer 16, 2021; published October 31, 2021

Abstract

In recent years, facial attribute editing has been successfully used to effectively change face images of various attributes based on generative adversarial networks and encoder–decoder models. However, existing models have a limitation in that they may change an unintended part in the process of changing an attribute or may generate an unnatural result. In this paper, we propose a model that improves the learning of the attention mask by adding a spatial attention mechanism based on the unified selective transfer network (referred to as STGAN) using semi-supervised learning. The proposed model can edit multiple attributes while preserving details independent of the attributes being edited. This study makes two main contributions to the literature. First, we propose an encoder–decoder model structure that learns and edits multiple facial attributes and suppresses distortion using an attention mask. Second, we define guide masks and propose a method and an objective function that use the guide masks for multiple facial attribute editing through semi-supervised learning. Through qualitative and quantitative evaluations of the experimental results, the proposed method was proven to yield improved results that preserve the image details by suppressing unintended changes than existing methods.

Keywords: Facial attribute editing, spatial attention mechanism, semi-supervised learning, generative adversarial network, STGAN.

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2020-0-01343, Artificial Intelligence Convergence Research Center(Hanyang University ERICA))

http://doi.org/10.3837/tiis.2021.10.012 **ISSN : 1976-7277**

1. Introduction

Facial attribute editing research focuses on discovering methods that can edit face images to feature desired attributes. Recently, numerous studies have been conducted to enable realistic editing of various facial attributes via generative adversarial networks (GANs) [1] and encoder–decoder models [2–8]. These methods can be used in various visual applications, such as entertainment, beauty, and art industries.

One of the drawbacks of the methods in existing facial attribute editing research is that, when a specific attribute is changed in an image, other correlated attributes may be unintentionally modified as well [2–8]. If only the attribute-specific area is not edited, an unintended part may be changed, or an unnatural result may be generated. For example, if the hair color is changed to blonde, the background or skin tone itself tends to change, or if the gender is changed, the hair area is unnaturally deformed.

Existing facial attribute editing methods have attempted to suppress unintended attribute changes using several different approaches. However, each model presents unique drawbacks; for example, an unintended area may be distorted [2–5,7], or only one attribute may be edited at a time [2,6]. Certain constraints are quite strict, which prevents sufficient image editing [6]. An approach to solve this problem is to incorporate additional information into the training or to add a loss function to the model to suppress unwanted changes. For example, in recent clustering-related studies, multi-view clustering was utilized to improve the clustering accuracy using data from various sources together [9–11]. In addition, in a study on deep embedding clustering, the performance was improved by applying a contractive autoencoder and adding the Frobenius norm as a penalty term [12].

By systematically combining the methodologies of existing models, we propose a model that can simultaneously train with multiple facial attributes and selectively change the desired areas of the image.

The key contributions of this paper are as follows:

① We propose a model that suppresses the distortion of an area unrelated to the attribute to be edited by simultaneously generating multiple facial attribute editing images, attention masks, and clothing masks with one encoder–decoder structure.

② We propose a method to create a more accurate attention mask by defining guide masks and defining an objective function to use the guide mask as semi-supervised learning.

This paper is organized as follows. Section 2 presents related research and highlights the differences between the proposed and existing methods. Section 3 explains the structure of the proposed method, loss functions, and the use of the guide masks. In Section 4, the experimental settings are presented, and the state-of-the-art methods, AttGAN [3] and the unified selective transfer network (referred to as STGAN) [4], are compared to the proposed method. In Section 5, the proposed method and experimental results are summarized, and the paper is concluded.

2. Related Work

Recent state-of-the-art facial attribute editing models are generally based on GANs [1] and encoder–decoder structures [13]. GANs are a generation model class that possess performance superior to conventional methods in generating realistic images. Recently, it has

been implemented successfully in the fields of facial attribute editing [2–8], image super-resolution [14], 2-D game sprite generation [15], and representation learning [16].

2.1 Generative Adversarial Networks

The basic GANs [1] receives a random vector to generate a realistic image. GANs has two parts: a generator G and a discriminator D. The generator G learns to generate a plausible image after receiving a random vector, while the discriminator D learns to distinguish the fake image created by the generator G from the real dataset. The generator G and discriminator D have opposing objectives, thereby competing against each other to achieve their goals. Through this competition, the generator G is trained to generate more natural images, while the discriminator D is trained to better distinguish fake images from real images. Because of its ability to generate samples by learning the data distribution, GANs are popular for use in tasks that aim to generate more natural images.

A conditional GAN is an extension of the traditional GAN that reflects conditional information during the training process. A model of this type is called the cGAN [17]. During the training process, the traditional GAN inputs a random vector into generator G to generate a realistic image, whereas the cGAN provides additional data to generator G and discriminator D. When such additional data are provided as a condition to generator G along with the random vector during the test process, an image matching such condition can be generated. For example, if training is conducted on the modified National Institute of Standards and Technology dataset composed of handwritten number images with class labels, then the class labels can be given as conditions to synthesize the number images corresponding to class labels [17].

2.2 Image-to-Image Translation

Facial attribute editing enables facial images to attain desired attributes. This process can be considered part of the image-to-image translation research field, which involves mapping an image from a source domain to a target domain.

Invertible cGAN (IcGAN) [18] is a combination of an encoder with a cGAN and can be used to convert an image. The encoder performs as an inverse function of the cGAN and can map an actual image to a latent code with a conditional representation. The IcGAN ensures that the image consists of the desired attributes and conditions. For example, facial attribute editing can be conducted to change the hair color, sex, condition of wearing glasses, etc.

Pix2pix [19] was first proposed as a general-purpose model for image-to-image translation. The pix2pix is an encoder–decoder type model that can convert various types of images. Although this model can modify the input image to have the desired attributes, it requires an aligned image pair between the input and output images. However, in many cases, images from different domains do not align in the output domain. For example, if an image of a horse is intended to be changed into an image of a zebra, an image of a zebra in the same place, of the same size, and in the same position as the horse, must be provided, which is practically impossible to obtain.

To address the limitations of pix2pix, CycleGAN [20] was proposed to facilitate the conversion between different domains based on the cycle consistency used in machine translation, regardless of related target domain image unavailability. An example of cycle consistency is when an English text is translated into Korean. The resulting translation of the corresponding text back into English should be the same as the original text. Similarly, if an

image is converted to an image in another domain and then converted back to the original domain, the output result should be the same as the original image. The CycleGAN is able to learn based on datasets of two different domains, despite the unavailability of paired training datasets. However, a dedicated model is required for each domain translation; thus, if the number of domains required for conversion increases, the number of required models may increase drastically.

To address the scalability issue of CycleGAN, StarGAN [21] was proposed as a method to train various domains using a single model by providing domain information when training a generator. The architecture of the StarGAN model allows for the simultaneous training of multiple similar datasets with a single model.

2.3 Facial Attribute Editing

Existing models, such as CycleGAN and StarGAN, focus on changing the desired attributes. However, this leads to certain drawbacks in several cases where attributes other than the desired attribute were unintentionally changed. Hence, several studies have been conducted on facial attribute editing to ensure selectively changing only the desired attributes to prevent unintended changes to other attributes [2–8].

The selective facial attribute editing approaches include using residual training [2], applying a spatial attention mechanism [6], setting constraints on restoration and attributes [3,4,7], and implementing masking methods [5].

GANs with residual image (referred to as ResGAN) [2] is a learning method that uses a residual image. To train the addition and removal of a single attribute, ResGAN reflects the cycle consistency loss through CycleGAN training and applies a skip connection of ResNet [22]. This process resulted in a more focused modification in the desired area. However, similar to CycleGAN, ResGAN requires generating two separate models to modify a single attribute.

2.3.1 SaGAN

GANs with spatial attention (referred to as SaGAN) [6] method, inspired by ResGAN, applies a spatial attention mechanism and enables changing only the desired attribute. However, unlike ResGAN, which uses residuals, SaGAN uses a spatial attention mechanism to train an attention mask that ensures changing only the desired regions. To achieve this, the model uses a method of simultaneously training the attribute manipulation network (AMN), which is responsible for editing the attribute, and the spatial attention network (SAN), which is responsible for localizing the attribute-specific region. However, SaGAN requires separate model training to change multiple attributes. In addition, because the attention mask is applied with unsupervised learning, it may be generated incorrectly or may not be activated at all, resulting in the original image being returned due to mistraining.

2.3.2 AttGAN

AttGAN [3] is a method of applying constraints on the restoration and attributes. Two approaches are used to achieve the aim of AttGAN, which is to suppress any undesirable changes in attributes. The first approach is to apply an attribute classification constraint to

the generated image to impose correct changes to the desired attribute only. The second approach involves applying a reconstruction loss to ensure that the output is similar to the input image when the input image is restored back to the original attribute. By applying these two methods simultaneously, AttGAN modifies the desired attributes while preventing unwanted changes in the rest. AttGAN uses an encoder–decoder structure to simultaneously learn multiple attributes using a single model, and it enables the simultaneous change of multiple attributes.

2.3.3 STGAN

STGAN [4] is a model that improves the image quality degradation due to the bottleneck layer in the AttGAN. Rather than using the attribute vectors that express the attributes of the image, the STGAN model uses a difference attribute vector, which represents the vector difference between the original attribute and the attribute to be changed to emphasize the attribute to be changed. Furthermore, the performance of the model is improved by proposing selective transfer units (STUs), which can selectively transfer the features of the encoder to the decoder according to the changed attribute. However, the AttGAN and STGAN models still face issues of unintentionally changing the background or facial details when modifying the desired attributes. For example, when the hair color of a person wearing a hat is changed, the region of the hat also becomes distorted.

The proposed method is based on STGAN. Therefore, the loss functions of STGAN are adopted, which include reconstruction, adversarial, and attribute manipulation losses.

○ Reconstruction Loss

The reconstruction loss calculates the difference between the input and output images as a loss value by restoring the image using a given attribute as the input. Similar loss functions have been previously used in various studies in related fields [3–4,6,21]. Equation (1) describes the reconstruction loss.

$$\mathcal{L}_{G_{\mathrm{id}}} = \|\boldsymbol{x} - G(\boldsymbol{x}, 0)\|_1, \tag{1}$$

the difference between the input image $\boldsymbol{x}$ and the image restored to the original by generator G is measured as a loss. Because the difference between the same attributes is obtained, the attribute difference is set to zero.

○ Adversarial Loss

Adversarial losses are commonly used in GAN-based models. The adversarial loss ensures the generation of a more realistic image by generating a fake image that is difficult to be distinguished from real images of the dataset. Therefore, and discriminator D learns to effectively distinguish real from fake images. Regarding the adversarial loss, the WGAN-GP method [23,24] was applied, as expressed in Equation (2):

$$\begin{aligned} \max_{D_{\mathrm{adv}}} \mathcal{L}_{D_{\mathrm{adv}}} &= \mathbb{E}_{\boldsymbol{x}} D_{\mathrm{adv}}(\boldsymbol{x}) - \mathbb{E}_{\hat{\boldsymbol{y}}} D_{\mathrm{adv}}(\hat{\boldsymbol{y}}) + \lambda_{\mathrm{gp}} \mathbb{E}_{\hat{\boldsymbol{x}}}[(\|\nabla_{\hat{\boldsymbol{x}}} D_{\mathrm{adv}}(\hat{\boldsymbol{x}})\|_2 - 1)^2], \\ \max_{G} \mathcal{L}_{G_{\mathrm{adv}}} &= \mathbb{E}_{\boldsymbol{x}, \boldsymbol{att}_{\mathrm{diff}}} D_{\mathrm{adv}}\big(G(\boldsymbol{x}, \boldsymbol{att}_{\mathrm{diff}})\big), \end{aligned} \tag{2}$$

here, $\hat{\boldsymbol{x}}$ denotes a sample taken from the line between actual images and generated images. $\boldsymbol{att}_{\text{diff}}$ is a difference attribute vector indicating the difference between attribute vectors. $\hat{\boldsymbol{y}}$ is the result image generated by facial attribute editing.

○ Attribute Manipulation Loss

The attribute discriminator D_{att} is trained to correctly classify the attributes based on the original attributes $\boldsymbol{att}_s$ of an input image $\boldsymbol{x}$. Equation (3) expresses the attribute manipulation loss of discriminator D, which is the sum of the binary cross-entropy loss of all attributes.

$$\mathcal{L}_{D_{\text{att}}} = -\sum_{i=1}^{n}\left[\boldsymbol{att}_s{}^{(i)} log D_{\text{att}}{}^{(i)}(\boldsymbol{x}) + (1-\boldsymbol{att}_s{}^{(i)})\log\left(1-D_{\text{att}}{}^{(i)}(\boldsymbol{x})\right)\right], \tag{3}$$

where $\boldsymbol{att}_s{}^{(i)}$ denotes the ith attribute of the image $\boldsymbol{x}$, $D_{\text{att}}{}^{(i)}(x)$ denotes the value predicted by the attribute discriminator D_{att} for the ith attribute of the image $\boldsymbol{x}$.

The main goal of attribute manipulation loss is to train the model so that image $\hat{\boldsymbol{y}}$ generated by generator G can easily trick the attribute discriminator D_{att}. Equation (4) expresses the attribute manipulation loss of generator G, which is the sum of the binary cross-entropy loss of all attributes.

$$\mathcal{L}_{G_{\text{att}}} = -\sum_{i=1}^{n}\left[\boldsymbol{att}_t{}^{(i)}\log D_{\text{att}}{}^{(i)}(\hat{\boldsymbol{y}}) + \left(1-\boldsymbol{att}_t{}^{(i)}\right)\log\left(1-D_{\text{att}}{}^{(i)}(\hat{\boldsymbol{y}})\right)\right], \tag{4}$$

where $\boldsymbol{att}_t{}^{(i)}$ denotes the ith target attribute and, $D_{\text{att}}{}^{(i)}(\hat{\boldsymbol{y}})$ denotes the value predicted by the attribute discriminator D_{att} for the ith attribute of the edited face image $\hat{\boldsymbol{y}}$.

2.3.4 Other Methods

The GAN with semantic masks (referred to as SM-GAN) [5] model uses the method of applying masks. By applying a separate semantic segmentation network to AttGAN, SM-GAN improves the distortion that occurs during the facial attribute editing of AttGAN by limiting the changeable region. However, it is necessary to train a separate semantic segmentation network for selective facial attribute editing. Moreover, the semantic segmentation mask does not correspond to the area to be edited, but to the entire specific semantic area. In other words, the implementation of the editing area limit is not sophisticated.

Recently, to improve the complex coupling relationship of STGAN, a multi-attention U-net-based GAN (referred to as MU-GAN) [7] has been proposed with the symmetric U-Net architecture and self-attention layer. In terms of applying a type of attention mechanism based on STGAN, it is similar to the proposed study. However, the proposed method is different in that it utilizes additional information by applying a spatial attention mechanism as in a semi-supervised learning method and focuses more on the suppression of image distortion.

Based on the STGAN model, the proposed method simultaneously performs training with facial attribute editing, spatial attention mechanism, and clothing segmentation with an encoder–decoder structure. Through this method, multiple attributes can be changed

simultaneously, and the regions related to the attribute to be changed are modified while preserving the clothing region. In addition, the accuracy of the attention mask is improved by using guide masks in training.

3. Proposed Method

3.1 System Overview

Fig. 1 provides an overview of the proposed method. In this study, two datasets, CelebA [25] and CelebAMask-HQ [26], were used together, and only CelebAMask-HQ had semantic segmentation masks. Semantic segmentation masks were used to generate guide masks. A guide mask is used when a generator is trained to generate attention masks and clothing masks in semi-supervised learning. The attention mask and clothing mask were used to edit only the desired attributes in the proposed method. The training mini-batch was randomly sampled from two datasets, so only some samples had meaningful guide masks. In the absence of guide masks, dummy masks were used instead. The network of the proposed method was trained using face images, attribute vectors, and guide masks. Finally, facial attribute editing was performed as the generator for the trained model.

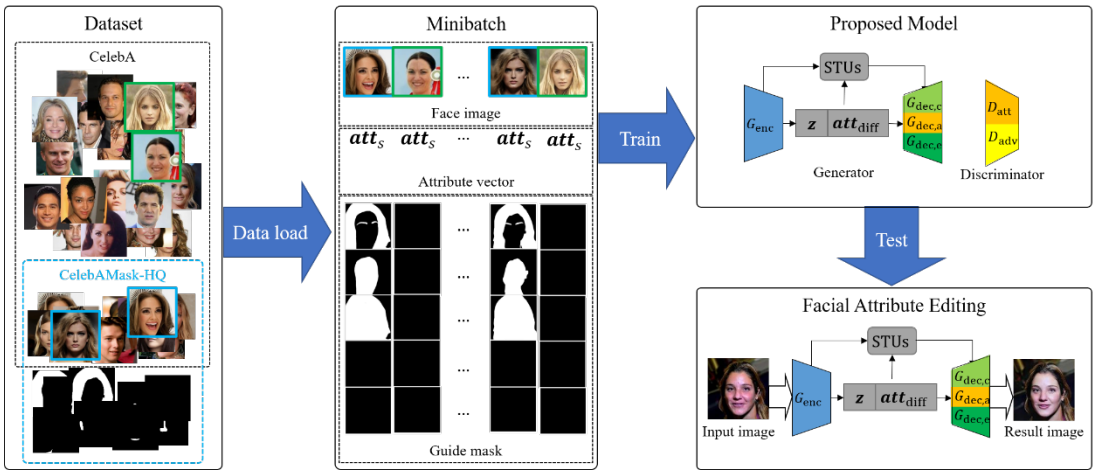


Fig. 1. Overview of the proposed method.

3.2 Semi-Supervised Attention Mask

Semi-supervised learning refers to a method of learning that uses a large amount of unlabeled data and a small amount of labeled data. In the proposed method, the two datasets were merged. The CelebA dataset consists of a large number of face images and facial attributes, and CelebAMask-HQ additionally includes a semantic segmentation mask as a subset composed of only the high-resolution images in CelebA. In the proposed method, semantic segmentation masks were used to generate the guide masks. Thus, one sample consisted of one face image, 13 face attributes, five guide masks, or five dummy masks. The implemented dummy mask is used to maintain the consistency of the tensor structure and to identify the case where there is no guide mask. In the proposed method, the guide mask generated from the CelebAMask-HQ dataset was used as the labeled data for the attention mask. Therefore, the proposed method can be considered a semi-supervised learning approach.

3.3 Guide Mask

A total of seven guide masks $\boldsymbol{m}_{\mathrm{g}}$ were used in this study. Each mask was generated by combining the semantic segmentation labels $\boldsymbol{\mu}$ of the CelebAMask-HQ dataset [26]. Equation (5) expresses the method of generating each guide mask $\boldsymbol{m}_{\mathrm{g}}$.

$$\begin{aligned}
\boldsymbol{m}_{\mathrm{hair}} &= \boldsymbol{\mu}_{\mathrm{hair}} \cup \boldsymbol{\mu}_{\mathrm{l_brow}} \cup \boldsymbol{\mu}_{\mathrm{r_brow}} - \boldsymbol{m}_{\mathrm{clothing}}, \\
\boldsymbol{m}_{\mathrm{skin}} &= \boldsymbol{\mu}_{\mathrm{skin}} \cup \boldsymbol{\mu}_{\mathrm{neck}} \cup \boldsymbol{\mu}_{\mathrm{l_ear}} \cup \boldsymbol{\mu}_{\mathrm{r_ear}} - \boldsymbol{m}_{\mathrm{clothing}}, \\
\boldsymbol{m}_{\mathrm{person}} &= \boldsymbol{m}_{\mathrm{hair}} \cup \boldsymbol{m}_{\mathrm{skin}} - \boldsymbol{m}_{\mathrm{clothing}}, \\
\boldsymbol{m}_{\mathrm{eye_g}} &= \boldsymbol{\mu}_{\mathrm{eye_g}}, \\
\boldsymbol{m}_{\mathrm{clothing}} &= \boldsymbol{\mu}_{\mathrm{cloth}} \cup \boldsymbol{\mu}_{\mathrm{hat}} \cup \boldsymbol{\mu}_{\mathrm{ear_r}} \cup \boldsymbol{\mu}_{\mathrm{neck_l}}, \\
\boldsymbol{m}_{\mathrm{unknown}} &= [[\,-255, -255, \ldots -255, -255], \ldots [-255, -255, \ldots, -255, -255]], \\
\boldsymbol{m}_{\mathrm{zero}} &= [[\,0, 0, \ldots 0, 0], \ldots [0, 0, \ldots, 0, 0]],
\end{aligned} \tag{5}$$

here, $\boldsymbol{m}_{\mathrm{hair}}$ denotes the hair and eyebrow regions, $\boldsymbol{m}_{\mathrm{skin}}$ indicates the skin region, $\boldsymbol{m}_{\mathrm{person}}$ denotes the combination of hair and skin regions, $\boldsymbol{m}_{\mathrm{eye\_g}}$ represents the glass region, $\boldsymbol{m}_{\mathrm{clothing}}$ represents the clothing (hats and clothes) and accessory (earrings and necklaces) regions excluding the glass region, and $\boldsymbol{m}_{\mathrm{zero}}$ is a mask where all values are set to zero. **Fig. 2** displays two examples of the masks defined above (excluding $\boldsymbol{m}_{\mathrm{unknown}}$ and $\boldsymbol{m}_{\mathrm{zero}}$).

The attention guide masks $\boldsymbol{m}_{\mathrm{a}}$ used for the estimated attention mask $\hat{\boldsymbol{m}}_{\mathrm{a}}$ are all the guide masks $\boldsymbol{m}_{\mathrm{g}}$, except for $\boldsymbol{m}_{\mathrm{clothing}}$. The clothing guide mask $\boldsymbol{m}_{\mathrm{clothing}}$ is used for the estimated clothing mask $\hat{\boldsymbol{m}}_{\mathrm{c}}$.

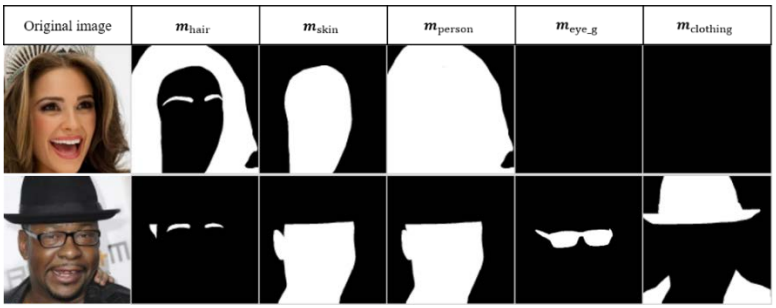


Fig. 2. Examples of each guide mask $\boldsymbol{m}_{\mathrm{g}}$

3.4 Network Architecture

The network architecture of the proposed method is depicted in **Fig. 3**. The proposed model mainly consists of generator G and discriminator D. The network structure of generator G is composed of encoder G_{enc} and decoder G_{dec}, and its main function is to convert the face input image into a face image that reflects the input attributes. However, discriminator D competes against generator G to distinguish between the real images form the dataset and the fake images generated by generator G and estimates the attributes of an input image. Generator G and discriminator D are designed to compete to improve their performance. **Tables 1** and **2** present the detailed network design of the proposed method.

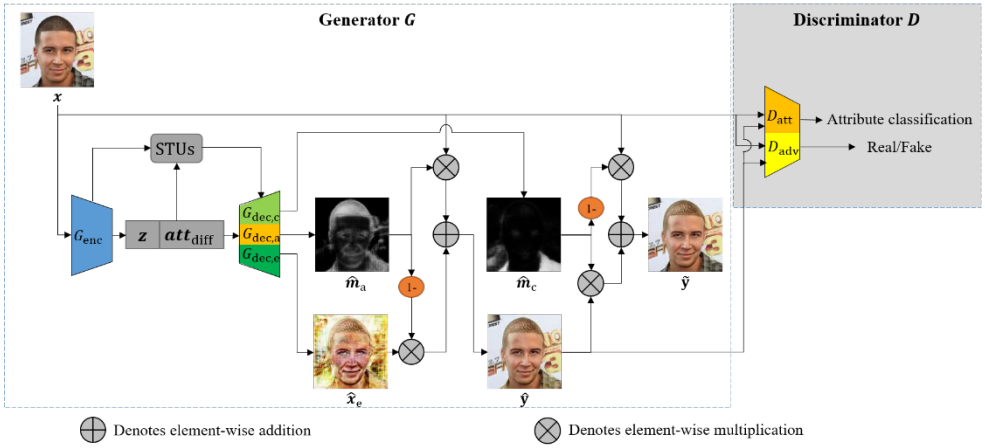


Fig. 1. Network architecture of the proposed method. Left: generator G; top right: discriminator D.

Table 1. Architecture of the generator G. Here, l is the layer number. Conv(d,k,s) is a convolution layer, DeConv(d,k,s) is a transposed convolution layer, d is output channel, k is kernel size, and s is stride. BN is batch normalization.

l	G^l_{enc}	$G^l_{\text{dec,e}}$	$G^l_{\text{dec,a}}$	$G^l_{\text{dec,c}}$
1	Conv(64,4,2), BN, Leaky ReLU	DeConv(3,4,2), Tanh	DeConv(1,4,2), Sigmoid	DeConv(1,4,2), Sigmoid
2	Conv(128,4,2), BN, Leaky ReLU	DeConv(128,4,2), BN, ReLU		
3	Conv(256,4,2), BN, Leaky ReLU	DeConv(256,4,2), BN, ReLU		
4	Conv(512,4,2), BN, Leaky ReLU	DeConv(512,4,2), BN, ReLU		
5	Conv(1024,4,2), BN, Leaky ReLU	DeConv(1024,4,2), BN, ReLU		

Table 2. Architecture of the discriminator D. Here, l is the layer number. Conv(d,k,s) is a convolution layer, DeConv(d,k,s) is a transposed convolution layer, d is output channel, k is kernel size, and s is stride. BN is batch normalization. IN is instance normalization. FC is a fully connected layer.

l	D^l_{adv}	D^l_{att}
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(512,4,2), IN, Leaky ReLU	
5	Conv(1024,4,2), IN, Leaky ReLU	
6	FC(1024), Leaky ReLU	FC(1024), Leaky ReLU
7	FC(1)	FC(13), Sigmoid

The decoder G_{dec} is divided into three branches: $G_{\text{dec,a}}$, $G_{\text{dec,e}}$, and $G_{\text{dec,c}}$. The first branch, $G_{\text{dec,a}}$, generates an estimated attention mask $\hat{\boldsymbol{m}}_{\text{a}}$. The estimated attention mask $\hat{\boldsymbol{m}}_{\text{a}}$ limits the changeable region according to the desired attribute changes in the input image. The second branch $G_{\text{dec,e}}$ synthesizes the facial attribute editing image $\hat{\boldsymbol{x}}_{\text{e}}$. The third branch $G_{\text{dec,c}}$ generates the clothing mask $\hat{\boldsymbol{m}}_{\text{c}}$. Although the clothing region is not part of the editing area, distortion may occur in the region when other attributes, such as hair color, change. Hence, the estimated clothing mask $\hat{\boldsymbol{m}}_{\text{c}}$ is generated to prevent changes in unwanted regions, preventing any distortion that may occur owing to the changes in the attributes. Because the process of estimating the clothing mask $\hat{\boldsymbol{m}}_{\text{c}}$ may unintentionally affect the training process of discriminator D, it was designed as an additional step. The aforementioned tasks share the same face domain; hence, they are trained within a single encoder–decoder structure.

Discriminator D is composed of two subnetworks, D_{adv} and D_{att}. Here, D_{adv} examines whether an input image is a real image from the dataset or a fake image generated by generator G. Meanwhile, D_{att} estimates the attributes of the input image.

3.5 Test Process

The test process generates the final edited face image $\tilde{\boldsymbol{y}}$ after inputting an image to generator G and providing the attribute to be changed. Once generator G receives input image $\boldsymbol{x}$ with n number of binary attributes $\boldsymbol{s} = [s_1, \ldots, s_n]$, encoder G_{enc} generates feature $\boldsymbol{f}$. Encoder G_{enc} is composed of five layers, and the features are extracted in each layer. Equation (6) is used in encoder G_{enc}.

$$\boldsymbol{f} = G_{\mathrm{enc}}(\boldsymbol{x}), \qquad \boldsymbol{f} = \{\boldsymbol{f}^1{}_{\mathrm{enc}}, \ldots, \boldsymbol{f}^5{}_{\mathrm{enc}}\}. \tag{6}$$

The proposed method is based on the STGAN model; hence, it uses the difference attribute vector and STU of the STGAN. The difference attribute vector represents the vector for only the changed attributes to preserve the input image information. When n number of binary attributes $\boldsymbol{t} = [t_1, \ldots, t_n]$ is provided for editing, the difference attribute vector can be obtained by using Equation (7) as follows:

$$\boldsymbol{att}_{\mathrm{diff}} = \boldsymbol{att}_t - \boldsymbol{att}_s. \tag{7}$$

The STU selectively transfers the features of the encoder G_{enc} to the features of the decoder G_{dec} according to the attributes to be changed, as shown in Equation (8):

$$(\boldsymbol{f}_t{}^l, \boldsymbol{s}^l) = G^l{}_{st}(\boldsymbol{f}^l{}_{\mathrm{enc}}, \boldsymbol{s}^{l+1}, \boldsymbol{att}_{\mathrm{diff}}), \tag{8}$$

where $G^l{}_{st}$ denotes the lth STU, $\boldsymbol{s}^l$ represents the hidden state of layer l, and $\boldsymbol{f}_t{}^l$ denotes the transformed encoder feature of the lth layer.

Decoder G_{dec} receives the encoded latent code and the output from the STUs to generate the final edited face image $\tilde{\boldsymbol{y}}$. During this process, three outputs are generated as follows: edited attribute image $\hat{\boldsymbol{x}}_{\mathrm{e}}$, attention mask $\hat{\boldsymbol{m}}_{\mathrm{a}}$ (highlighting the area to be changed), and clothing mask $\hat{\boldsymbol{m}}_{\mathrm{c}}$(highlighting the area of clothing to be preserved). The values of the attention mask $\hat{\boldsymbol{m}}_{\mathrm{a}}$ can range from 0 to 1 for each pixel; a value closer to 1 denotes a region that needs to be changed, whereas a value closer to 0 denotes a region that does not need to be changed. Then, based on the attention mask values $\hat{\boldsymbol{m}}_{\mathrm{a}}$, the edited face image $\hat{\boldsymbol{y}}$ is generated. Equation (9) expresses the process of generating the edited face image $\hat{\boldsymbol{y}}$.

$$\begin{gathered}(\hat{\boldsymbol{x}}_{\mathrm{e}}, \hat{\boldsymbol{m}}_{\mathrm{a}}, \hat{\boldsymbol{m}}_{\mathrm{c}}) = G_{\mathrm{dec}}(\boldsymbol{f}^5{}_{\mathrm{enc}}, \boldsymbol{f}_t), \\ \hat{\boldsymbol{y}} = \hat{\boldsymbol{x}}_{\mathrm{e}} \cdot \hat{\boldsymbol{m}}_{\mathrm{a}} + \boldsymbol{x} \cdot (1 - \hat{\boldsymbol{m}}_{\mathrm{a}}).\end{gathered} \tag{9}$$

In the edited face image $\hat{\boldsymbol{y}}$, there may be instances where the clothing region is not sufficiently preserved. Thus, the model is trained with clothing mask $\hat{\boldsymbol{m}}_{\mathrm{c}}$ to preserve the clothing region. The value of the clothing masks $\hat{\boldsymbol{m}}_{\mathrm{c}}$ can range from 0 to 1 for each pixel; a value closer to 1 denotes a clothing region that needs to be preserved, whereas a value closer to 0 denotes a non-clothing region. Then, based on the clothing mask values $\hat{\boldsymbol{m}}_{\mathrm{c}}$, a final

edited face image $\tilde{\boldsymbol{y}}$ is generated. Equation (10) describes the process of generating the final edited face image $\tilde{\boldsymbol{y}}$.

$$\tilde{\boldsymbol{y}} = \boldsymbol{x} \cdot \hat{\boldsymbol{m}}_{\mathrm{c}} + \hat{\boldsymbol{x}}_t \cdot (1 - \hat{\boldsymbol{m}}_{\mathrm{c}}). \tag{10}$$

In summary, as expressed in Equation (11), image $\boldsymbol{x}$ and difference attribute vector $\boldsymbol{att}_{\mathrm{diff}}$ are received as the inputs that generate the edited image $\tilde{\boldsymbol{y}}$ as the output.

$$\tilde{\boldsymbol{y}} = G(\boldsymbol{x}, \boldsymbol{att}_{\mathrm{diff}}). \tag{11}$$

3.6 Training Process

In this study, the main goals of the training process are to preserve the details of the original image by using loss functions and to generate a natural output image with the desired attributes. The proposed model uses the semi-supervised learning method via training both with and without guide masks $\boldsymbol{m}_{\mathrm{g}}$. The guide masks $\boldsymbol{m}_{\mathrm{g}}$ are used to supplement the region information for the attention mask $\hat{\boldsymbol{m}}_{\mathrm{a}}$ and clothing mask $\hat{\boldsymbol{m}}_{\mathrm{c}}$ during the training process. Here, the attention guide mask $\boldsymbol{m}_{\mathrm{a}}$ is designated for the estimated attention mask $\hat{\boldsymbol{m}}_{\mathrm{a}}$, while $\boldsymbol{m}_{\mathrm{c}}$ is designated for the estimated clothing mask $\hat{\boldsymbol{m}}_{\mathrm{c}}$. The loss functions used in the STGAN model are commonly used regardless of the presence of the guide mask $\boldsymbol{m}_{\mathrm{g}}$. In addition, the cycle reconstruction and attention identity reconstruction losses are used (Section 3.6.1).

When guide masks $\boldsymbol{m}_{\mathrm{g}}$ are present, an additional loss function is used in each attention mask $\hat{\boldsymbol{m}}_{\mathrm{a}}$ and clothing mask $\hat{\boldsymbol{m}}_{\mathrm{c}}$. The loss function of the attention mask and attention guide mask $\boldsymbol{m}_{\mathrm{a}}$ to be used (Section 3.6.2) are determined based on the attribute to be changed in the training process. The loss function of the clothing mask $\hat{\boldsymbol{m}}_{\mathrm{c}}$ is denoted as the clothing mask loss, and the guide mask is expressed as the clothing guide mask $\boldsymbol{m}_{\mathrm{c}}$.

3.6.1 Basic Losses

The loss functions applied regardless of the presence of the guide mask $\boldsymbol{m}_{\mathrm{g}}$ include reconstruction loss, adversarial loss, attribute manipulation loss, and attention identity reconstruction loss functions.

○ Dual Reconstruction Loss

Equation (12) denotes the cycle reconstruction loss, where the facial attribute editing image is generated with respect to the target attribute vector $\boldsymbol{t}$, and the image is edited to retain the original sample attribute vector $\boldsymbol{s}$. Because the image is restored back to the original sample attribute, the difference between the output and input image $\boldsymbol{x}$ is calculated.

$$\mathcal{L}_{\mathrm{dual}} = \|\boldsymbol{x} - G(G(\boldsymbol{x}, \boldsymbol{att}_{\mathrm{diff}}), -\boldsymbol{att}_{\mathrm{diff}})\|_1. \tag{12}$$

○ Attention Identity Reconstruction Loss

Attention identity reconstruction loss is used to restore original attributes during the training process to suppress excessive activation of the attention mask. To generate the restored image $\hat{\boldsymbol{y}}$ using the input image $\boldsymbol{x}_s$ and original attribute vector $\boldsymbol{s}$, all values of the attention mask $\boldsymbol{m}_{\mathrm{a}}$ need to be zero because no attribute values are changed. Although previous studies have used similar loss functions [2,25], our proposed method differs in that attention identity

reconstruction loss is used only when reconstruction is conducted. The attention identity reconstruction loss can be obtained using Equation (13) as follows:

$$\mathcal{L}_{\text{a\_id}} = \left\| \hat{\boldsymbol{m}}_{\text{a},ss} \right\|_1, \tag{13}$$

where the estimated attention mask $\hat{\boldsymbol{m}}_{\text{a},ij}$ represents the attention mask $\hat{\boldsymbol{m}}_{\text{a}}$ generated when the image $\boldsymbol{x}_i$ with an attribute vector $\boldsymbol{i}$ is edited into image $\hat{\boldsymbol{y}}$ with an attribute vector $\boldsymbol{j}$. Therefore, $\hat{\boldsymbol{m}}_{\text{a},ss}$ indicates the attention mask $\hat{\boldsymbol{m}}_{\text{a}}$ generated when a reconstructed image is generated by inputting the source attribute vector $\boldsymbol{s}$ of the original image as an attribute vector.

3.6.2 Mask Losses with Guide Mask

Mask losses using guide masks $\boldsymbol{m}_{\text{g}}$ are related to two masks: estimated attention mask $\hat{\boldsymbol{m}}_{\text{a}}$ and estimated clothing mask $\hat{\boldsymbol{m}}_{\text{c}}$. The estimated attention mask $\hat{\boldsymbol{m}}_{\text{a}}$ represents the regions to be changed in relation to the attributes within the image. Meanwhile, the estimated clothing mask $\hat{\boldsymbol{m}}_{\text{c}}$ represents the clothing and accessory regions that should not be changed. Here, the glass region is not included in the clothing mask as it is subject to editing. When training these two masks, attention guide mask $\boldsymbol{m}_{\text{a}}$ and clothing guide mask $\boldsymbol{m}_{\text{c}}$ are used for generating the estimated attention mask $\hat{\boldsymbol{m}}_{\text{a}}$ and estimated clothing mask $\hat{\boldsymbol{m}}_{\text{c}}$, respectively.

3.6.2.1 Attention Mask Losses

Considering the attention mask training process, one loss function and one attention guide mask $\boldsymbol{m}_{\text{a}}$ are applied according to the rules defined by the multiple attribute values to be changed. Because multiple attribute training is an expanded version of single attribute training, first, single attribute training is explained and then expanded to describe multiple attribute training. The training process of the estimated attention mask $\hat{\boldsymbol{m}}_{\text{a}}$ is divided into three cases. The first case is when attention guide mask $\boldsymbol{m}_{\text{a}}$ clearly highlights the exact regions to be edited, in which the target region loss is used. The second case is when attention guide mask $\boldsymbol{m}_{\text{a}}$ highlights the broad regions to be edited, in which nontarget suppression loss is used. The third case is when no appropriate attention guide mask $\boldsymbol{m}_{\text{a}}$ is available, in which no loss is used. When conducting training on each sample, the losses that are not being used are multiplied by 0 to deactivate them. At the end of this section, attention mask loss and attention guide mask selection rules according to the changing multiple attributes are explained.

① Target Region Loss

The target region loss is used when attention guide mask $\boldsymbol{m}_{\text{a}}$ overlaps sufficiently with attention mask $\hat{\boldsymbol{m}}_{\text{a}}$. The difference between attention mask $\hat{\boldsymbol{m}}_{\text{a}}$ and attention guide mask $\boldsymbol{m}_{\text{a}}$ is expressed as the loss used to reduce this difference. The target region loss is applied in both converting attribute vector $\boldsymbol{s}$ to $\boldsymbol{t}$ and vice versa. Whether using target region loss and what guide mask to use depends on what attribute is changed and whether the change is addition or removal. For example, when adding glasses, if the original image does not contain a glass region, guide mask $\boldsymbol{m}_{\text{eye\_g}}$ cannot be applied. Furthermore, when applying a baldness attribute, the hair region can be used as attention guide mask $\boldsymbol{m}_{\text{a}}$ because the hair region removal corresponds to the intention. However, when removing the baldness attribute,

the target region loss cannot be applied because an attention guide mask $\boldsymbol{m}_{\mathrm{a}}$ is unavailable to determine the hair region size. The target region loss can be obtained using Equation (14):

$$\mathcal{L}_{G_{\mathrm{target}}} = g_{\mathrm{target}}\left(\left\|\hat{\boldsymbol{m}}_{\mathrm{a},st} - \boldsymbol{m}_{\mathrm{a}}\right\|_2 + \left\|\hat{\boldsymbol{m}}_{\mathrm{a},ts} - \boldsymbol{m}_{\mathrm{a}}\right\|_2\right), \tag{14}$$

where $\hat{\boldsymbol{m}}_{\mathrm{a},st}$ is the estimated attention mask $\hat{\boldsymbol{m}}_{\mathrm{a}}$ when changing attribute vector $\mathbf{s}$ to attribute vector $\boldsymbol{t}$. In contrast, $\hat{\boldsymbol{m}}_{\mathrm{a},ts}$ is the estimated attention mask $\hat{\boldsymbol{m}}_{\mathrm{a}}$ when changing the edited target attribute vector $\boldsymbol{t}$ back to the original attribute vector $\boldsymbol{s}$. Furthermore, $\boldsymbol{m}_{\mathrm{a}}$ denotes the selected attention guide mask. The distance between the masks was measured based on the $L2$ norm distance. Finally, g_{target} indicates the gate variable with a value of 0 or 1 depending on the presence of an attention guide mask and multiple attribute conditions during the training process.

② Nontarget Suppression Loss

The nontarget suppression loss is used when attention mask $\boldsymbol{m}_{\mathrm{a}}$ does not sufficiently overlap with attention mask $\hat{\boldsymbol{m}}_{\mathrm{a}}$ but can limit the editing area. Compared to the target region loss, the nontarget suppression loss is a less strict version of the limit for when information of attention guide mask $\boldsymbol{m}_{\mathrm{a}}$ is insufficient. There is no mask in CelebAMask-HQ that directly responds to areas such as bangs, beards, gender, and youth. However, there are masks in the areas that can be changed owing to each attribute. For example, a mustache is more likely to appear within the skin area. Using these assumptions, changes in unrelated areas can be defined as unnecessary changes. To measure such unnecessary changes, the area that should not be changed is defined using attention guide masks $\boldsymbol{m}_{\mathrm{a}}$. Subsequently, the average of the estimated attention mask values $\hat{\boldsymbol{m}}_{\mathrm{a}}$ activated in the immutable area is measured as the nontarget suppression loss, which can suppress unnecessary changes. The nontarget suppression loss can be obtained using Equation (15):

$$\begin{aligned}\mathcal{L}_{\mathrm{nontarget}} = g_{\mathrm{nontarget}}(&\mathbb{E}\max((\hat{\boldsymbol{m}}_{\mathrm{a},st} - \boldsymbol{m}_{\mathrm{a}}), \boldsymbol{m}_{\mathrm{zero}}) \\ &+\mathbb{E}\max((\hat{\boldsymbol{m}}_{\mathrm{a},ts} - \boldsymbol{m}_{a}), \boldsymbol{m}_{\mathrm{zero}})),\end{aligned} \tag{15}$$

where $g_{\mathrm{nontarget}}$ corresponds to the gate variable with a value of 0 or 1 depending on the presence of an attention guide mask and multiple attribute conditions during the training process.

③ None

When an appropriate attention guide mask $\boldsymbol{m}_{\mathrm{a}}$ is unavailable to train attention mask $\hat{\boldsymbol{m}}_{\mathrm{a}}$, the attention mask loss is not used. For example, the mouth region changes upon opening and closing of the mouth, but the regions of the face also change in the image after editing. Therefore, semantic segmentation labels $\boldsymbol{\mu}$ are not provided for the mouth region. The same applies to the changes in the jaw region. Furthermore, when removing the baldness attribute, semantic segmentation labels $\boldsymbol{\mu}$ are not provided for the hair region; thus, no attention mask loss is used.

○ Selection of Attention Guide Mask $\boldsymbol{m}_{\mathrm{a}}$ and Attention Mask Loss Suitable for Multiple Attributes

The proposed method is focused on editing multiple attributes at once. Consequently, attention guide mask $\boldsymbol{m}_{\mathrm{a}}$ and attention mask loss should be selected according to the

multiple attributes to be changed. **Tables 3** and **4** list the rules for selecting a proper attention guide mask $\boldsymbol{m}_a$. First, all editing region flags φ are initialized to "False." Subsequently, based on the multiple attributes to be edited, the editing region flags φ that meet the conditions listed in **Table 3** are activated. Finally, attention guide masks $\boldsymbol{m}_a$ are selected based on the selection criteria listed in **Table 4**. Through this process, the attention guide masks $\boldsymbol{m}_a$ are selected to cover the regions to be edited.

Table 3. Edit region flag according to the attribute to be edited

Attributes to be edited (+/-/±)*	Edit region flag
Black_Hair±, Blond_Hair±, Brown_Hair±, Bald+	φ_{hair}=True
Bushy_Eyebrows±, Eyeglasses+, Male±, Mustache±, No_Beard±, Pale_skin±, Young±,	φ_{skin}=True
Bangs±	φ_{hair}=True, φ_{skin}=True
Eyeglasses-	$\varphi_{\text{eye\_g}}$=True
Mouth_Slightly_Open±, Bald-	φ_{unknown}=True
Reconstruction**	φ_{zero} =True

* +: add attribute, -: remove attribute, ±: add or remove attribute
**reconstruction: when there is no change in the attribute due to the same attribute input.

Table 4. Attention guide mask by decision condition

Attention guide mask decision condition	Selected attention guide mask
if(φ_{unknown})	$\boldsymbol{m}_{\text{unknown}}$*
if(φ_{hair} and not (φ_{skin} or $\varphi_{\text{eye\_g}}$ or φ_{unknown}))	$\boldsymbol{m}_{\text{hair}}$
if(φ_{skin} and not (φ_{hair} or φ_{unknown}))	$\boldsymbol{m}_{\text{skin}}$
if($\varphi_{\text{eye\_g}}$ and not (φ_{hair} or φ_{skin} or φ_{unknown}))	$\boldsymbol{m}_{\text{eye\_g}}$
if(φ_{hair} and (φ_{skin} or $\varphi_{\text{eye\_g}}$) and not φ_{unknown})	$\boldsymbol{m}_{\text{person}}$
if(not (φ_{hair} or $\varphi_{\text{eye\_g}}$ or φ_{skin} or φ_{unknown}))	$\boldsymbol{m}_{\text{zero}}$

* $\boldsymbol{m}_{\text{unknown}}$ is a dummy mask for implementation when there is no appropriate attention guide mask $\boldsymbol{m}_a$. When this dummy mask is activated, the loss function of the attention guide mask $\boldsymbol{m}_a$ is deactivated, hence there is no negative impact on the resulting image.

Table 5. Attention mask loss flag setting depending on the attribute to be edited

Target mask	Attributes to be edited (+/-/±)*	Attention mask loss flag
Attention mask	Bald+, Black_Hair±, Blond_Hair±, Brown_Hair±, Eyeglasses-, Pale_Skin±	φ_{target} = True
	Bangs±, Bushy_Eyebrows±, Eyeglasses+, Male±, Mustache±, No_Beard±, Young±	$\varphi_{\text{nontarget}}$ = True
	Mouth_Slightly_Open±, Bald-	φ_{none} = True

* +: add attribute, -: remove attribute, ±: add or remove attribute

The attention mask loss is selected using a similar process. **Tables 5** and **6** present the rules for selecting the attention mask loss. First, all attention mask loss flags φ are initialized to "False." Subsequently, based on the attribute to be changed, attention mask loss flags φ that meet the conditions reported in **Table 5** are activated. Finally, the attention mask losses to be used are selected based on the selection criteria listed in **Table 6**. Through this process, only the appropriate loss among the attention mask losses is applied.

Table 6. Selected attention guide mask loss by decision condition

Attention guide loss decision condition	Selected attention guide loss
if(φ_{none})	None
if($\varphi_{\text{nontarget}}$ and not φ_{none})	$\mathcal{L}_{\text{nontarget}}$
if(φ_{target} and not ($\varphi_{\text{nontarget}}$ or φ_{none}))	$\mathcal{L}_{\text{target}}$

3.6.2.2 Clothing Mask Loss

○ Clothing Mask Loss

In this study, clothing items, such as hats and shirts, are not subject to editing. However, when conducting other attribute changes, such as hair color changes, the clothing regions may become distorted. To address this issue, a clothing mask is applied after clothing segmentation. The clothing mask loss can be obtained using Equation (16):

$$\mathcal{L}_{\text{clothing}} = g_{\text{clothing}}(\left\|\hat{\boldsymbol{m}}_{\text{c},st} - \boldsymbol{m}_{\text{c}}\right\|_2 + \left\|\hat{\boldsymbol{m}}_{\text{c},ts} - \boldsymbol{m}_{\text{c}}\right\|_2), \tag{16}$$

where g_{clothing} is a gate variable that has a value of 0 or 1 depending on the presence or absence of a clothing guide mask during training.

3.6.3 Objective Function

The equations for final objective functions of G and D are as follows.

$$\min_D \mathcal{L}_D = -\mathcal{L}_{D_{\text{adv}}} + \mathcal{L}_{D_{\text{att}}}, \tag{17}$$

$$\begin{aligned}\min_G \mathcal{L}_G = &-\mathcal{L}_{G_{\text{adv}}} + \lambda_{\text{rec}}\mathcal{L}_{\text{rec}} + \lambda_{G_{\text{att}}}\mathcal{L}_{G_{\text{att}}} + \lambda_{\text{dual}}\mathcal{L}_{\text{dual}} + \lambda_{\text{a\_id}}\mathcal{L}_{\text{a\_id}} \\ &+\lambda_{\text{target}}\mathcal{L}_{\text{target}} + \lambda_{\text{nontarget}}\mathcal{L}_{\text{nontarget}} + \lambda_{\text{clothing}}\mathcal{L}_{\text{clothing}},\end{aligned} \tag{18}$$

here, λ denotes the weight of each loss function.

4. Experimental Results

4.1 Experimental Environment

The settings of the proposed method were as follows. Most of the weight settings were the same as those for STGAN. The ADAM optimizer was set to $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The learning rate was 2×10^{-4} up to 100 epochs, and 2×10^{-5} after 100 epochs. Each model trained a total of 200 epochs. The loss function weights related to the spatial attention mechanism were searched using a random search. The final loss weights were set to $\lambda_{\text{rec}} = 0.05, \lambda_{\text{dual}} = 4.69, \lambda_{\text{a\_id}} = 2.25, \lambda_{\text{target}} = 3.75, \lambda_{\text{nontarget}} = 0.09,$ and $\lambda_{\text{clothing}} = 4.45, \lambda_{\text{gp}} = 10, \lambda_{G_{\text{att}}} = 10$.

The dataset used was CelebA [27], which consists of a total of 202,559 images of 10,177 celebrities. In addition, 40 binary attributes and five landmarks are provided for each image. This dataset was split into 182,000 training images and 19,962 test images. There were 13 attributes used for learning (Bald, Bangs, Black_Hair, Blond_Hair, Brown_Hair, Bushy_Eyebrows, Eyeglasses, Male, Mouth_Slightly_Open, Mustache, No_Beard, Pale_Skin, and Young). CelebA did not provide a semantic segmentation mask. Instead, CelebAMask-HQ [26] was used for utilize the semantic segmentation mask. CelebAMask-

HQ consists of 30,000 high-resolution CelebA images. In addition, it provides 19 types of semantic segmentation masks (hair, skin, eyes, hat, etc.) for the face region of each image.

In the experiment, the official codes of AttGAN v1 [28] and of STGAN [29] were used. The graphics card used was NVIDIA GeForce RTX 2080 Ti.

Quantitative evaluation metrics were used for two purposes. The first purpose was to measure whether the facial attributes were edited as intended. For this, we used attribute generation accuracy (AGA) [4]. The second purpose was to measure the degree of image preservation in the background area to distinguish unintended editing. For this, the peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM) [30], which are generally used for image quality evaluation, were used. The reason for limiting these to the background area was that the background was not subject to change. Thus, if the background was preserved, it could be estimated that there was little unintended distortion. To obtain only the background area, when there is a corresponding CelebAMask-HQ sample in the test set, the human area is masked.

4.2 Hyperparameter Optimization

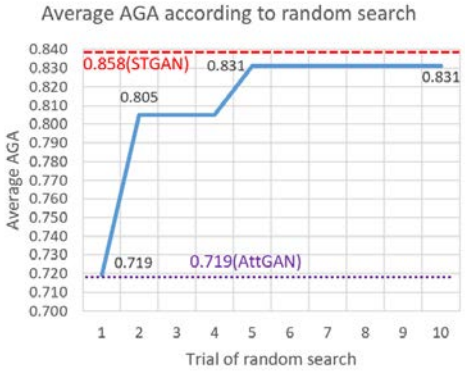
In the proposed method, various loss functions are used to accurately edit only the part related to the attribute to be changed. Therefore, it is necessary to properly determine the weights of the loss functions. We used the random search method [31,32], which is one of the baselines of the hyperparameter optimization method, to determine the weights of the loss functions. Random search is a simple method and is more efficient than a grid search because it does not cause overlaps during search [31], and it can be easily applied even when learning multiple models in parallel.

Fig. 3 displays the quantitative evaluation metrics of the random search process. In a random search, one criterion is required to determine the final model. In this study, a search was performed based on the average AGA. In addition, in the proposed method, we searched between 0.0 and 5.0 for each loss weight by referring to the initial manual search result. From the result shown in **Fig. 3(a)**, we can observe that the average AGA gradually increases as the number of random search trials increases also. However, in **Fig. 3(b)** and **(c)**, notice that the average PSNR and SSIM gradually decrease. This phenomenon indicates that the goal of changing the face attributes clearly and goal of preserving images are in a tradeoff relationship. Because the proposed method is based on the average AGA, it is expected to be higher if additional trials are performed. However, PSNR and SSIM are expected to be lower owing to the tradeoff relationship.

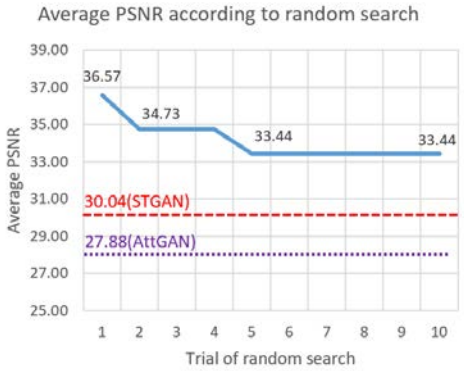
In this study, the model of the second trial was evaluated as the most satisfactory by visually comparing the three models selected in the random search process. This result shows that the average AGA is slightly lower than that of STGAN, while the average PSNR and average SSIM are higher (**Fig. 4**).

4.3 Comparative Evaluation

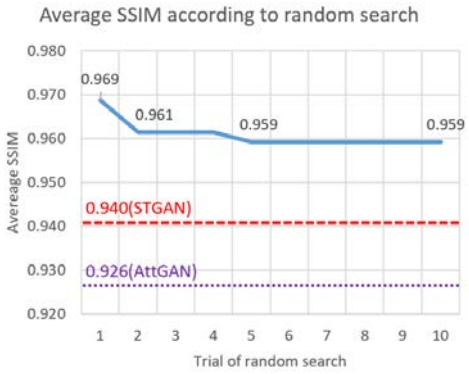
In the comparative evaluation, the selected model was compared to state-of-the-art methods. We present first the visual and then the quantitative results.



(a) Average AGA



(b) Average PSNR



(c) Average SSIM

Fig. 4. Changes in quantitative evaluation results according to random search for hyperparameter optimization. The horizontal axis represents the number of times for which a random search is performed. The vertical axis represents each evaluation scale. For AGA, PSNR and SSIM, larger values indicate better quality.

One drawback of existing methods is the color distortion in unintended areas that occurs when the color is changed. **Fig. 5** shows three samples to which two color-changing attributes (i.e., Blond_Hair and Pale_Skin) were applied. When using existing methods, it can be observed that color distortion occurs in three areas. The first distortion is the color distortion of the background area when using existing methods. For example, yellow color was added when changing to blonde hair, and white color was added when changing to a pale face, during which first and third samples have yellow added to the background color unintentionally. In addition, in the case of the red square area of the first sample, notice that the dark part of the text is brightened owing to color distortion. In the second sample, white was added to the red color of the background area, resulting in a brighter color. The second distortion is the distortion of skin tone and facial details. The skin tone should not change when changing the hair to blond attribute. However, existing methods distort skin tone. Distortion can be observed by looking at the eyes and mouth of the images changed using existing methods. Finally, color distortion of the clothing area can be observed. In the third sample, we can observe that a yellow color was added to the hat area when using existing methods to change the hair to blonde. Observing the images generated by the proposed method, as shown in **Fig. 5**, such distortions were clearly suppressed.

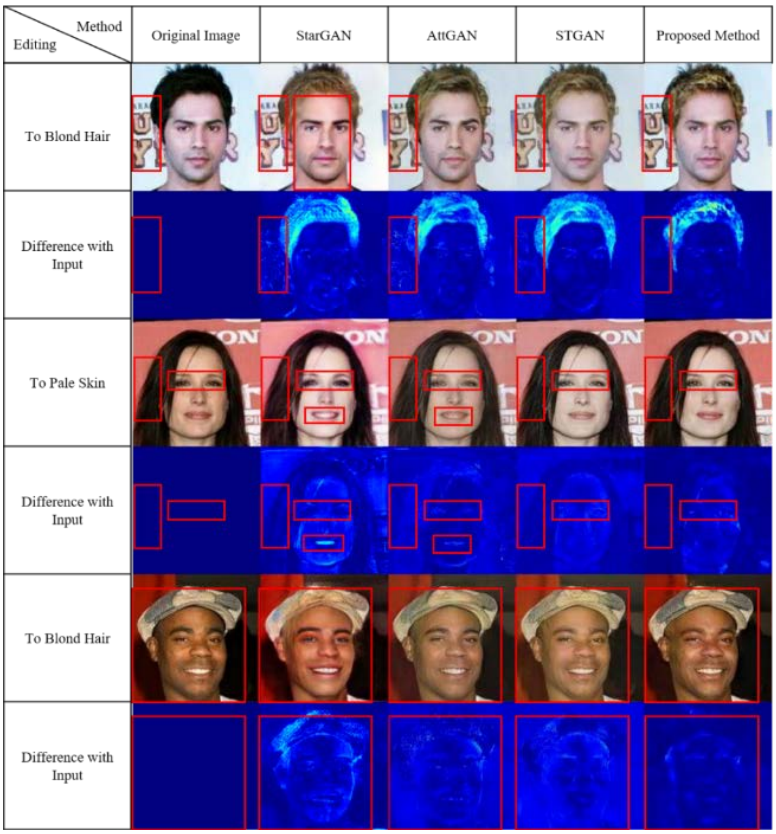


Fig. 5. Comparison of background and detail changes when changing colors

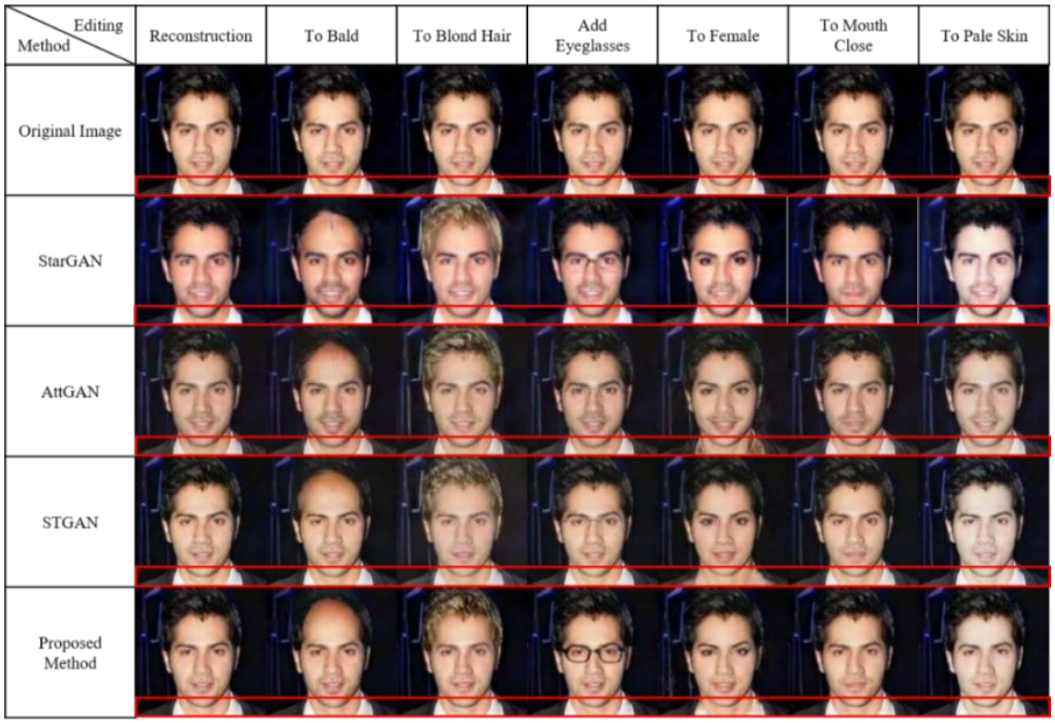


Fig. 6. Comparison of the results of editing face attributes by each attribute

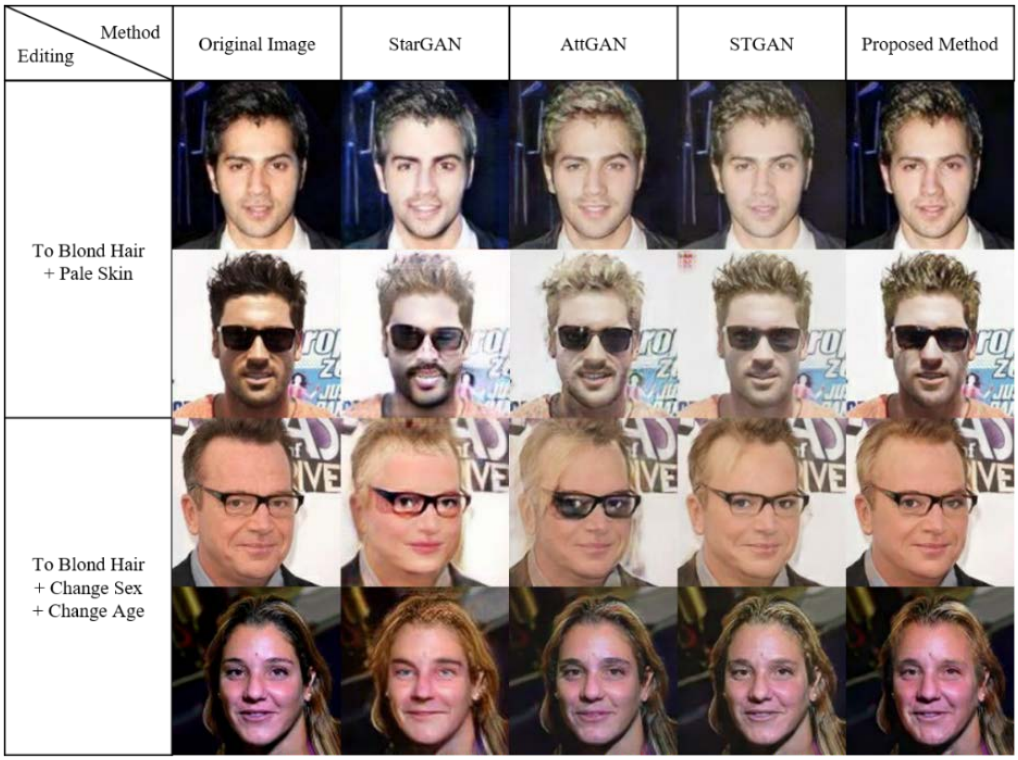


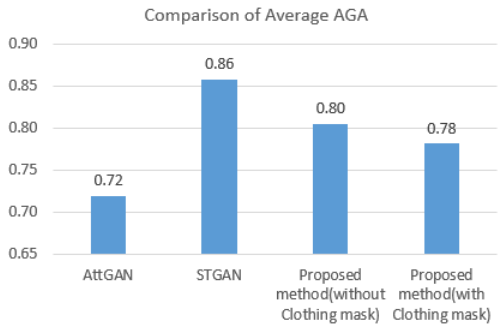
Fig. 7. Comparison of editing results of multiple attributes

Similar problems arise with other attributes. **Fig. 6** shows the changes in various attributes. When using the existing methods, color distortion of the clothing can be observed in the images, including the reconstructed image. In addition, in the case of hair change to blonde, color distortion of the background and details are noticeable. However, the proposed method demonstrates less distortion compared to the existing methods. In addition, the intended attribute changes are appropriately completed.

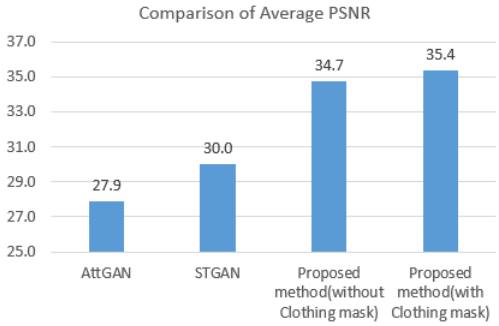
Fig. 7 shows a comparison of the results of multiple-attribute edits. The results of two combinations of attribute changes with two samples each are shown. In the first sample with existing methods, the distortion of the details of the background color, clothing, and face are obvious. Similar results were observed in the second sample. StarGAN yields a significant distortion of the mouth and background. In the case of AttGAN, the texture near the hair is distorted, and the background details are distorted. STGAN produced artifacts in the upper-left area, and the sunglasses were unintentionally brightened. In the third sample, StarGAN appears unnatural with excessive changes. The glass area of AttGAN was darkened, and blurry blond hair was added, resulting in color distortion of the clothing. Moreover, notice that the skin color was unintentionally changed to a lighter tone. The image achieved by the STGAN is less distorted than that achieved by the AttGAN, but shows some similar distortion patterns. In the last sample, existing methods did not intend to change the skin tone; nonetheless, yellow was added. It can be visually confirmed that the proposed method achieved the desired attribute change while effectively suppressing distortion.

Compared to existing methods, the proposed method suppresses unintended changes effectively. In addition, the intended attribute change was limited to the relevant area. However, the average AGA values tend to decrease as a tradeoff. **Fig. 8** compares the results

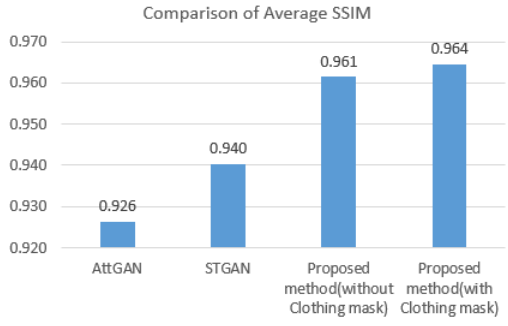
of existing methods and the proposed method using three quantitative evaluation metrics. The proposed method was evaluated by dividing the process in two steps: before and after using the clothing mask. The proposed method achieved a higher average AGA than that of AttGAN but lower than that of STGAN (**Fig. 8(a)**). In addition, the average AGA decreased slightly when a clothing mask was used. The proposed method obtained higher average PSNR and SSIM compared to the two existing methods (**Fig. 8(b)** and **Fig. 8(c)**). When the clothing mask was used, we can observe that the average PSNR and SSIM were further improved. There seems to be a tradeoff between these metrics. However, regarding visual quality, it seems that changing only the intended attribute accurately is more consistent with the intended attribute, despite a decrease in the average AGA.



(a) Average AGA



(b) Average PSNR



(c) Average SSIM

Fig. 8. Comparison of quantitative evaluation results

5. Conclusion

In this study, based on STGAN, we proposed a model to edit the intended area more accurately by adding a spatial attention method using semi-supervised learning when editing facial attributes. In order to change only the desired attributes, guide masks were defined to utilize additional area information, and guide masks and loss functions suitable for the attributes to be learned were defined. To properly set the various loss weights of the proposed method, a random search, which is a hyperparameter optimization method, was used. The final hyperparameter setting was decided using quantitative evaluation metrics and visual evaluation.

Compared with the existing state-of-the-art methods, AttGAN and STGAN, we concluded after visual examination that the proposed method could suppress background distortion, detail distortion, and clothing area distortion. In addition, it was confirmed that the image

was preserved through the improvement of the average PSNR and SSIM in the quantitative evaluation. Although the average AGA decreased somewhat due to the trade-off relationship, visual inspection confirmed that a sufficiently edited result was obtained.

A limitation of this study is that it is difficult to change the domain or add new attributes, as it would be necessary to define rules for the guide mask and loss function according to the attributes to be learned. Further research should be conducted to improve the editing of attributes pertaining to hair and clothing.

References

[1] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” *Advances in Neural Information Processing Systems*, pp. 1-9, 2014. Article (CrossRef Link)

[2] W. Shen and R. Liu, “Learning residual images for face attribute manipulation,” in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1225-1233, Jul. 2017. Article (CrossRef Link)

[3] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, “AttGAN: facial attribute editing by only changing what you want,” *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5464-5478, Nov. 2019. Article (CrossRef Link)

[4] M. Liu, Y. Ding, M. Xia, X. Liu, E. Ding, W. Zuo, and S. Wen, “STGAN: a unified selective transfer network for arbitrary image attribute editing,” in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 3668-3677, Jun. 2019. Article (CrossRef Link)

[5] P. Chen, Q. Xiao, J. Xu, X. Dong, and L. Sun, “Facial attribute editing using semantic segmentation,” in *Proc. of 2019 Int. Conf. on High Performance Big Data and Intelligent Systems (HPBD&IS)*, pp. 97-103, May 2019. Article (CrossRef Link)

[6] G. Zhang, M. Kan, S. Shan, and X. Chen, “Generative adversarial network with spatial attention for facial attribute editing,” in *Proc. of the European Conf. on Computer Vision (ECCV)*, pp. 422-437, Oct. 2018. Article (CrossRef Link)

[7] K. Zhang, Y. Su, X. Guo, L. Qi, and Z. Zhao, “MU-GAN: Facial Attribute Editing Based on Multi-Attention Mechanism,” *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 9, pp. 1614-1626, Sep. 2021. Article (CrossRef Link)

[8] X. Zheng, Y. Guo, H. Huang, Y. Li, and R. He, “A survey to deep facial attribute analysis,” *Int. Journal of Computer Vision*, vol. 128, pp. 2002-2034, Mar. 2020. Article (CrossRef Link)

[9] G. A. Khan, J. Hu, T. Li, B. Diallo, and H. Wang, "Multi-view data clustering via non-negative matrix factorization with manifold regularization," *Int. J. Mach. Learn. & Cyber*, pp. 1-13, Mar. 2021. Article (CrossRef Link)

[10] B. Diallo, J. Hu, T. Li, G. A. Khan, and A. S. Hussein, "Multi-view document clustering based on geometrical similarity measurement," *Int. J. Mach. Learn. & Cyber*, pp. 1-13, Mar. 2021. Article (CrossRef Link)

[11] G. A. Khan, J. Hu, T. Li, B. Diallo, and Y. Zhao, "Multi-view low rank sparse representation method for three-way clustering," *Int. J. Mach. Learn. & Cyber*, pp. 1-21, Aug. 2021. Article (CrossRef Link)

[12] B. Diallo, J. Hu, T. Li, and G. A. Khan, "Deep Embedding Clustering Based on Contractive Autoencoder," *Neurocomputing*, vol. 433, pp. 96-107, Jan. 2021. Article (CrossRef Link)

[13] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, “Image segmentation using deep learning: a survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. Article (CrossRef Link)

[14] Z. Wei, H. Bai, and Y. Zhao, “Stage-GAN with semantic maps for large-scale image super-resolution,” *KSII Transactions on Internet and Information Systems*, vol. 13, no. 8, pp. 3942-3961, Aug. 2019. Article (CrossRef Link)

[15] S. Hong, S. Kim, and S. Kang, "Game sprite generator using a multi discriminator GAN," *KSII Transactions on Internet and Information Systems*, vol. 13, no. 8, pp. 4255-4269, Aug. 2019. Article (CrossRef Link)
[16] C. Hu, X. Wu, and Z. Shu, "Bagging deep convolutional autoencoders trained with a mixture of real data and GAN-generated data," *KSII Transactions on Internet and Information Systems*, vol. 13, no. 11, pp. 5427-5445, Nov. 2019. Article (CrossRef Link)
[17] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, pp. 1-7, Nov. 2014. Article (CrossRef Link)
[18] G. Perarnau, J. V. D. Weijer, B. Raducanu, and J. M. Álvarez, "Invertible conditional GANs for image editing," in *Proc. of NIPS 2016 Workshop on Adversarial Training*, pp. 1-9, Nov. 2016. Article (CrossRef Link)
[19] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 5967-5976, Jul. 2017. Article (CrossRef Link)
[20] J. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. of the IEEE Int. Conf. on Computer Vision*, pp. 2242-2251, Oct. 2017. Article (CrossRef Link)
[21] Y. Choi, M. Choi, M. Kim, J. Ha, S. Kim, and J. Choo, "StarGAN: unified generative adversarial networks for multi-domain image-to-image translation," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 8789-8797, Jun. 2018. Article (CrossRef Link)
[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 770-778, Jun. 2016. Article (CrossRef Link)
[23] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. of the 34th Int. Conf. on Machine Learning*, vol. 70, pp. 214-223, 2017. Article (CrossRef Link)
[24] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, A. Courville, "Improved training of Wasserstein GANs," *Advances in Neural Information Processing Systems*, pp. 5767-5777, Dec. 2017. Article (CrossRef Link)
[25] X. Chen, C. Xu, X. Yang, and D. Tao, "Attention-GAN for object transfiguration in wild images," in *Proc. of the European Conf. on Computer Vision (ECCV)*, pp. 167-184, Oct. 2018. Article (CrossRef Link)
[26] C. Lee, Z. Liu, L. Wu, and P. Luo, "MaskGAN: towards diverse and interactive facial image manipulation," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, pp. 5548-5557, Jun. 2020. Article (CrossRef Link)
[27] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. of the IEEE Int. Conf. on Computer Vision*, pp. 3730-3738, Dec. 2015. Article (CrossRef Link)
[28] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, TensorFlow implementation of AttGAN: Facial Attribute Editing by Only Changing What You Want, 2019, [Online]. Available: https://github.com/LynnHo/AttGAN-Tensorflow/tree/v1
[29] M. Liu, Y. Ding, M. Xia, X. Liu, E. Ding, W. Zuo, and S. Wen, Tensorflow implementation of STGAN: A Unified Selective Transfer Network for Arbitrary Image Attribute Editing, 2019, [Online]. Available: https://github.com/csmliu/STGAN
[30] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600-612, Apr. 2004. Article (CrossRef Link)
[31] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, pp. 281-305, Feb. 2012. Article (CrossRef Link)
[32] C. A. Floudas and P. M. Pardalos, *Encyclopedia of optimization*, Boston, MA, USA: Springer, 2009. Article (CrossRef Link)

Hyeon Seok Yang received his B.S. degree in the Department of Electronics and Information Engineering from Yeungnam University, Korea, in 2010. He received the M.S. degrees in the Department of Computer Science & Engineering from Hanyang University, Korea, in 2012. He received the PhD. degree in the Department of Computer Science & Engineering from Hanyang University, Korea, in 2020. He is currently working at DeepBio from 2021. His research interests include computer vision, pattern recognition, and deep learning.
Email : scbwc@hanmail.net

Jeong Hoon Han received his B.S. degree in the Department of Computer Science and Engineering from Hallym University, Korea, in 2016. He received the PhD. degree in the Department of Computer Science & Engineering from Hanyang University, Korea, in 2021. He is currently working at SEMES from 2021. His research interests include computer vision, pattern recognition.
Email : bghan@visionlab.or.kr

Young Shik Moon received the B.S. and M.S. degrees in Electronics Engineering from Seoul National University and Korea Advanced Institute of Science and Technology, Korea, in 1980 and 1982, respectively, and PhD. degree in Electrical and Computer Engineering from the University of California at Irvine, CA, in 1990. From 1982 to 1985, he had been a researcher at the Electronics and Telecommunication Research Institute, Daejon, Korea. In 1992, he joined the Department of Computer Science and Engineering at Hanyang University, Korea, as an Assistant Professor, and is currently a Professor. From 2021, he is serving as the Vice President of Hanyang Cyber University, Korea.
Email : ysmoon@hanyang.ac.kr