# Supplementary Material: Image-to-image Translation via Hierarchical Style Disentanglement

## A. Module Architecture

The architectural details of HiSD are shown in Figure 1. The mapper module ($\mathbf{M}$) consists of an MLP. The tag $i$ and attribute $j$ are used to index before the first layer and middle layer, respectively. The extractor module ($\mathbf{F}$) consists of five downsampling blocks, of which inherit pre-activation residual units (ResBlock) [2]. The tag $i$ is used to index before the last layer. The encoder module ($\mathbf{E}$) consists of two downsampling blocks while the generator module ($\mathbf{G}$) consists of two upsampling blocks. We use the Instance Normalization (IN) [13] in these two shared modules. The translator module ($\mathbf{T}$) consists of eight immediate blocks with Adaptive Instance Normalization (AdaIN) [3] and decreased channel dimension. The tag-relevant style code is injected into all AdaIN layers, providing scaling and shifting vectors through a linear layer. The tag $i$ is used to index before the first layer. The discriminator module ($\mathbf{D}$) uses the same architecture as $\mathbf{F}$ but uses both the tag $i$ and the attribute $j$ to index before the last layer. For all residual units, We use the Leaky ReLU (LReLU) [8] as the activation function. The Average Pooling and Nearest Neighbor Upsampling are used to resample the feature maps.

## B. Implementation Details

The batch size is 8 and the model is trained for 200K iterations. The images of CelebA-HQ are resized to 128×128. The training time is around 40 hours on a single GTX 1080Ti GPU with our implementation in PyTorch [11]. To stabilize the training, we adopt the hinge version of adversarial loss [10] with R1-regularization [9] using $\gamma = 1$. We use the Adam [6] optimizer with $\beta_1 = 0$ and $\beta_2 = 0.99$. The learning rate is $0.0001$ except for the mapper, of which the learning rate is $0.000001$ [5]. We use the historical average version [4] of the intermediate modules for test where the update weight is 0.001. We initialize the weights of all modules using He initialization [1] and set all biases to zero, except for the biases associated with the scaling vectors of AdaIN that are set to one. For hyper-parameters, we easily set $\lambda_{rec} = 1$ and $\lambda_{sty} = 1$ in all experiments.

## C. Comparison without Cherry-picking

We provide some additional qualitative results without cherry-picking of baselines and our method for the latent-guided multi-style task in Figures 2, 3, and 4. The results are completely random without manual selection. Besides the limitations we mentioned in the paper, the baselines (*i.e.* SDIT and StarGANv2) are observed to suffer from mode-collapse. Furthermore, the satisfying diversity of our generated results demonstrates the effectiveness of HiSD.

## D. Interpolation of Tag-relevant Styles

We show the interpolation results by interpolating between the extracted tag-relevant style codes from two different reference images (with the same attribute or not) in Figures 5, 6, and 7. The interpolation is smooth, which implies that the space of each tag-relevant style is continuous. The continuous tag-relevant style space allows the translations to manipulate images with a novel tag-relevant style which is not seen by the framework during training.

## E. Visualization of Tag-relevant Styles

We further explore the style space learned by the extractor by using t-SNE to visualize the extracted tag-relevant styles in a two-dimensional space. As shown in Figure 8, for all tags, images with the same tag-specific attribute are grouped together in the style space. Notably, the attribute is not inputted into the extractor in our method. For each tag, there is a main direction and various secondary directions between different attributes. InterFaceGAN [12] takes advantage of the main direction to manipulate attributes with unsupervised GANs [5, 4]. However, it cannot guarantee the disentanglement, especially for unnecessary global and identity manipulations. The images at the edge of the style space are always the most obvious examples for a specific attribute, while the images in the middle space are always confusing ones. More interestingly, for tag 'Hair color', the tag-relevant styles of images with attribute 'brown' are extracted to be a middle state between 'blond' and 'black'.

To prove the generalization ability of HiSD, we show some examples on other datasets [7, 5] and attributes in Figure 9.

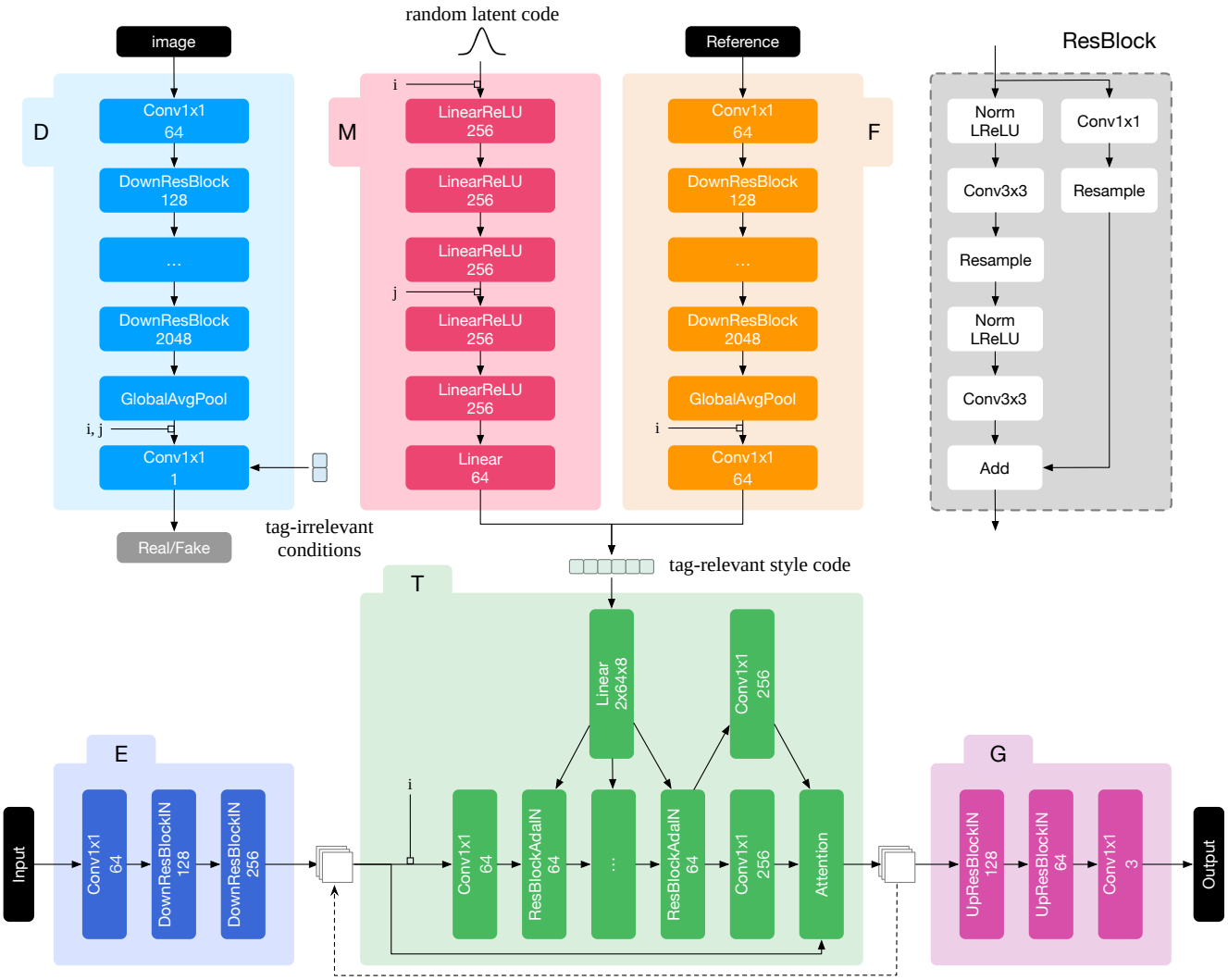


Figure 1: Architectural details of HiSD. To manipulate the input image, we first encode the image into its feature by $\mathbf{E}$. Then, the feature is manipulated by a single or multiple T. The manipulation is guided by the tag-relevant style code which can be either generated by $\mathbf{M}$ or extracted by $\mathbf{F}$. Finally, the output image is generated by $\mathbf{G}$. $\mathbf{D}$ is used to determine whether a image, given tag and attribute, is real or not. The details of ResBlocks (*i.e.* DownResBlock, DownResBlockIN, ResBlockAdaIN, UpResBlockIN) are shown at the upper right corner.

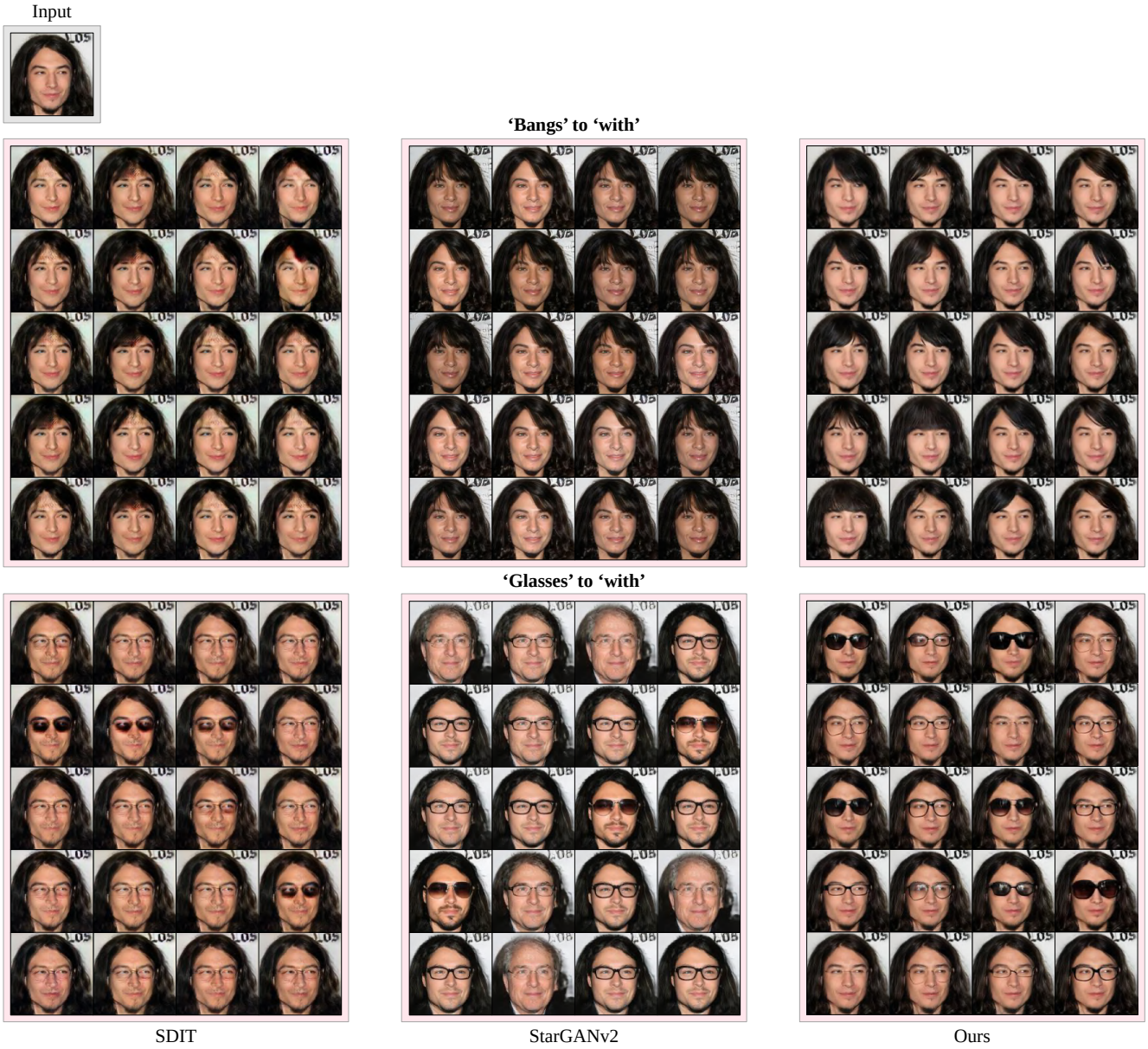


Figure 2: Additional qualitative results without cherry-picking of the baselines and our method for the latent-guided multi-style task. We respectively manipulate the input image to attribute 'with' for tag 'Bangs' and 'Glasses' by using 20 random latent codes, which are drawn from Gaussian distribution, to generate diverse outputs.


Figure 3: More additional qualitative results without cherry-picking for the latent-guided multi-style task.

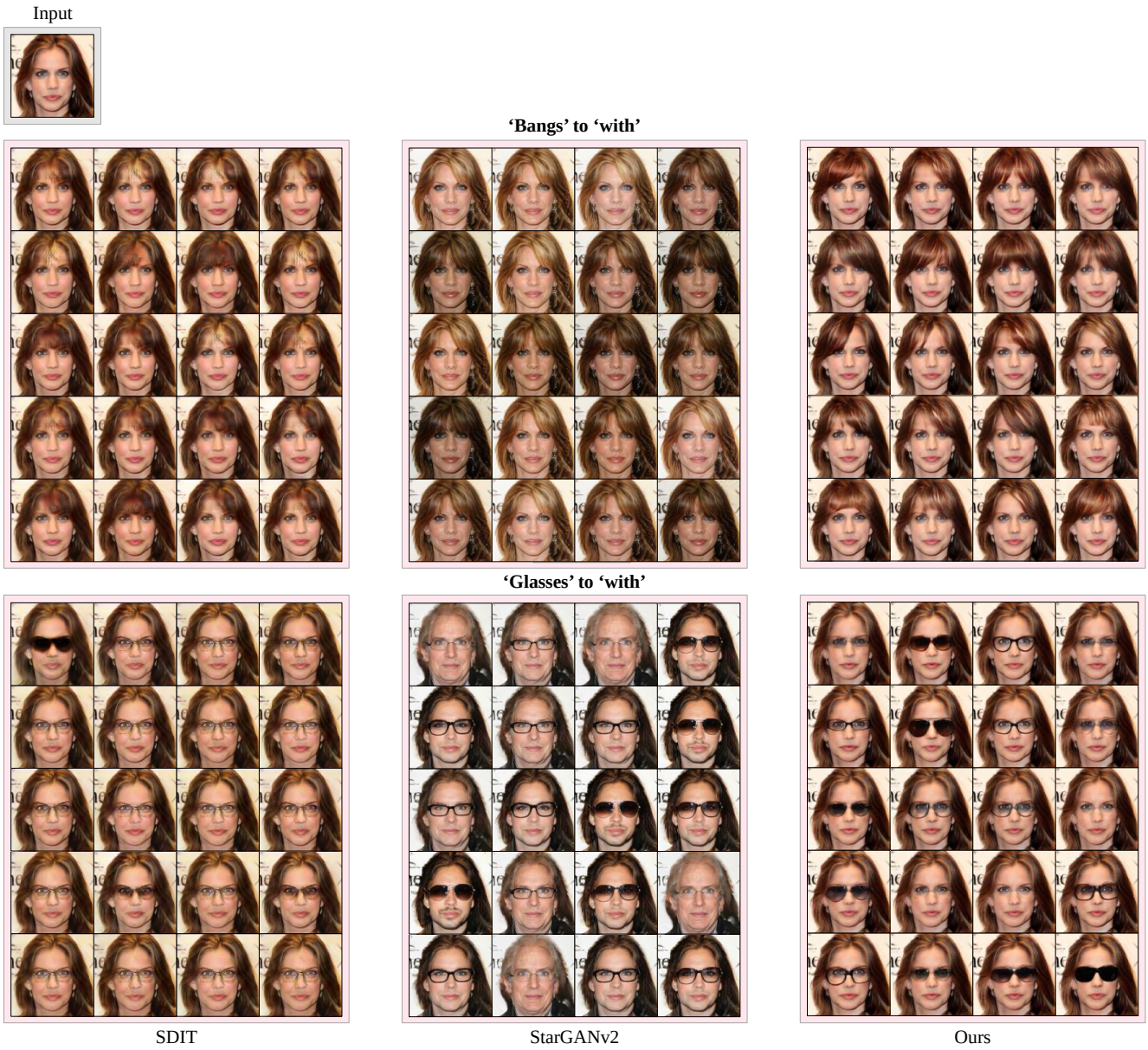


Figure 4: More additional qualitative results without cherry-picking for the latent-guided multi-style task.


Figure 5: Interpolation results between two extracted tag-relevant styles for different tags. We use the linear interpolation between the style codes extracted from two different reference images to observe the continuous manipulations of the output images.

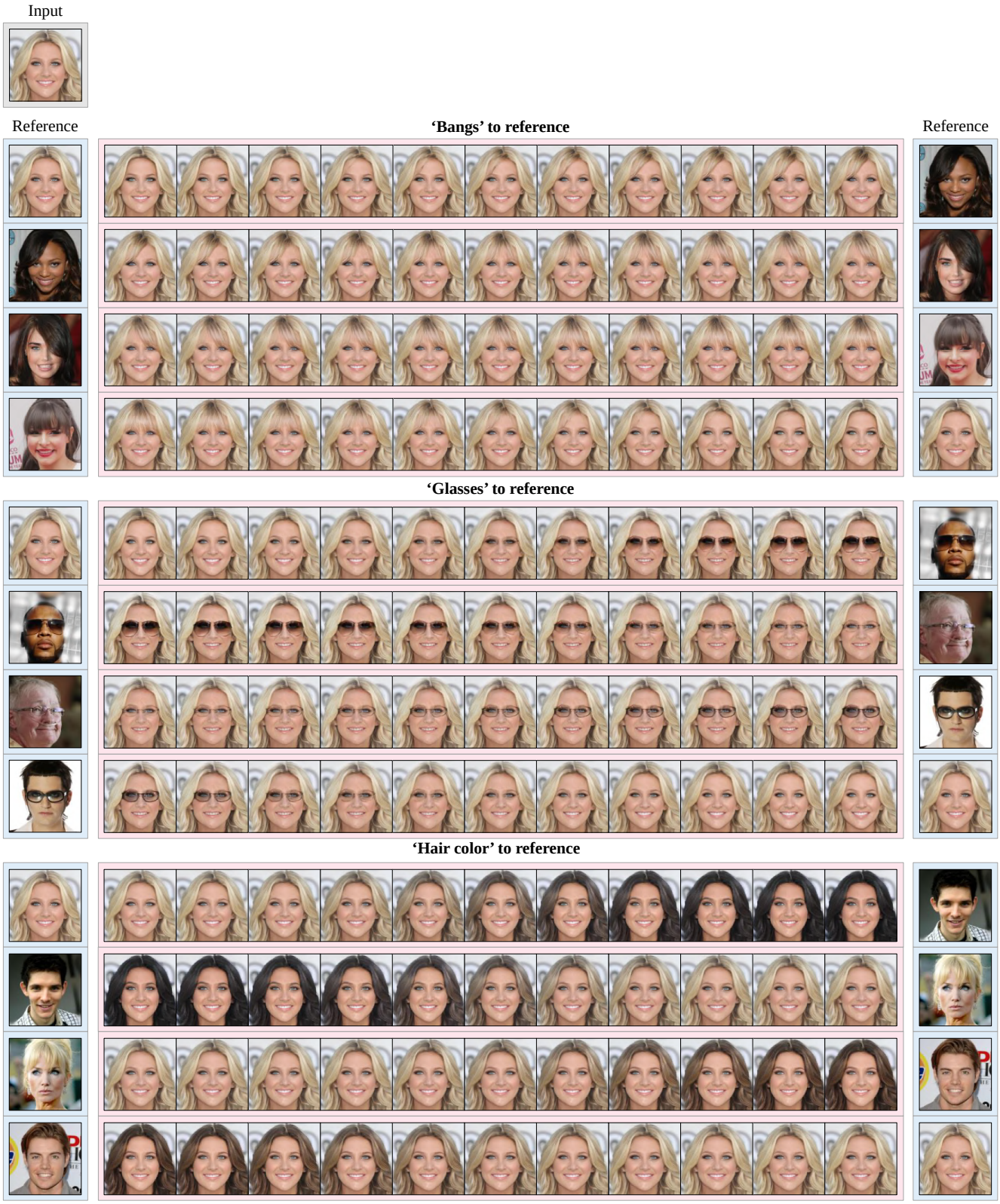


Figure 6: More interpolation results between two extracted tag-relevant styles for different tags.


Figure 7: More interpolation results between two extracted tag-relevant styles for different tags.

**(a) 'Bangs'**

**(b) 'Glasses'**

**(c) 'Hair color'**

Figure 8: 2-D representation of the extracted tag-relevant styles from 180 images using t-SNE for different tags. Please zoom-in for details.

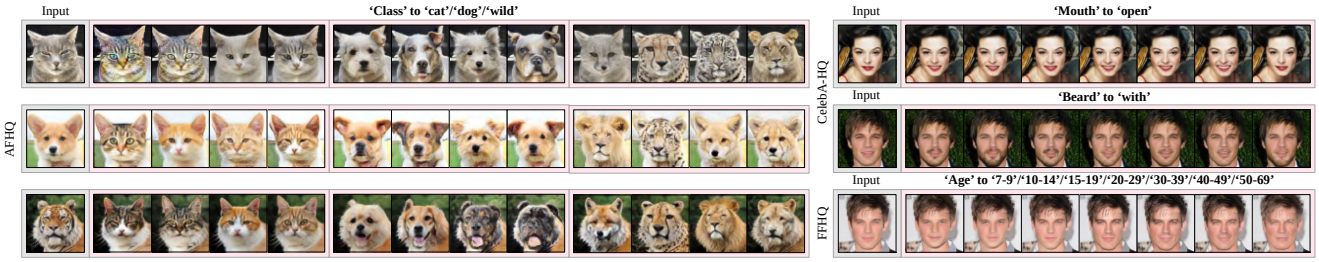


Figure 9: Examples produced by HiSD on other datasets and attributes, The trained tags include: 'Class' ('cat', 'dog' and 'wild' on AFHQ), 'Mouth' ('open' and 'close' on CelebA-HQ), 'Beard' ('with' and 'without' on CelebA-HQ), and 'Age' (from '7-9' to '50-69' on FFHQ).

## References


[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *CVPR*, 2015. 1

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *ECCV*, 2016. 1

[3] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, 2017. 1

[4] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *ICLR*, 2018. 1

[5] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019. 1

[6] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1

[7] Ming-Yu Liu, Xun Huang, Arun Mallya, Tero Karras, Timo Aila, Jaakko Lehtinen, and Jan Kautz. Few-shot unsupervised image-to-image translation. In *ICCV*, 2019. 1

[8] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *ICML*, 2013. 1

[9] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *ICML*, 2018. 1

[10] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *ICLR*, 2018. 1

[11] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017. 1

[12] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. Interfacegan: Interpreting the disentangled face representation learned by gans. *TPAMI*, 2020. 1

[13] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016. 1