Bridging the Gap between Label- and Reference-based Synthesis in Multi-attribute Image-to-Image Translation

Qiusheng Huang[1], Zhilin Zheng[2], Xueqi Hu[1], Li Sun[1]*, Qingli Li[1]
[1]Shanghai Key Laboratory of Multidimensional Information Processing,
[2]PingAn Technology

A. Explanation for Attribute Keeping Loss

To compute L_{ak} in the equation (10), we use E to extract features for unspecified attributes from both the original image X_s and edited image X_g^l, constraining them to be close. The $\mathrm{att}_{ak}^{Y_s\downarrow}$ used in (10) is calculated as (11), and we illustrate its computation process in Fig.1. Firstly, we calculate the two parts A and B in (11) from $\mathrm{att}_{diff}^{Y_s\to Y_t}$. For the part A, we want to select the attributes marked in red, which need to be maintained during the translation. For the part B, we want to flip the attributes of the two images in the form of att_{diff}, so that the attributes can be extracted by E. Finally, we multiply the two parts A and B, and the result $\mathrm{att}_{ak}^{Y_s\downarrow}$ assists E to accurately extract the attribute information that needs to be retained.

B. Details about Network Architecture

In this section, we provide structure details of our method. It consists of four modules as below.

Generator (Fig.2). Our generator G consists of three down-sampling blocks, two intermediate blocks, three up-sampling blocks, and two convolutional layers. We use IN [10] and SPADE [8] for down-sampling and up-sampling blocks, respectively. The relevant feature of the target attribute is added to the G through the SPADE. Note that we use ReLU as activations in G.

Discriminator and Classifier (Fig.3). The discriminator D and the classifier C share four residual blocks with leaky ReLU [7] and a convolutional layer. Due to different tasks, the parameters of last two layers are not shared.

Mapping network (Fig.4). Our Mapping network M consists of a fully connected layer, a convolutional layer and four up-sampling blocks. At the beginning, we concatenate the vector R and att_{diff} in channel dimension. For the residual blocks, we also use IN.

Encoder network (Fig.5). The encoder E is built by four residual blocks with IN and two convolutional layers. At the input of the network, we concatenate the image X and att_{diff} in channel dimension.

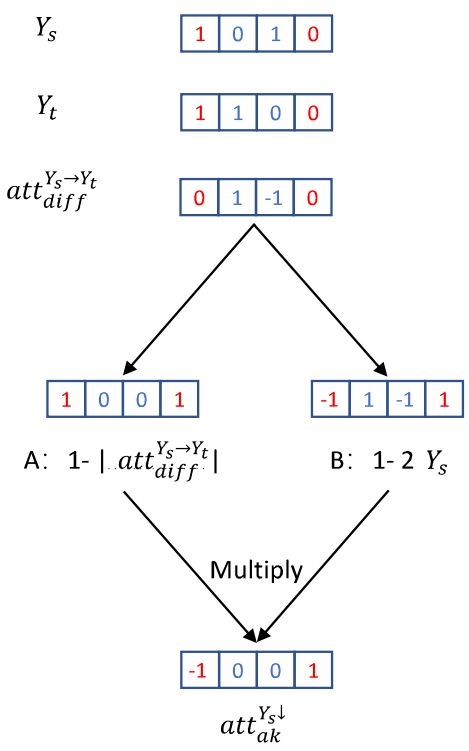

Figure 1: Example of production process of $\mathrm{att}_{ak}^{Y_s\downarrow}$. As is shown above, Y_s is the source label, Y_t is the target label. Numbers marked in red represent attributes that need to be retained, while numbers marked in blue represent attributes that need to be converted.

*Corresponding author, email: sunli@ee.ecnu.edu.cn. This work is supported by the Science and Technology Commission of Shanghai Municipality (No.19511120800)

method	Young	Mouth Slightly Open	Smiling	Black Hair	Blond Hair	Brown Hair	Gray Hair	Receding Hairline	Bangs	Male	No Beard	Mustache	Goatee	Sideburns
StarGAN[2]	85.1	92.0	90.1	86.7	81.8	**85.8**	52.8	84.5	94.8	82.7	66.8	28.5	21.6	20.5
STGAN[5]	82.5	**96.2**	**95.4**	**97.6**	**91.5**	85.0	**86.4**	84.4	**96.9**	**96.5**	70.4	35.6	44.8	65.9
SMIT[9]	32.0	50.1	48.3	41.2	38.2	28.9	7.3	12.3	45.9	51.6	13.9	5.4	4.8	4.3
Ours (Label-based)	**86.9**	95.5	92.2	94.1	90.7	82.1	73.5	**88.0**	94.0	91.8	**86.8**	**71.6**	**80.1**	**89.3**
ELEGANT[11]	18.3	41.2	39.6	52.6	36.8	46.9	11.4	23.0	57.7	19.0	13.5	3.9	5.2	3.7
HomoGAN[1]	26.7	**82.4**	**63.3**	48.3	36.0	21.9	3.6	11.2	43.7	43.7	25.3	4.0	4.8	5.3
Ours (Reference-based)	**71.6**	81.6	61.6	**77.9**	**78.7**	**51.2**	**48.2**	**32.3**	**84.8**	**79.4**	**51.2**	**5.4**	**17.2**	**15.3**

Table 1: Accuracy of each attribute on CelebA. Each column corresponds to one attribute. The best performance is indicated in **bold**.

method	Young	Mouth Slightly Open	Smiling	Black Hair	Blond Hair	Brown Hair	Gray Hair	Receding Hairline	Bangs	Male	No Beard	Mustache	Goatee	Sideburns
StarGAN[2]	19.48	16.64	17.59	22.21	27.59	19.79	47.81	21.52	26.01	22.76	24.36	35.05	25.31	37.29
STGAN[5]	9.32	**3.00**	9.52	17.77	29.53	12.51	50.48	**9.20**	17.73	31.88	**9.53**	**5.53**	**6.42**	13.16
SMIT[9]	11.31	9.98	10.64	**13.45**	**17.71**	11.58	**16.84**	11.15	12.38	16.09	10.00	9.31	9.59	**9.92**
Ours(Label-based)	**8.72**	6.54	**6.13**	15.11	21.08	**9.85**	29.29	11.59	**10.99**	**14.70**	13.16	14.52	14.19	23.90
ELEGANT[11]	**6.37**	7.17	10.12	**9.52**	**17.42**	8.97	50.69	23.26	13.89	30.61	102.08	218.20	228.18	237.90
HomoGAN[1]	16.47	13.70	13.72	18.62	19.60	**17.02**	**17.05**	14.83	14.63	20.11	19.23	19.47	19.96	20.36
Ours(Reference-based)	9.63	**3.98**	**4.23**	9.53	26.36	**7.56**	30.30	**7.60**	**8.61**	**11.27**	**10.74**	**5.70**	**11.06**	**12.88**

Table 2: FID of each attribute on CelebA. Each column corresponds to one attribute. The best performance is indicated in **bold**.

LAYER	RESAMPLE	NORM	OUTPUT
Image X	-	-	128*128*3
Conv1*1	-	-	128*128*32
ResBlk	AvgPool	IN	64*64*64
ResBlk	AvgPool	IN	32*32*128
ResBlk	AvgPool	IN	16*16*256
ResBlk	-	IN	16*16*256
ResBlk	-	SPADE	16*16*256
ResBlk	Upsample	SPADE	32*32*128
ResBlk	Upsample	SPADE	64*64*64
ResBlk	Upsample	SPADE	128*128*32
Conv1*1	-	-	128*128*3

Figure 2: Generator G architecture.

TYPE	LAYER	RESAMPLE	ACTVATION	OUTPUT
Shared	Image X	-	-	128*128*3
Shared	Conv1*1	-	-	128*128*16
Shared	ResBlk	AvgPool	LReLU	64*64*32
Shared	ResBlk	AvgPool	LReLU	32*32*64
Shared	ResBlk	AvgPool	LReLU	16*16*128
Shared	ResBlk	AvgPool	LReLU	8*8*256
Unshared	Conv8*8	-	LReLU	1*1*256
Unshared	FC	-	LReLU	1*1*n / 1

Figure 3: Discriminator D and Classifier C architecture. In our experiment, $n = 14$ which is the number of attributes.

method	dog2cat	wild2cat	cat2dog	wild2dog	cat2wild	dog2wild
StarGAN-V2	15.89	**14.17**	**43.33**	43.21	34.45	35.38
Ours(Label-based)	**15.03**	15.23	50.45	**42.85**	**14.84**	**16.80**
StarGAN-V2	**14.82**	**15.31**	**44.85**	49.21	37.59	38.12
Ours(Reference-based)	17.99	19.63	47.60	**42.88**	**16.14**	**15.82**

Table 3: FID of each conversion on AFHQ. Each column corresponds to one conversion. The best performance is indicated in **bold**.

C. Additional Training Details

The training time is about three days on a single Tesla V100 GPU. The batch size is set to 48. The learning rates for all networks are set to 2×10^{-4}. Loss weights are set as $\lambda_{cls} = 10$, $\lambda_{rec} = 100$, $\lambda_{cyc} = 100$, $\lambda_{ms} = 0.005$, $\lambda_{sty} = 50$, and $\lambda_{ak} = 100$. All networks are optimized by Adam solver [4] ($\beta 1 = 0.5$, $\beta 2 = 0.999$). We show more verification results on AFHQ and CelebA [6] datasets.

D. More Results

Numerous synthesis results of our model on AFHQ and CelebA are shown below. Besides, we compare ours and other methods of all the 14 attributes on CelebA.

Quantitative results. In Tab.1 and Tab.2, we list Accuracy and FID metrics for experiment on CelebA. In the tables, we compare the related works in different generation modes. In Tab.3, we list FID metrics for experiment on the AFHQ dataset, and compare with StarGAN-V2. In this

LAYER	RESAMPLE	ACTVATION	NORM	OUTPUT
Latent R	-	-	-	1*1*d
LabelConcat	-	-	-	1*1*(d+n)
FC	-	-	-	1*1*1024
Reshape	-	-	-	4*4*64
ResBlk	Upsample	ReLU	IN	8*8*32
ResBlk	Upsample	ReLU	IN	16*16*32
ResBlk	Upsample	ReLU	IN	32*32*32
ResBlk	Upsample	ReLU	IN	64*64*32
Conv1*1	-	ReLU	IN	64*64*32

Figure 4: Mapping network M architecture. In our experiment, $d = 16$ which is the length of the noise vector R, $n = 14$ which is the number of attributes.

LAYER	RESAMPLE	ACTVATION	NORM	OUTPUT
Image X	-	-	-	128*128*3
LabelConcat	-	-	-	128*128*(3+n)
Conv1*1	-	-	-	128*128*16
ResBlk	AvgPool	-	IN	64*64*32
ResBlk	-	-	IN	64*64*32
ResBlk	-	-	IN	64*64*32
ResBlk	-	ReLU	IN	64*64*32
Conv1*1	-	ReLU	IN	64*64*32

Figure 5: Style encoder E architecture. In our experiment, $n = 14$ which is the number of attributes.

part, we train the network strictly in accordance with the requirements of StarGAN-V2.

Qualitative results. Fig.6, 7, 8, 9, 10 show our validation results on AFHQ, with a resolution of 256×256. We train StarGAN v2 [3] on the attribute of Bangs, and show the results in the Fig.13. Obviously, the attribute of the generated image and the reference image are not similar, the diversity of the attribute is poor. Besides, the generated image's background, hair color, and other information are not maintained. In CelebA, Fig.11 and 12 show the reference-based results of our model, while Fig.14, 15, 16, 17, 18, and 19 show the results of the comparison between ours and other methods on Label-based synthesis.

References

[1] Ying-Cong Chen, Xiaogang Xu, Zhuotao Tian, and Jiaya Jia. Homomorphic latent space interpolation for unpaired image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2408–2416, 2019.

[2] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8789–8797, 2018.

[3] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8188–8197, 2020.

[4] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *Computer ence*, 2014.

[5] Ming Liu, Yukang Ding, Min Xia, Xiao Liu, Errui Ding, Wangmeng Zuo, and Shilei Wen. Stgan: A unified selective transfer network for arbitrary image attribute editing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3673–3682, 2019.

[6] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. 2014.

[7] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, volume 30, page 3, 2013.

[8] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[9] Andrés Romero, Pablo Arbeláez, Luc Van Gool, and Radu Timofte. Smit: Stochastic multi-label image-to-image translation. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

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

[11] Taihong Xiao, Jiapeng Hong, and Jinwen Ma. Elegant: Exchanging latent encodings with gan for transferring multiple face attributes. In *Proceedings of the European conference on computer vision (ECCV)*, pages 168–184, 2018.

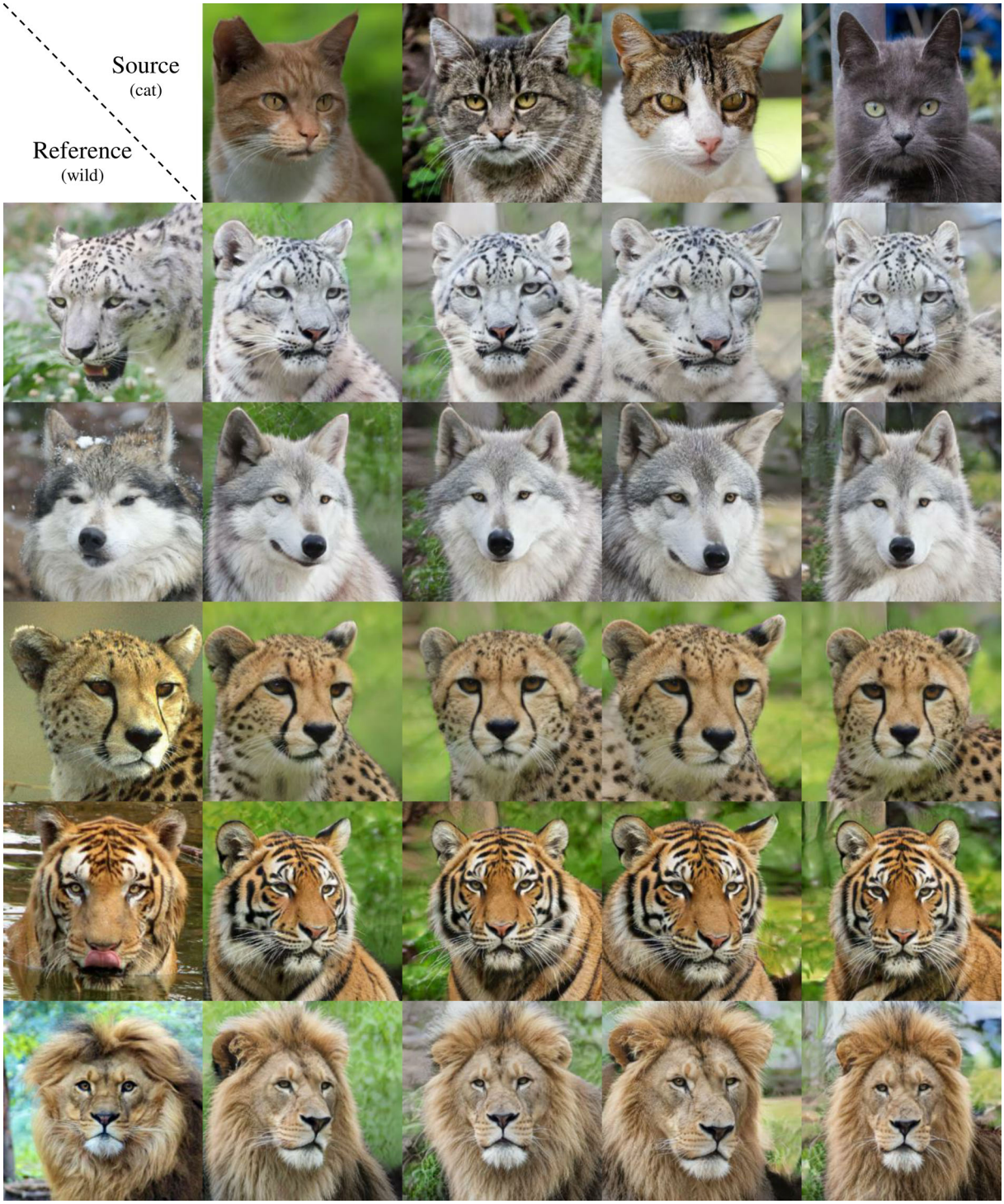

Figure 6: Reference-based synthesis of our model on AFHQ. The first row shows the source images, and the leftmost column represents the reference images.

Figure 7: Reference-based synthesis of our model on AFHQ, following the same format as Figure 6.

Figure 8: Reference-based synthesis of our model on AFHQ, following the same format as Figure 6.

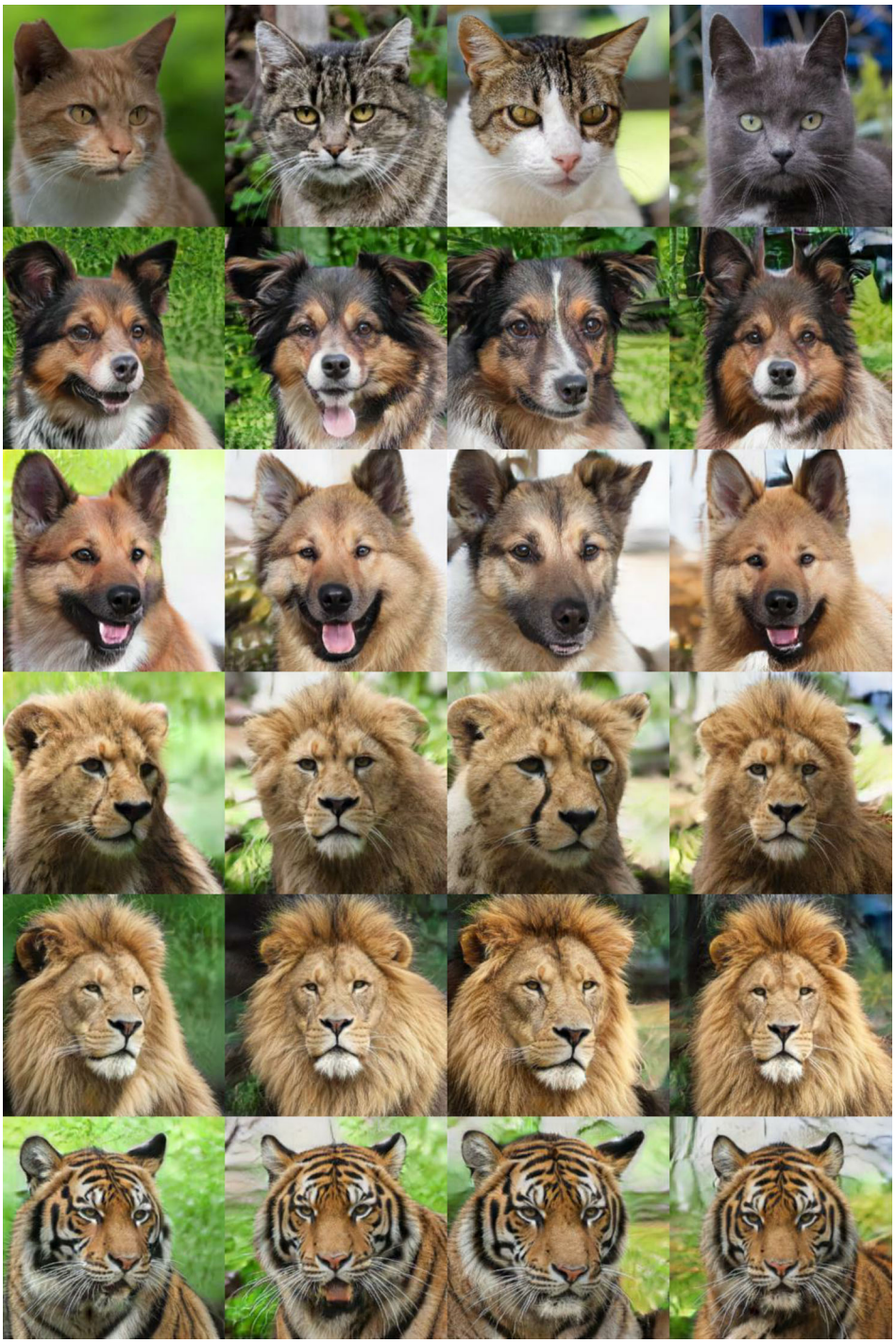

Figure 9:. Label-based synthesis of our model on AFHQ. The first row shows the source images, remaining rows are generated by giving labels.

Figure 10: Label-based synthesis of our model on AFHQ, following the same format as Figure 9.

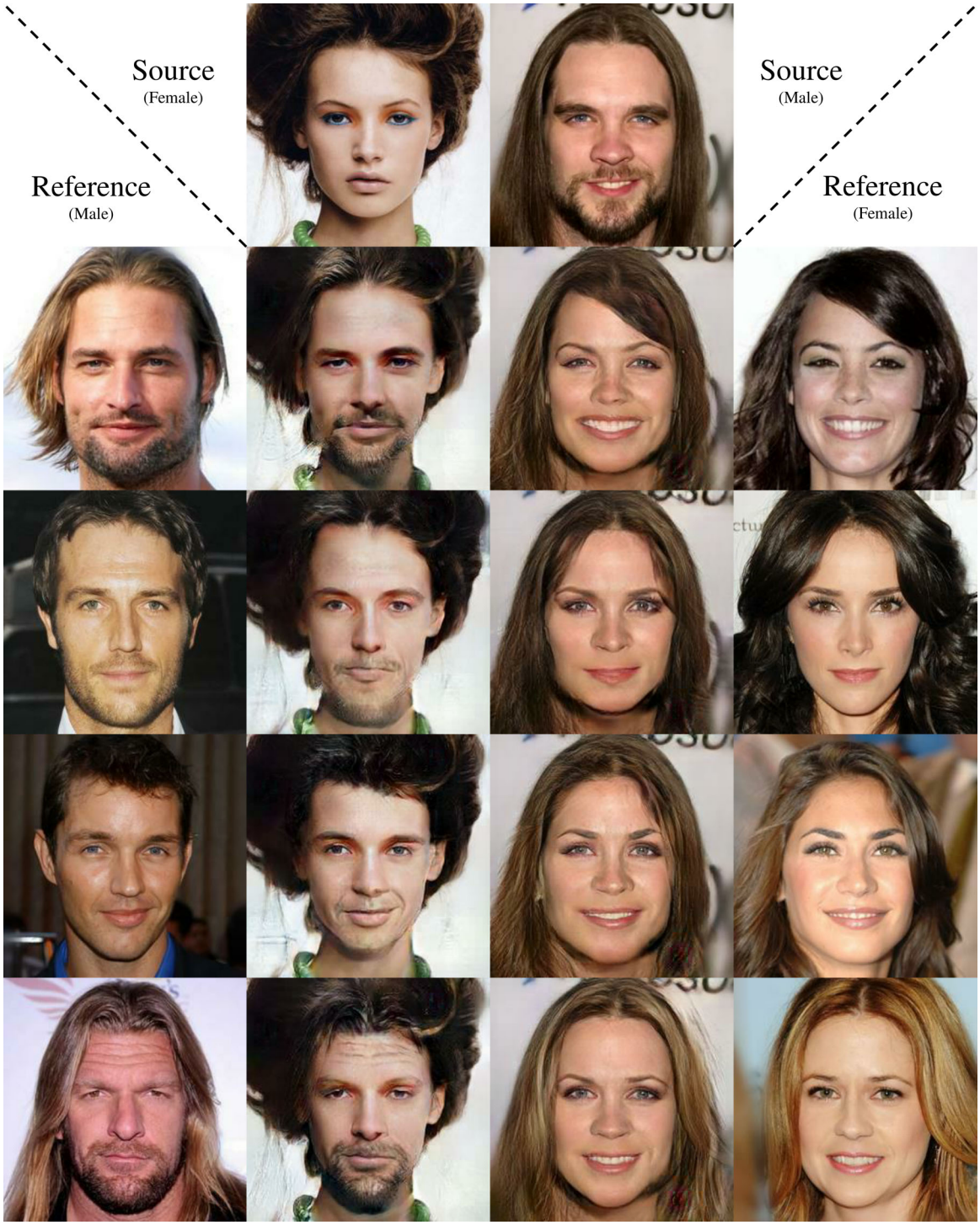

Figure 11: Reference-based synthesis results on the attribute of gender. The first row shows the source images, two columns on the edge are the reference images, and the middle two columns are the generated images.

Figure 12: Reference-based synthesis results on the attributes of mouth-open and bangs, following the same format as Figure 11. Note that in multi-attributes translation, the two attributes should be changed at the same time.

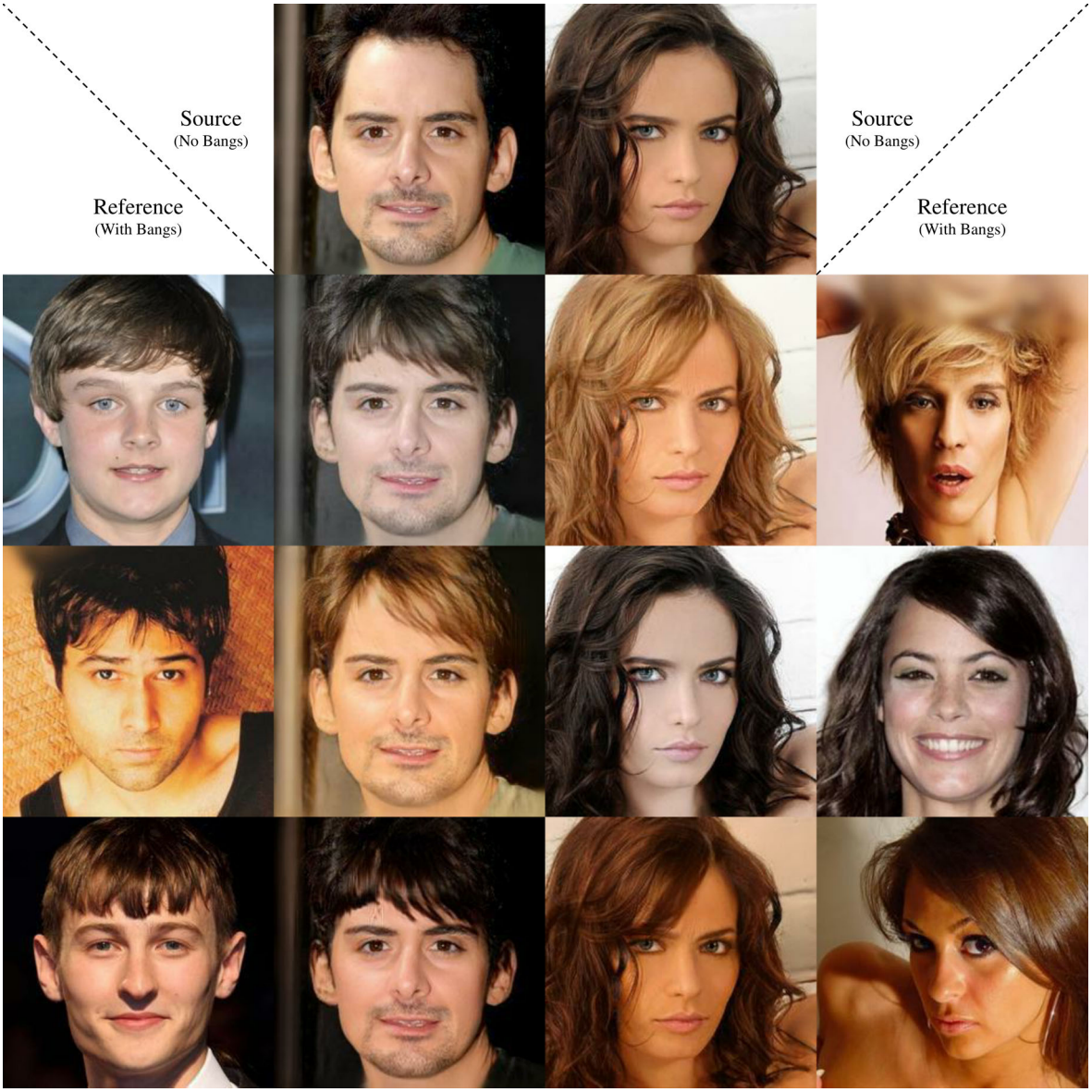

Figure 13: Reference-based synthesis results on the attribute of bangs by StarGAN v2. Please compare these results with ours in Figure 11 and 12

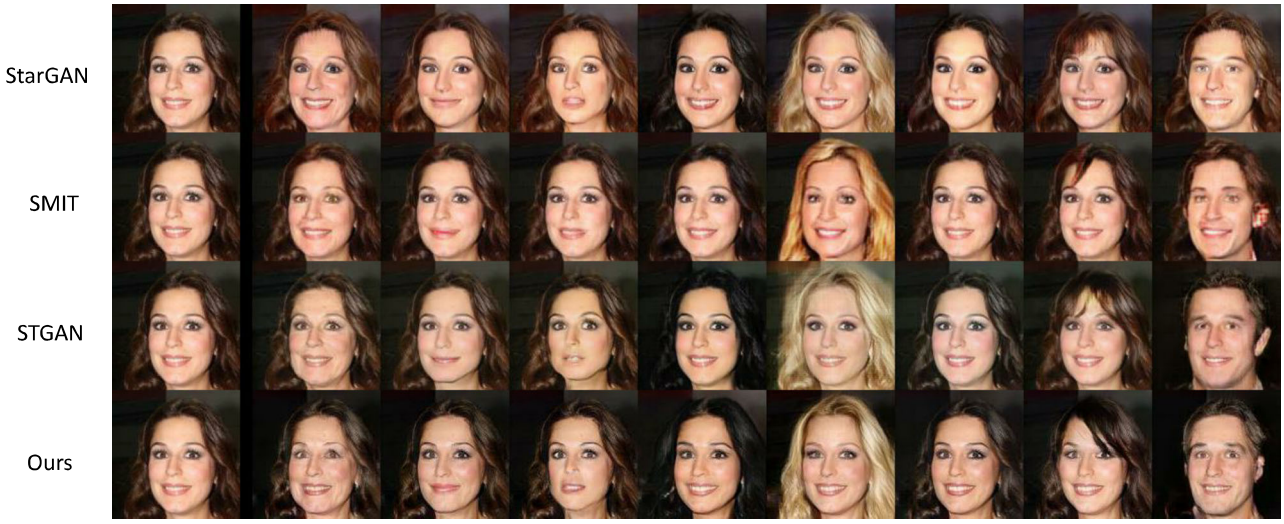

Figure 14: Label-based synthesis of 4 models on 14 attributes. From left to right: source, young, mouth slightly open, smiling, black, blond, brown hair, bangs, male. Please zoom in for details.

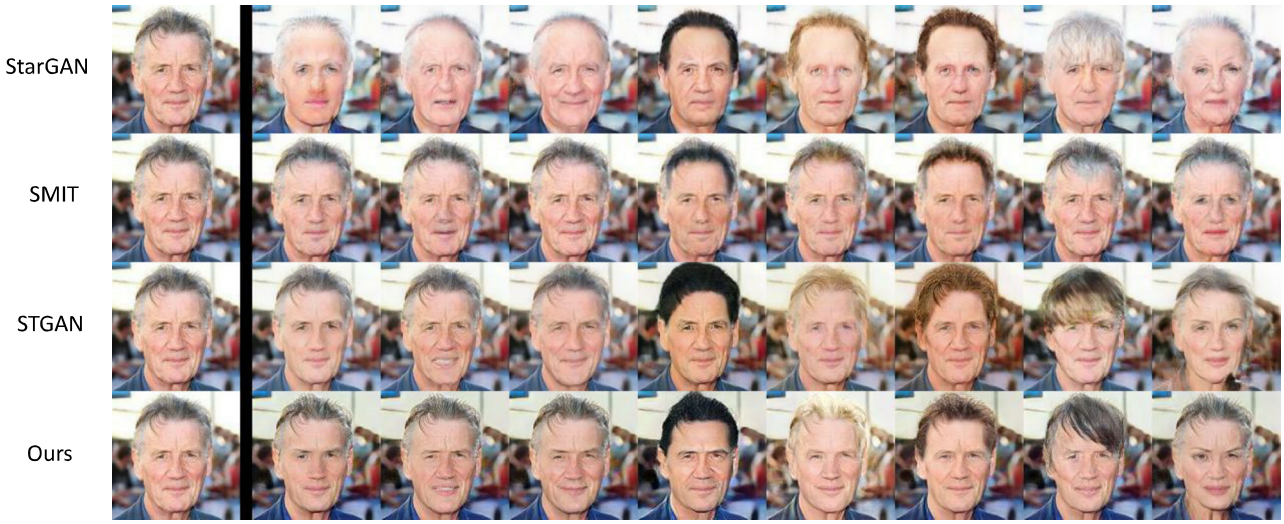

Figure 15: Label-based synthesis of 4 models on 14 attributes, following the same format as Figure 14.

Figure 16: Label-based synthesis of 4 models on 14 attributes, following the same format as Figure 14.

Figure 17: Label-based synthesis of 4 models on 14 attributes, following the same format as Figure 14.

Figure 18: Label-based synthesis of 4 models on 14 attributes, following the same format as Figure 14.

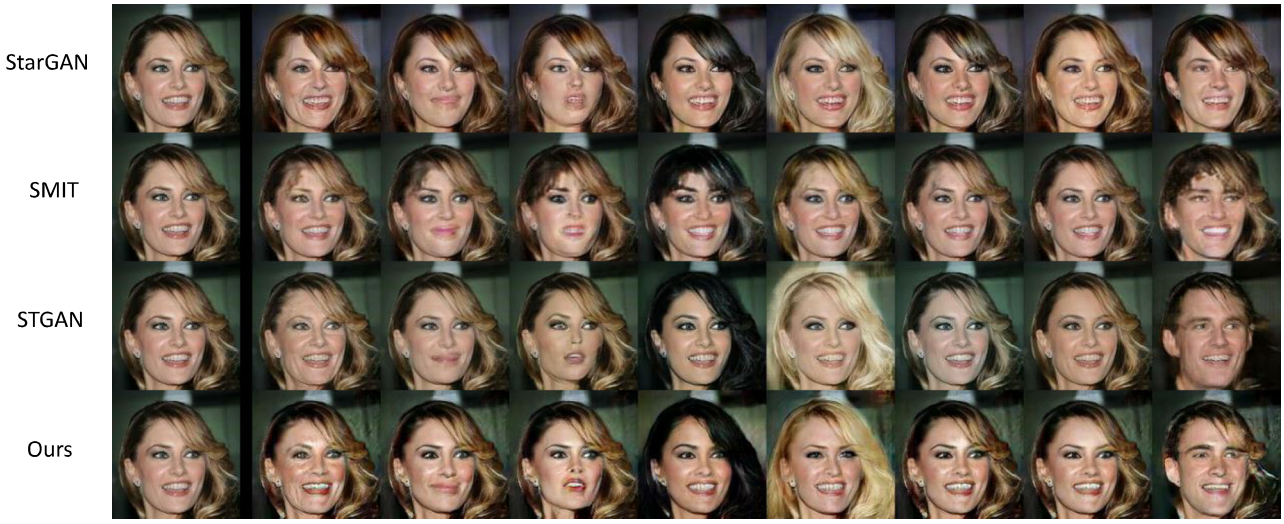

Figure 19: Label-based synthesis of 4 models on 14 attributes, following the same format as Figure 14.