Talk-to-Edit: Fine-Grained Facial Editing via Dialog Supplementary File

Yuming Jiang[1*] Ziqi Huang[1*] Xingang Pan[2] Chen Change Loy[1] Ziwei Liu[1✉]
[1]S-Lab, Nanyang Technological University [2]The Chinese University of Hong Kong
{yuming002, hu0007qi, ccloy, ziwei.liu}@ntu.edu.sg px117@ie.cuhk.edu.hk

In this supplementary file, we will explain the detailed annotation definition of CelebA-Dialog Dataset in Section A. Then we will introduce implementation details in Section B. In Section C, we will give more detailed explanations on experiments, including evaluation dataset, evaluation metrics and implementation details on comparison methods. Then we provide more visual results in Section D. Finally, we will discuss failure cases in Section E.

A. CelebA-Dialog Dataset Annotations

For each image, fine-grained attribute annotations and textual descriptions are provided. In Table A1 - A5, we give detailed definitions of fine-grained attribute annotations. With each fine-grained attribute label, we also provide an example image and its corresponding textual description.

Table A1: **Annotation Definition and Examples of *Bangs* Attribute.**

Attribute Degree	Fine-Grained Definition	Examples	
0	without bangs, full forehead exposed		*The lady has no bangs.*
1	very short bangs, 80% forehead exposed		*She has very short bangs covering her forehead.*
2	short bangs, 60% forehead exposed		*The man has short bangs that cover a small portion of the forehead.*
3	medium bangs, 40% forehead exposed		*The woman has bangs of medium length.*
4	long bangs, 20% forehead exposed		*The guy has long bangs.*
5	extremly long bangs, all forehead covered		*The woman has bangs that cover the eyebrows.*

*Equal contribution.

Table A2: **Annotation Definition and Examples of *Eyeglasses* Attribute.**

Attribute Degree	Fine-Grained Definition	Examples	
0	no eyeglasses		*The man doesn't wear eyeglasses.*
1	eyeglasses with a very thin metal frame or no frame.		*He wears a pair of rimless eyeglasses.*
2	eyeglasses with a thicker metal frame or thinner plastic frame.		*His eyeglasses have a thin frame.*
3	eyeglasses with a thick frame or plastic frame		*The man wears eyeglasses of a thick frame.*
4	sunglasses with a thin frame		*The lady wears a pair of sunglasses with a thin frame.*
5	sunglasses with a thick frame		*He wears sunglasses that have a thick frame.*

Table A3: **Annotation Definition and Examples of *Beard* Attribute.**

Attribute Degree	Fine-Grained Definition	Examples	
0	no beard		*There is no beard on his face.*
1	with a shaved beard, very short in length		*The man's face is covered with the short pointed beard.*
2	with a beard that hasn't been shaved for a while		*He has a short beard.*
3	with a deliberate beard of medium length		*His face is covered with beard of medium length.*
4	with a long but tiny beard		*He has a long but tiny beard.*
5	with a very bushy, long and untidy beard		*He has a very long beard.*

Table A4: **Annotation Definition and Examples of *Smiling* Attribute.**

Attribute Degree	Fine-Grained Definition	Examples	
0	no smile on face		*The woman looks serious.*
1	smile without teeth exposed		*She has a tight-lipped smile on her face.*
2	smile with some teeth exposed		*He smiles with the corners of the mouth curved up and some teeth exposed.*
3	laughing with the whole row of teeth exposed		*She has a beaming face.*
4	laughing with mouth moderately open		*There is a big smile on her face.*
5	exaggerated laughing with mouth widely open		*The woman smiles with the mouth widely open.*

Table A5: **Annotation Definition and Examples of *Young* Attribute.**

Attribute Degree	Fine-Grained Definition	Examples	
0	under 15 years old with childish face		*The person in the picture is under 15 years old.*
1	15-30 years old, adolescent		*He is at the age of adolescent.*
2	30-40 years old, mature youth		*The woman is in the thirties.*
3	40-50 years old, middle-aged		*She looks like a middle age one.*
4	50-60 years old		*He is at the age of his fifties.*
5	over 60 years old		*The man is very old.*

B. Implementation Details

B.1. User Request Understanding

The language encoder E has three components: 1) a learnable 300-D word embedding; 2) a two-layer LSTM with cell size of 1024; 3) fully-connected layers following the LSTM to generate the editing encoding $\boldsymbol{e}_r$. The learning rate is set as 10^{-3}, the batch size is 2048, and the Adam optimizer [4] is adopted.

Commonly, users' editing requests could be roughly classified into three major types: **1)** Describe the attribute and specify the target degree, *e.g.*, *Let's try extremely long bangs that cover the entire forehead.* **2)** Describe the attribute of interest and indicate the relative degree of change, *e.g.*, *The bangs can be slightly longer.* **3)** Describe the attribute and only the editing direction without specifying the degree of change, *e.g.*, *Let's make the bangs longer.* Since the types of facial editing requests are relatively fixed, we use template-based text generation methods to form a pool of editing requests. The request pool is used to train the language encoder. We prepare more than 300 request templates with diverse sentence patterns. A pool of synonymous words is used to enrich the user request templates. We generate 10,000 user requests in total. For each generated request, we provide their corresponding hard labels to train the language encoder E.

The editing encoding $\boldsymbol{e}_r$ generated by the language encoder E is implemented as hard labels containing the following information: (1) request type, (2) the attribute of interest, (3) the editing direction, and (4) the change of degree. In practice, the same user request could be interpreted differently depending on the dialog context. For example, simply saying *"Yes"* has different meanings under different scenarios. If the system makes a suggestion *"Do you want to make the bangs longer?"*, by replying *"Yes"*, the user means to make the bangs longer. However, if the system asks if the desired effect is achieved in the previous round, *"Yes"* means the editing is satisfactory in this context. Therefore, multiple language encoders are needed to parse the user request under different dialog context. During training, the weights of word embedding and LSTM are shared across different language encoders. The current system feedback decides which language encoder would be used.

We track the dialog-based editing system using a finite-state machine. The editing system is in one of the four states at any moment: **1)** *start*, that is, the first round of dialog, **2)** *edit*, where the system performs editing in the current round of dialog, **3)** *no edit*, where the system does not edit the image and wait for further instructions from the user. and **3)** *end*, where the system ends the conversation upon the user's request.

B.2. Semantic Field

The training of semantic field requires the following pretrained models: fine-grained attribute predictor P, face recognition model $Face$, StyleGAN generator G and discriminator D. The fine-grained attribute predictor P is pretrained on CelebA-Dialog dataset using our fine-grained attribute labels with a multi-class cross-entropy loss. StyleGAN G and its corresponding discriminator D are trained on CelebA dataset [6] and FFHQ dataset [3] for 128×128 and 1024×1024 facial images respectively. As for the $Face$ Model, we use the off-the-shelf ArcFace model [1] trained on LFW dataset [2, 5].

Since the pretrained StyleGAN has the mode collapse problem, during the training of semantic field, we need to sample the training latent codes such that all fine-grained attribute classes are more balancedly distributed. The mapping network of semantic field F is composed of 8 fully-connected (FC) layers with dimension 512. Except for the last FC layer, each FC layer is followed by a leaky ReLU with slope 0.2. The learning rate for training the semantic field is 10^{-4}, batch size is set as 32, and Adam optimizer [4] is adopted.

We also provide editing results on W+ sapce. When editing on W+ space, to enforce the field vector to be a valid vector that would not make the edited latent code fall into the outlier region of pretrained StyleGAN latent space, we adopt a regularization method proposed by Pan *et al*. [7]. The latent code is updated as follows:

$$\begin{aligned} \boldsymbol{z}' &= \boldsymbol{z} + \alpha(M(\boldsymbol{f}_z) - M(\mathbf{0})) \\ &= \boldsymbol{z} + \alpha(M(F(\boldsymbol{z})) - M(\mathbf{0})), \end{aligned} \tag{1}$$

where $M(\cdot)$ denotes the mapping network of StyleGAN and $F(\cdot)$ denotes the mapping network of the semantic field.

Besides, we found that the last few layers of latent codes of W+ space control the low-level features of a facial images, such as color, brightness, illuminations and etc. During facial editing, we need to keep these factors fixed. Therefore, when updating latent codes using Eq. (1), we only update first k layers of latent codes. We empirically set k as 8 for 128×128 images and 10 for 1024×1024 images.

B.3. System Feedback

After editing an image, the *Talk* module will provide a feedback, which belongs to one of the following categories: 1) checking whether the attribute degree is satisfying, in order to achieve fine-grained editing desired by the user. For example,

after the user requests to make the bangs longer, the system could give the following feedback, *e.g.*, *"Are the bangs now of the length you like?"*. If in the previous round the user agrees on to edit an attribute suggested by system but does not specify the editing direction, then the system feedback will always be checking with the user about the attribute degree. 2) providing further editing suggestions, *e.g.*, *"Do you want to try manipulating the age?"* In order to let the user fully explore possible manipulation options, the system tends not to suggest editing an attribute that has been edited before. If there exist a larger number of attributes not edited by user yet, then there is a higher probability for the system to make a suggestion, and 3) asking for user instructions , *e.g.*, *"Ok, what's next?"*.

We sample a sentence from a pool of templates of the chosen feedback category, and randomly replace phrases using a predefined pool of synonyms to extend the language richness. We observe that this simple design can provide meaningful feedback to some extent.

C. Further Explanations on Experimental Details

C.1. Evaluation Dataset

The latent code used for evaluation is formed by sampling latent codes from StyleGAN pretrained on CelebA datasets [6]. Though the StyleGAN has demonstrated its powerful generative ability in facial image generation, some synthesized images are still of low quality. Thus, we need to manually filter the bad images with artifacts out. For the evaluation dataset of the eyeglasses attribute, latent codes whose corresponding images with degree above 0 are selected, as we observe that for all methods (including baselines) editing images with degree 0 would often make the latent code fall into out-of-distribution regions (corresponding images become artifacts). To avoid the error introduced by the aforementioned issue, we only use latent codes with attribute degree above 0. When constructing the evaluation dataset of the beard attribute, we adopt the same strategy so that images with females are excluded (No females would have beard attribute degree larger than 0).

C.2. Evaluation Metrics

We employ Identity Preservation Metrics and Attribute Preservation Metrics to evaluate the identity and attribute preservation respectively. Here we explain these two metrics in detail.

Identity Preservation Metrics. We use the off-the-shelf face model *FaceNet* [8] to extract features for images before and after editing. Then we compute the euclidean distance between features of the edited facial images and the feature of the original facial image. The identity preservation metrics is expressed as follows:

$$\text{IdentityPreservation} = \frac{1}{N}\sum_{i=1}^{N} \left\| FaceNet(I_i) - FaceNet(I_0) \right\|_2 , \tag{2}$$

where I_0 is the original image, I_i are edited images, and N is the total number of edited images.

Attribute Preservation Metrics. We retrain a attribute predictor P' (different from the one we use for training), and use the retrained predictor to output cross entropy score. The attribute preservation metrics is defined as follows:

$$\text{AttributePreservation} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1,j\neq m}^{k}\sum_{c=0}^{C} y_{j,c} log(p'_{j,c}), \tag{3}$$

where N is the total number of edited images, k is the number of attributes, m is the index of the attribute being edited, $p'_{j,c}$ is the softmax output of predictor P', $y_{j,c}$ is the binary indicator with respect to the target class and it is obtained by feeding the original image to the attribute predictor.

C.3. Implementation Details on Comparison Methods

InterfaceGAN. InterfaceGAN [9] is a latent space based method. The continuous editing is achieved by moving the latent code along a straight line, *i.e.*, adding the a same vector to the original latent code. The direction used for changing the attribute degree is obtained by computing the normal vector of the binary classification SVM boundary. This direction is fixed throughout the editing. We first train binary attribute predictors to classify the generated images. Then the corresponding latent codes are used to train the binary SVM.

Multiclass SVM. We further propose an extended version of InterfaceGAN as one of the baseline methods, named Multiclass SVM. Instead of the binary classification SVM, we train multiple SVM boundaries for fine-grained labels. More specifically, for each pair of neighbouring classes, a classification SVM would be trained. Thus, for one attribute, there are five SVM

boundaries in total. During the editing, directions will be switched according to current states. The attribute predictor used for the classification of generated images is the same as the one we use for predictor loss.

Enjoy Your Editing. Enjoy your editing [10] learns a mapping network to generate identity-specific directions for each initial latent codes. The identity-specific directions keep same during editing for one image. We reimplement the method, train the mapping network with the original design and same hyper-parameters are adopted. To achieve more attribute degrees, we use larger step-sizes than the original setting, *i.e.* $\varepsilon > 1.0$.

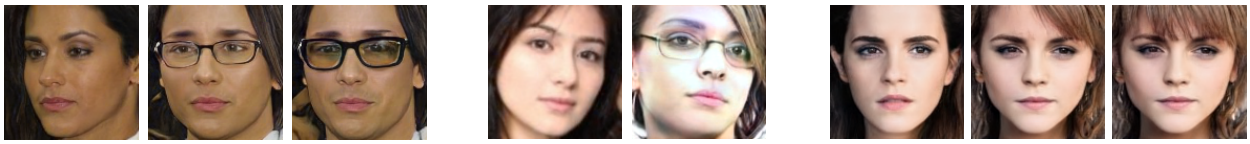
D. More Qualitative Results

In this section, we will provide more high-resolution image editing results in Fig. A2, A3, and Fig. A4. We also provide more real image editing results in Fig. A5 and more qualitative comparisons with baselines in Fig. A6 - Fig. A10.

E. Failure Cases Discussion

Here, we take the eyeglasses attribute as an example to illustrate the failure case of synthetic image editing. As shown in Fig. A1 (a), identity loss could be observed in some cases, and this issue is severer on female images. The problem may attribute to the dataset bias and the mode collapse issue of the pretrained GAN. For example, the CelebA dataset [6] has only a small number of females with eyeglasses. Thus, females with eyeglasses are only a minority in the image distribution of the pretrained GAN. In this case, given a randomly sampled female without eyeglasses as the initial image, it is sometimes difficult to wear a pair of eyeglasses for her in a well-disentangled manner. Another issue is the artifacts problem shown in Fig. A1 (b). For some latent code, it is difficult to change the attribute from degree 0 to degree 1. After many latent code updating iterations, the latent code falls into the outlier region of the latent space so that the corresponding image would bear artifacts. Our proposed semantic field may not perfectly model the non-linearity property for this attribute.

As for editing real images, it is more prone to change the identities. As shown in Fig. A1 (c), adding bangs would change the face shape. This is because that GAN-inversion, as an ill-posed problem, may introduce an additional gap between the inverted latent code and the original latent space. This could potentially be addressed by adopting more advanced GAN-inversion techniques that better keep the latent codes within the latent domain.



(a) Identity Loss (b) Artifacts (c) Real Cases

Figure A1: **Failure Case Discussion.**

References

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

[2] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007. 4

[3] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, pages 4401–4410, 2019. 4

[4] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 4

[5] Gary B. Huang Erik Learned-Miller. Labeled faces in the wild: Updates and new reporting procedures. Technical Report UM-CS-2014-003, University of Massachusetts, Amherst, May 2014. 4

[6] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, pages 3730–3738, 2015. 4, 5, 6

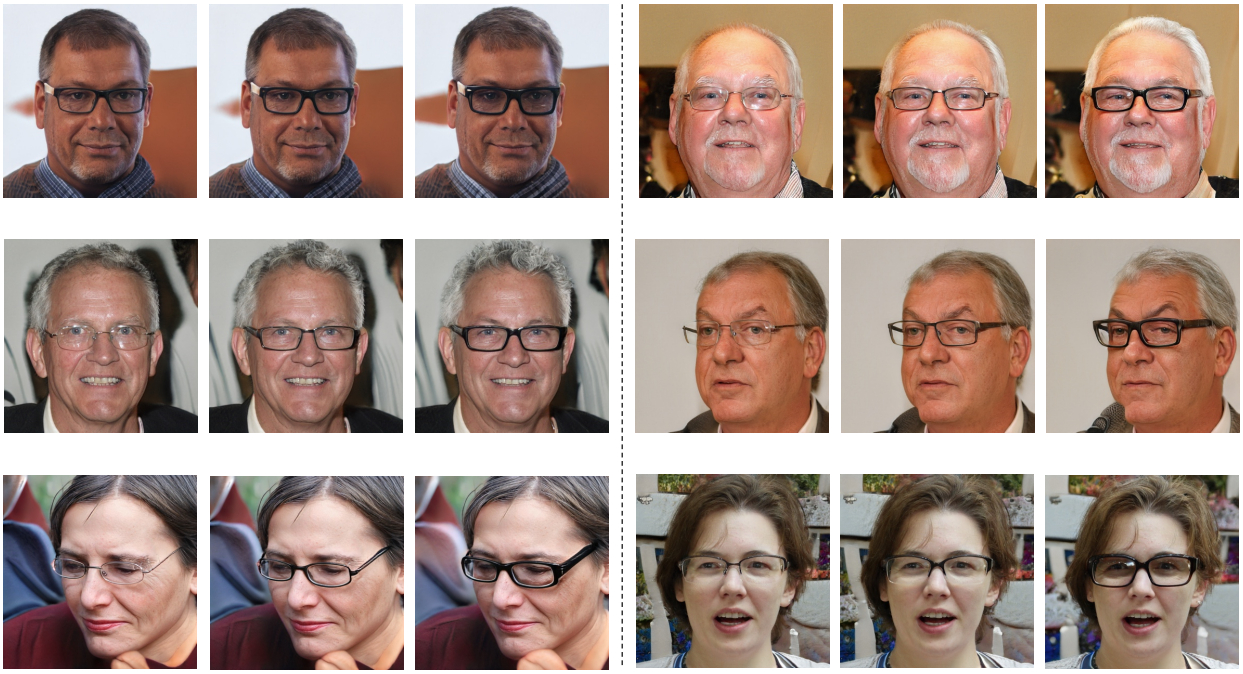
[7] Xingang Pan, Bo Dai, Ziwei Liu, Chen Change Loy, and Ping Luo. Do 2d gans know 3d shape? unsupervised 3d shape reconstruction from 2d image gans. In *ICLR*, 2021. 4

[8] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, pages 815–823, 2015. 5

[9] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *CVPR*, pages 9243–9252, 2020. 5

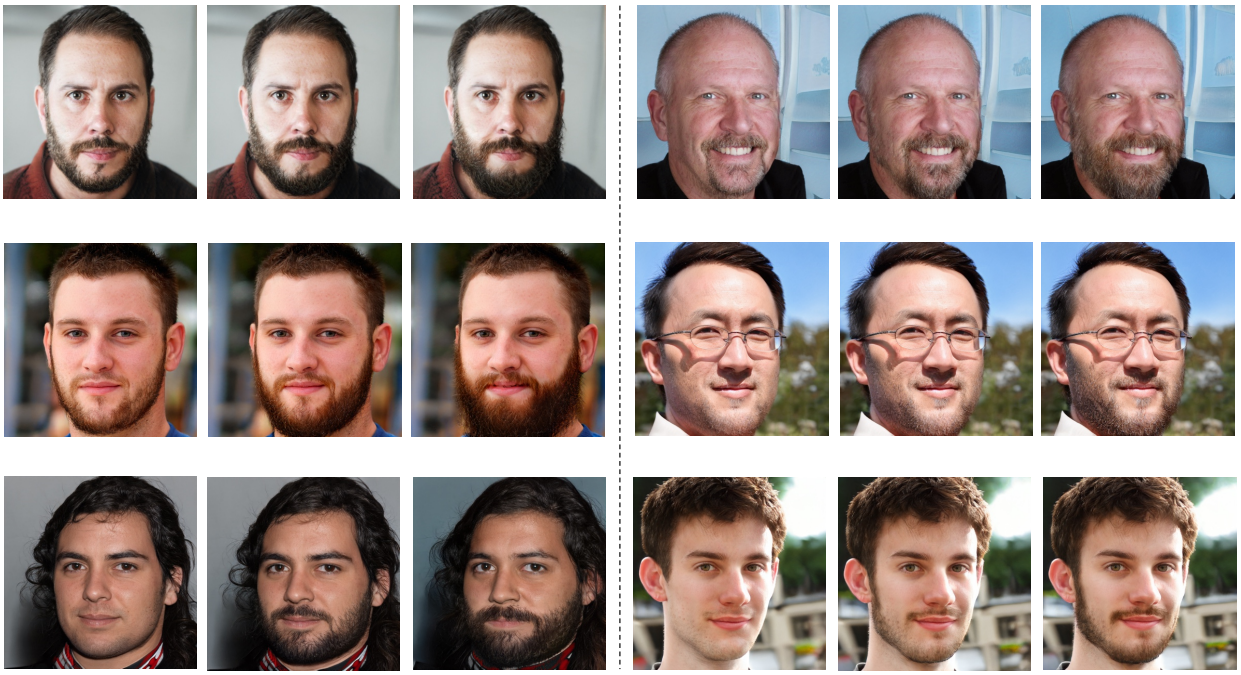
[10] Peiye Zhuang, Oluwasanmi Koyejo, and Alexander G Schwing. Enjoy your editing: Controllable gans for image editing via latent space navigation. In *ICLR*, 2021. 6

(a) Bangs



(b) Eyeglasses

Figure A2: **High-Resolution Image Editing.**



(c) Beard

(d) Smiling

Figure A3: **High-Resolution Image Editing.**

(e) Young

Figure A4: **High-Resolution Image Editing.**

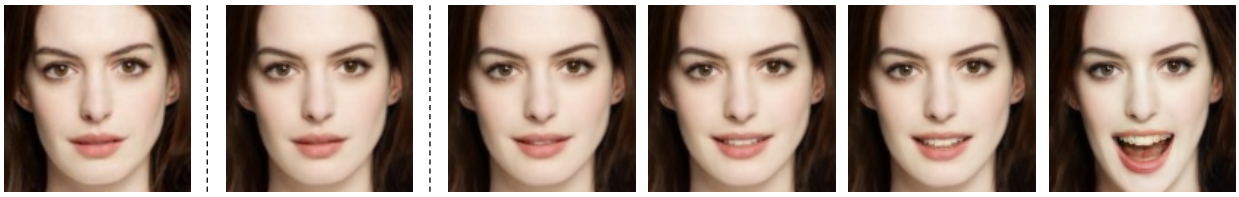


Figure A5: **Real Image Editing.**

Figure A6: **Qualitative Comparison on *Bangs* Attribute.**

Figure A7: **Qualitative Comparison on *Beard* Attribute.**

Figure A8: **Qualitative Comparison on *Eyeglasses* Attribute.**

Figure A9: **Qualitative Comparison on *Smiling* Attribute.**

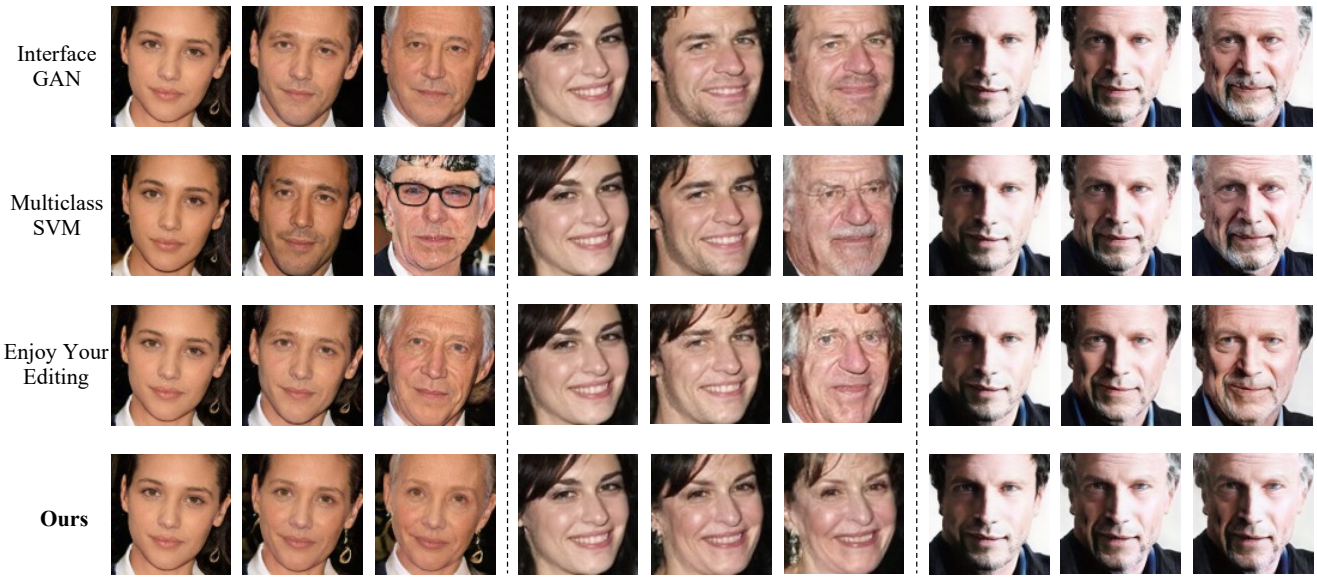


Figure A10: **Qualitative Comparison on *Young* Attribute.**