Supplementary Material of Domain Adaptive Image-to-image Translation

Ying-Cong Chen[1] Xiaogang Xu[1] Jiaya Jia[1,2]
[1]The Chinese University of Hong Kong [2]SmartMore
yingcong.ian.chen@gmail.com, {xgxu,leojia}@cse.cuhk.edu.hk

1. Implementation Details

1.1. Network Architecture

Our model contains a base I2I model $F_{\boldsymbol{A}^-\mapsto\boldsymbol{A}^+}$, two domain adaption submodules $F_{\boldsymbol{B}\mapsto\boldsymbol{A}}$ and $F_{\boldsymbol{A}\mapsto\boldsymbol{B}}$, one discriminator $\boldsymbol{D}$, and one image classification network $\mathcal{C}$. The architecture of $F_{\boldsymbol{A}^-\mapsto\boldsymbol{A}^+}$ depends on the base I2I model, which will not be discussed here. $F_{\boldsymbol{B}\mapsto\boldsymbol{A}}$ and $F_{\boldsymbol{A}\mapsto\boldsymbol{B}}$ have the same architecture which is defined in Table 1, i.e., both contain three convolutional layers that conduct 4x downsampling, 4 residual blocks, two transposed convolutional layers that upsample the feature map to the original size, and finally a convolutional layer with Tanh activation that produces output of the right shape and scale. The discriminator $\boldsymbol{D}$ stacks several convolutional layers with LeakyReLU, whose architecture is shown in Table 3. The classifier $\mathcal{C}$ is a typical CNN with two fully connected layers, whose architecture is shown in Table 4.

1.2. Training Details

We summarize the training detail of our framework in Algorithm 1. Note that Eq. (1) - Eq. (4) are defined in the manuscript. Data augmentation is used in both source domain and target domain, which includes random shit, random rotation, random scale, random color jitter and random flip. We use Adam optimizer [1] to train our framework, with β_1 and β_2 set as 0.5 and 0.999 respectively. The learning rate is set as 10^{-4}, and the batch size is set as 16. The model is trained until $\boldsymbol{\mathcal{L}}_{ADA}$, $\boldsymbol{\mathcal{L}}_{rec}$ and $\boldsymbol{\mathcal{L}}_{PA}$ converge.

1.3. Datasets

Table. 5 shows the links of related datasets that are used in this paper. Note that RaFD [2] serves as the source domain $\boldsymbol{A}$ in our paper, and all frontal samples are used during training. For Multi-PIE, images of Session 1 are used for training, following [3]. For CelebA, the training/testing separation follows [4]. For cat faces, oil paintings and sketches, please refer to the manuscript of the training/testing separation. All images are resized to 128×128.

Algorithm 1 Training the DAI2I framework

Require: Training samples of $\boldsymbol{A}^-$, $\boldsymbol{A}^+$ and $\boldsymbol{B}^-$; A well trained I2I model $F_{\boldsymbol{A}^-\mapsto\boldsymbol{A}^+}$; A well trained classification network $\mathcal{C}$;
Ensure: $F_{\boldsymbol{B}\mapsto\boldsymbol{A}}$ and $F_{\boldsymbol{A}\mapsto\boldsymbol{B}}$;
  while not converged **do**
    $t \leftarrow 0$
    while $t < 5$ **do**
      Update the Critic $\boldsymbol{D}$ based on Eq. (1);
    end while
    Update $F_{\boldsymbol{B}\mapsto\boldsymbol{A}}$ and $F_{\boldsymbol{A}\mapsto\boldsymbol{B}}$ based on Eq. (1-5);
    $t = t + 1$.
  end while

1.4. Running time

We test our framework in a TITAN V GPU. For cross-domain expression manipulation task, it takes 22ms to process an image; for the cross-domain novel view synthesis task, it takes 26ms. The difference in running speed is due to different base I2I model ($F_{\boldsymbol{A}^-\mapsto\boldsymbol{A}^+}$).

2. The user study

Fig. 1 shows snapshots of the expression classification test and the quality comparison test. In Table 2 and Table 3 of the manuscript, different datasets are counted separately, and each entry includes 2500 comparisons - that is, it is tested by 50 subjects, and each subject conducts 50 comparisons randomly sampled from the testing sets of sketches (59 images), paintings (92 images), or cats (100 images).

3. Additional Experiments

3.1. Visualization of the style feature

Fig. 2 shows visualization of the style feature tSNE. It shows that the feature is meaningful, as images of similar appearance are located nearly, and those with different styles/categories are located separately.

Layer Type	Norm	Activation	Kernel	Stride	Padding	Output Size
Input	-	-	-	-	-	$128 \times 128 \times 3$
Convolution	AdaIN	LeakyReLU	7	1	3	$128 \times 128 \times 128$
Convolution	AdaIN	LeakyReLU	4	2	1	$64 \times 64 \times 128$
Residual Block	AdaIN	LeakyReLU	3	1	1	$64 \times 64 \times 256$
Residual Block	AdaIN	LeakyReLU	3	1	1	$64 \times 64 \times 256$
Residual Block	AdaIN	LeakyReLU	3	1	1	$64 \times 64 \times 256$
Residual Block	AdaIN	LeakyReLU	3	1	1	$64 \times 64 \times 256$
Transposed Convolution	AdaIN	LeakyReLU	4	2	1	$128 \times 128 \times 64$
Convolution	-	Tanh	7	1	3	$128 \times 128 \times 3$

Table 1. The architecture of the adapter $F_{\boldsymbol{B}\mapsto\boldsymbol{A}}$ and the reconstructor $F_{\boldsymbol{A}\mapsto\boldsymbol{B}}$. The Residual Block is formulated as $y = f(x) + x$ where $f(\cdot)$ is a convolutional neural network that sequentially stacks Convolution, AdaIN, LeakyReLU, Convolution and AdaIN.

Layer Type	Norm	Activation	Kernel	Stride	Padding	Output Size
Input	-	-	-	-	-	$128 \times 128 \times 3$
Convolution	-	LeakyReLU	4	2	1	$64 \times 64 \times 16$
Convolution	-	LeakyReLU	4	2	1	$32 \times 32 \times 32$
Convolution	-	LeakyReLU	4	2	1	$16 \times 16 \times 64$
Convolution	-	LeakyReLU	4	2	1	$8 \times 8 \times 128$
Convolution	-	LeakyReLU	4	2	1	$4 \times 4 \times 256$
AvePool	-	-	4	1	0	$1 \times 1 \times 256$

Table 2. Architecture of the style net $S(\cdot)$.

Layer Type	Norm	Activation	Kernel	Stride	Padding	Output Size
Input	-	-	-	-	-	$128 \times 128 \times 3$
Convolution	-	LeakyReLU	4	2	1	$64 \times 64 \times 64$
Convolution	-	LeakyReLU	4	2	1	$32 \times 32 \times 128$
Convolution	-	LeakyReLU	4	2	1	$16 \times 16 \times 256$
Convolution	-	LeakyReLU	4	2	1	$8 \times 8 \times 512$
Convolution	-	LeakyReLU	8	1	0	$1 \times 1 \times 1$

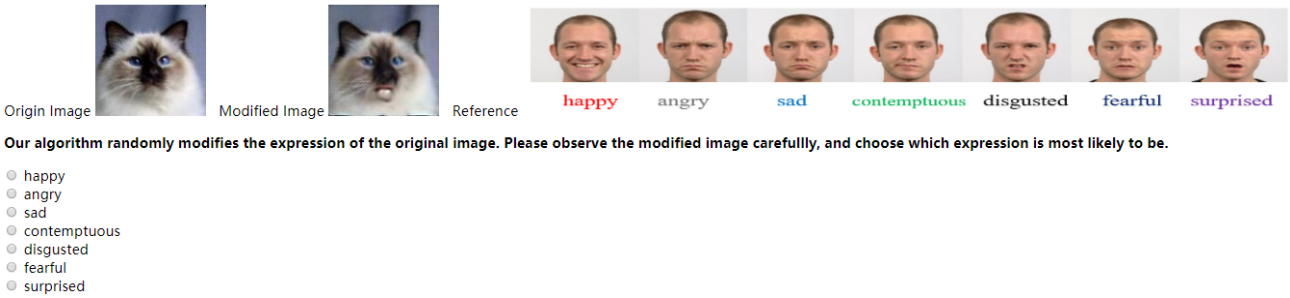
Table 3. Architecture of the Discriminator $\boldsymbol{D}$.

Layer Type	Norm	Activation	Kernel	Stride	Padding	Output Size
Input	-	-	-	-	-	$128 \times 128 \times 3$
Convolution	BatchNorm	LeakyReLU	4	2	1	$64 \times 64 \times 8$
Convolution	BatchNorm	LeakyReLU	4	2	1	$32 \times 32 \times 16$
Convolution	BatchNorm	LeakyReLU	4	2	1	$16 \times 16 \times 32$
Convolution	BatchNorm	LeakyReLU	4	2	1	$8 \times 8 \times 64$
Convolution	BatchNorm	LeakyReLU	4	2	1	$4 \times 4 \times 128$
Convolution	BatchNorm	LeakyReLU	4	2	1	$2 \times 2 \times 256$
Convolution	BatchNorm	LeakyReLU	4	2	1	$1 \times 1 \times 512$
Fully Connected Layer	-	LeakyReLU	512×64			64
Fully Connected Layer	-	LeakyReLU	$64\times$ Number of Class			Number of Class

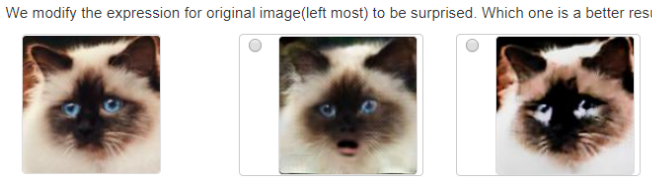
Table 4. The architecture of Image Classification Network $\mathcal{C}$.

Dataset	Link
RaFD [2]	http://www.socsci.ru.nl:8180/RaFD2/RaFD
CelebA [4]	http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html
Multi-PIE [5]	http://www.cs.cmu.edu/afs/cs/project/PIE/MultiPie/Multi-Pie/Home.html
cat faces [6]	https://github.com/HsinYingLee/DRIT
oil paintings [6]	https://github.com/HsinYingLee/DRIT
sketches [7]	http://mmlab.ie.cuhk.edu.hk/archive/facesketch.html

Table 5. Links of the related datasets.



(a) Expression Classification Test.



(b) Quality Comparison Test.

Figure 1. Snapshots of the user study.

Figure 2. Visualization of the style feature with tSNE.

3.2. Comparison with Image Analogies

In principle, our framework is related to image analogies [8] as we infer $\boldsymbol{B}^+$ based on the relation of $\boldsymbol{A}^-$ and $\boldsymbol{A}^+$. One may concern whether similar results can be obtained if paired A^- and A^+ is available. In Fig. 5, we show that even with paired information, image analogies [8] cannot handle high-level semantic translation. Specifically, we take an RaFD image with neutral expression as A^-, another image of the same identity but different expression as A^+, and a sketch/painting/cat face image as B^-, then use [8] to generate B^+ based on A^-, A^+ and B^-. Note that A^- and A^+ are of the same identity as shown in row 1 of Fig. 5. The results are shown in row 3, 5 and 7 of Fig. 5, which indicate that traditional image analogies [8] cannot deal with our task.

3.3. Additional Results

We have tried our cross-domain expression manipulation model with oil paintings downloaded from Internet. For these images, faces are detected and cropped, processed with our DAI2I framework, and blend back to the original images. As shown in Fig. 3 and 4, our model can correctly modify the expression in oil paintings. This demonstrates the effectiveness of our method in practical use.

Fig. 6, 7 and 8 shows additional results of cross-domain expression manipulation on sketches, oil paintings and cat faces. The base I2I model is StarGAN trained on RaFD. Besides the baseline StarGAN, we also compare our results with another image-to-image translation method, the ComboGAN [9]. As shown, our model consistently outperforms the two methods.

Note that even if the base I2I model is trained on a large dataset (i.e., CelebA [4]), it still cannot generalize well for very different target domains. This is shown in Fig. 9. In this experiment, all base I2I models are trained on CelebA, and the *Smiling* attribute is used. It shows that although StarGAN and ComboGAN produce lighter artifacts than Fig. 8, they cannot modify the target attribute correctly, as cat faces are very different from human. Our model successfully turns a cat towards smiling without introducing many artifacts.

Fig. 10 and 11 shows additional results of cross-domain novel view synthesis. As shown, our results are consistently better than the two baselines.

References

[1] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv e-prints*, 2014. 1

[2] Oliver Langner, Ron Dotsch, Gijsbert Bijlstra, Daniel HJ Wigboldus, Skyler T Hawk, and AD Van Knippenberg. Presentation and validation of the radboud faces database. *Cognition and emotion*, 2010. 1, 2

[3] Yu Tian, Xi Peng, Long Zhao, Shaoting Zhang, and Dimitris N Metaxas. Cr-gan: learning complete representations for multi-view generation. *arXiv e-prints*, 2018. 1

[4] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *IEEE International Conference on Computer Vision*, 2015. 1, 2, 4

[5] Ralph Gross, Iain Matthews, Jeffrey Cohn, Takeo Kanade, and Simon Baker. Multi-pie. *Image and Vision Computing*, 2010. 2

[6] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 35–51, 2018. 2

[7] Xiaogang Wang and Xiaoou Tang. Face photo-sketch synthesis and recognition. 2008. 2

[8] Aaron Hertzmann, Charles E Jacobs, Nuria Oliver, Brian Curless, and David H Salesin. Image analogies. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001. 4, 7

[9] Asha Anoosheh, Eirikur Agustsson, Radu Timofte, and Luc Van Gool. Combogan: Unrestrained scalability for image domain translation. In *CVPRW*, 2018. 4

Input Happy Angry sad

Contemptuous Disgusted Fearful Surprised

Input Happy Angry sad

Contemptuous Disgusted Fearful Surprised

Figure 3. Results of oil painting images downloaded from Internet.

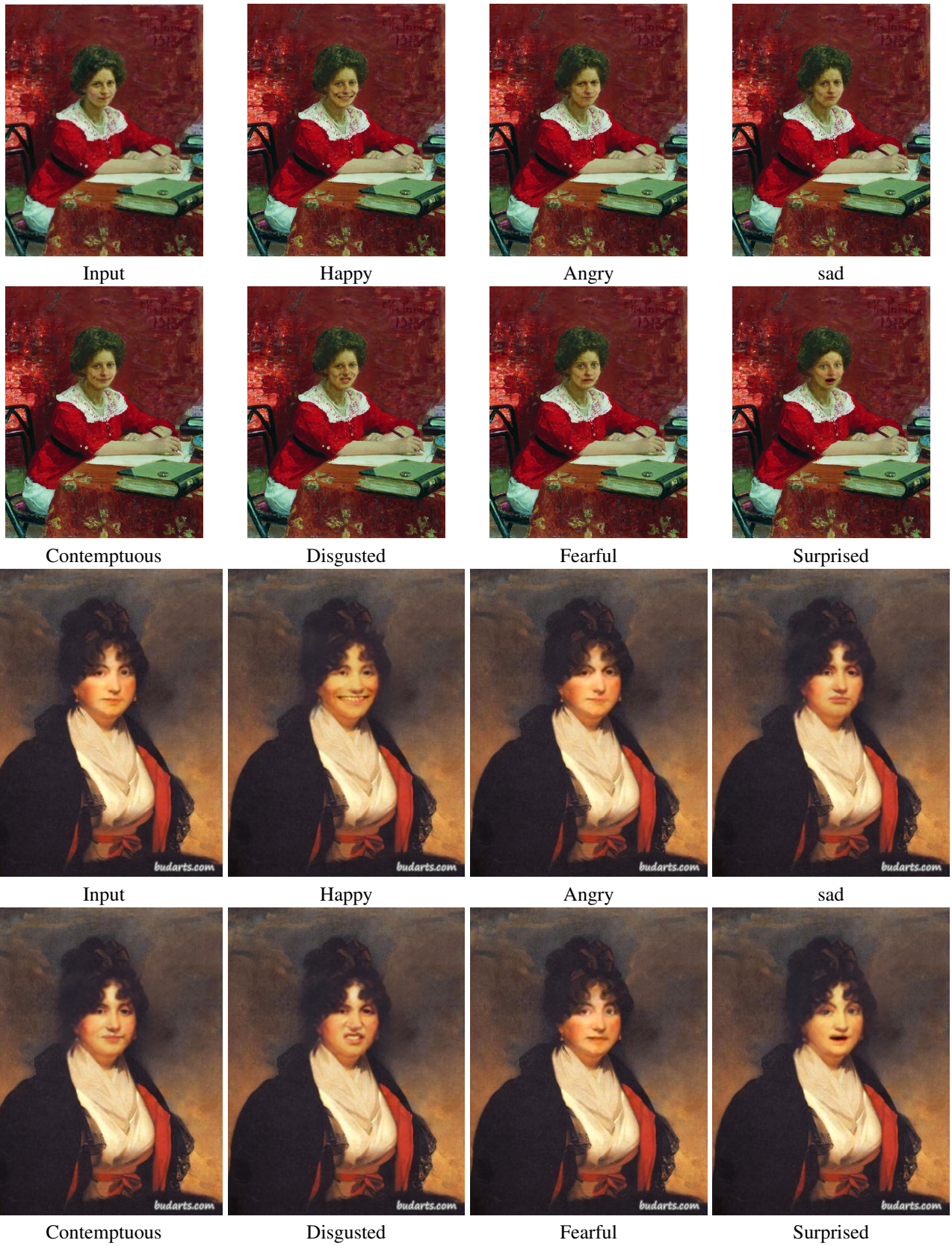


Figure 4. Results of oil painting images downloaded from Internet.

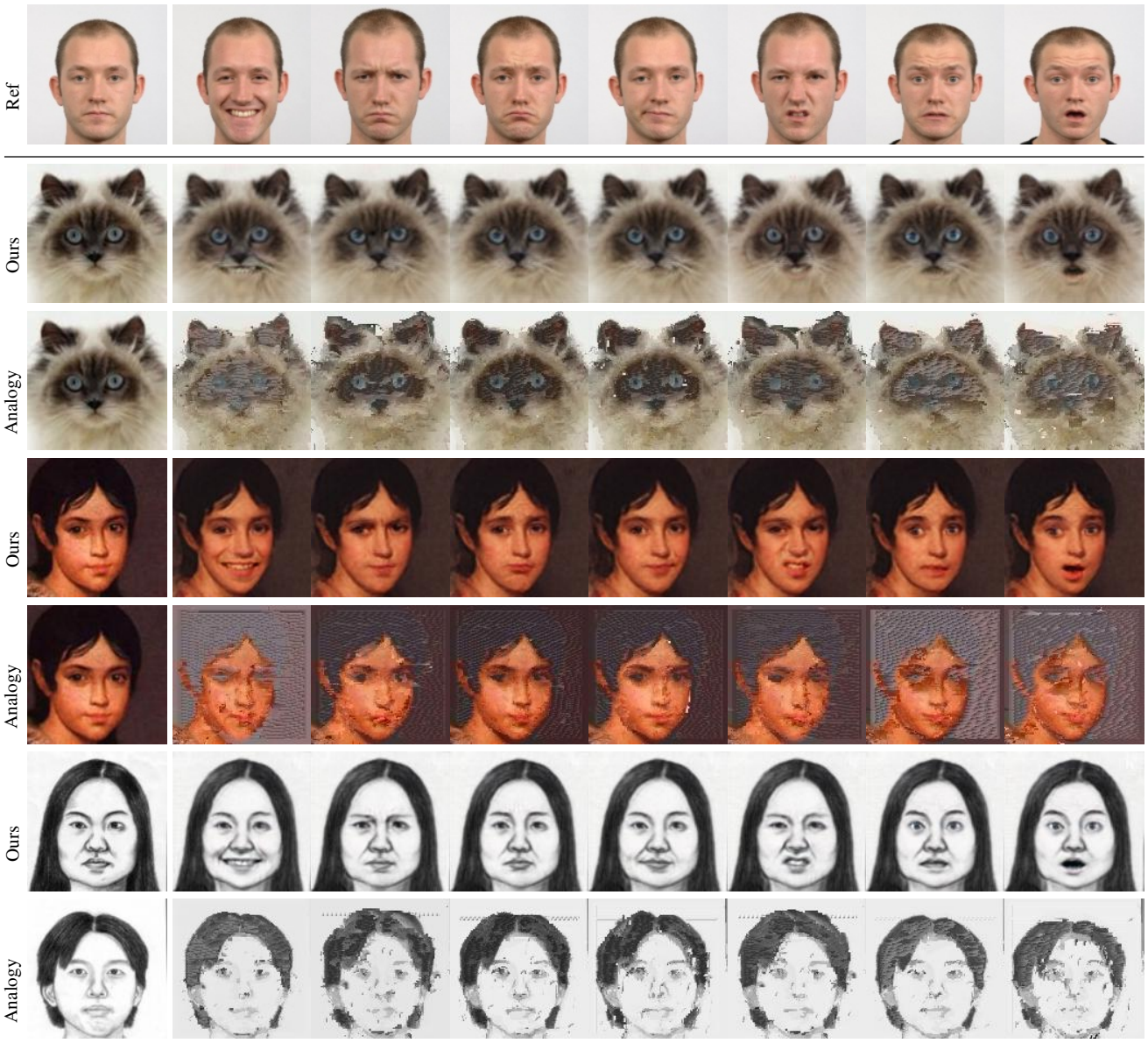


Figure 5. Comparing with image analogies [8]. The 1st row shows images that are used as reference images (A^- and A^+) for [8]. Other rows compare the results of our approach with image analogies [8].

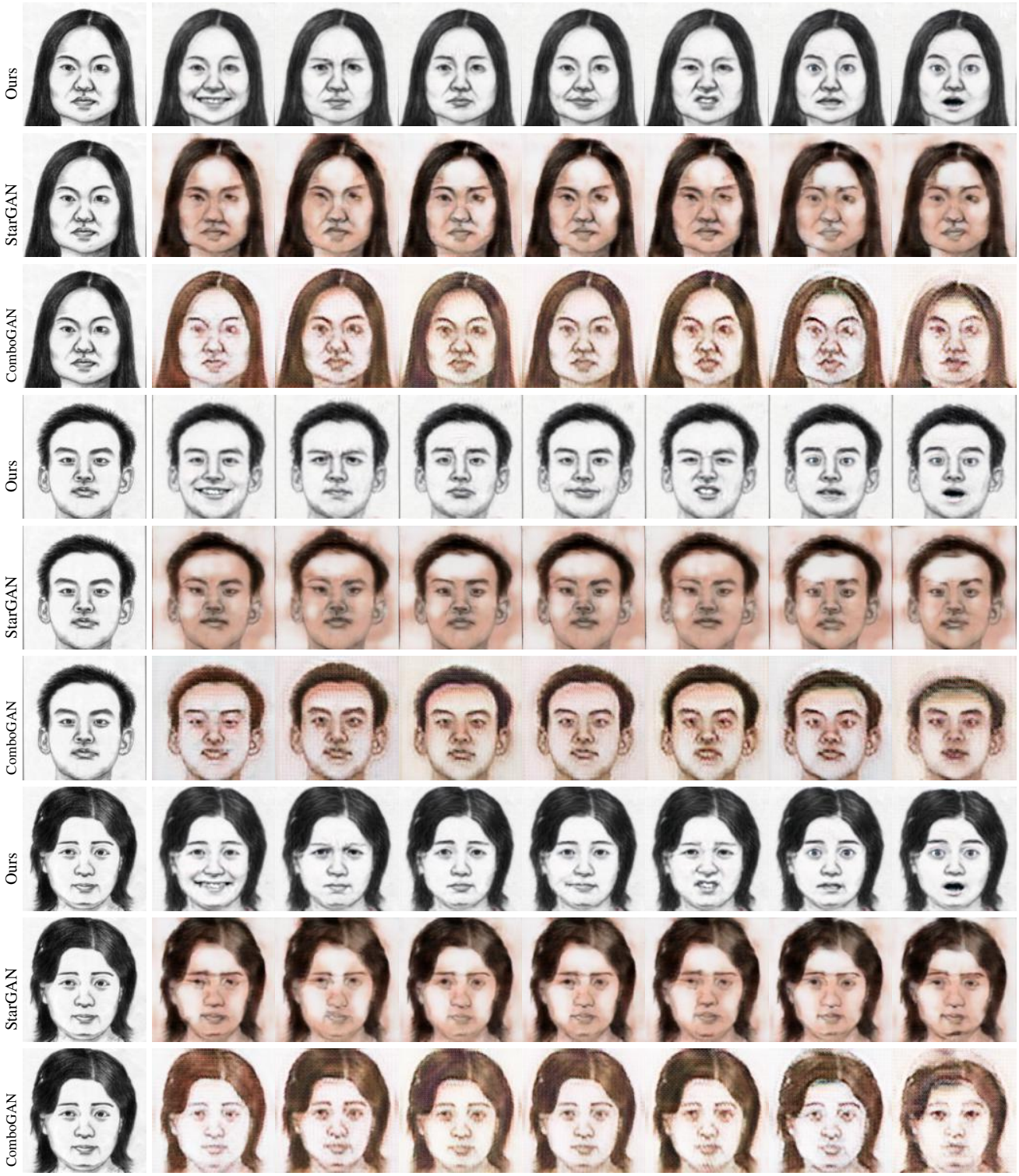


Figure 6. Additional results on Sketches.

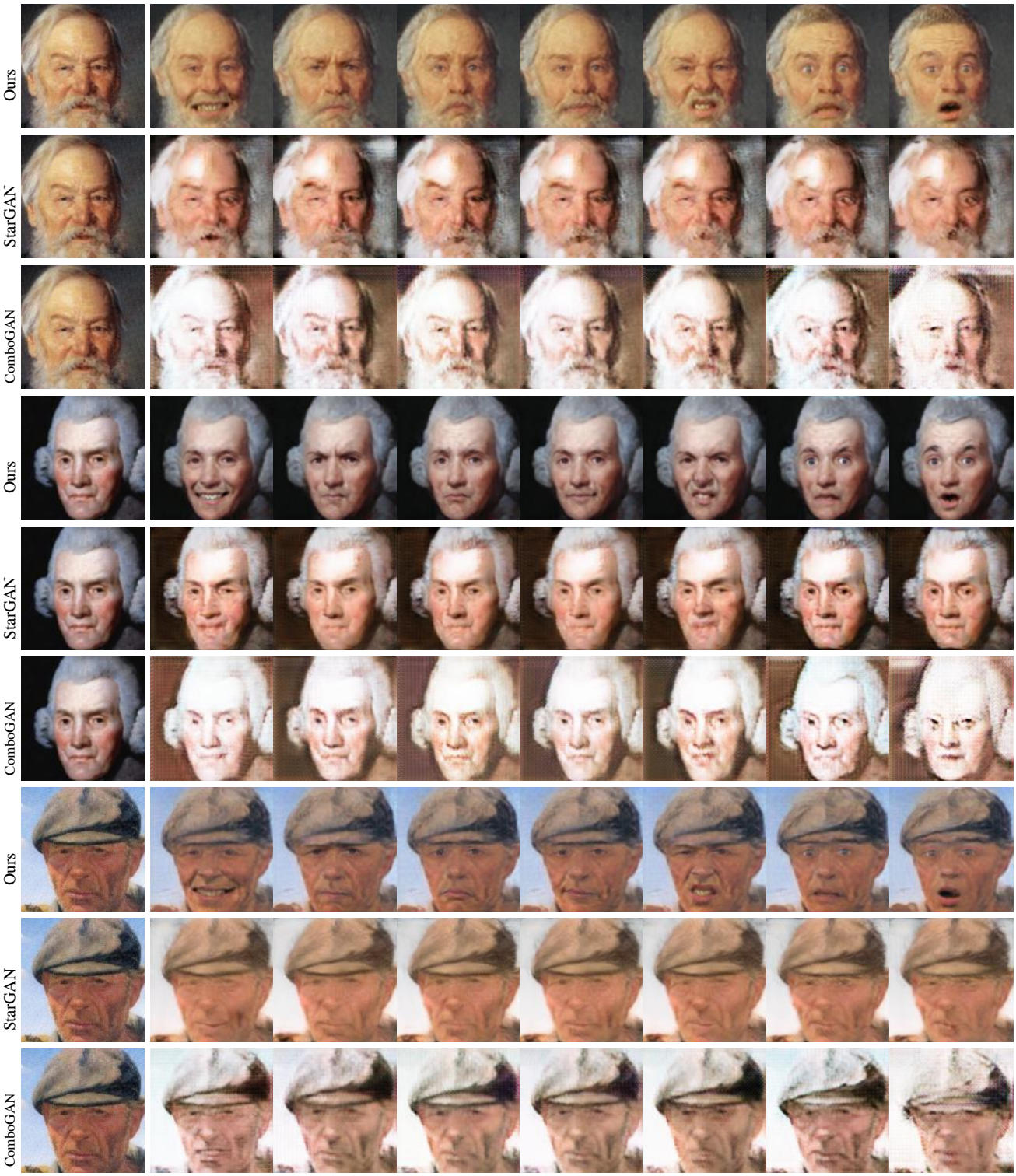


Figure 7. Additional results on oil paintings.

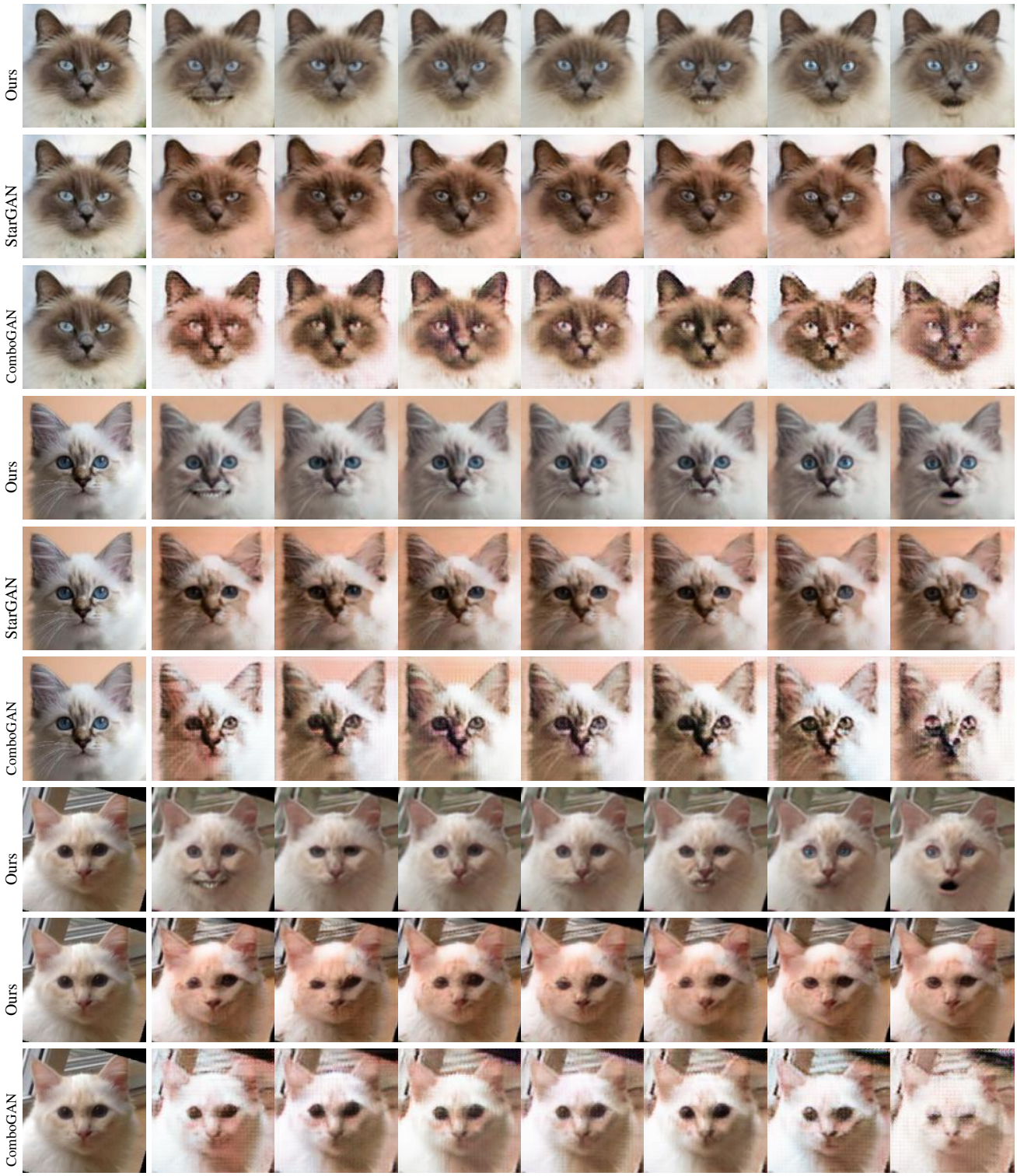


Figure 8. Additional results on cat faces.

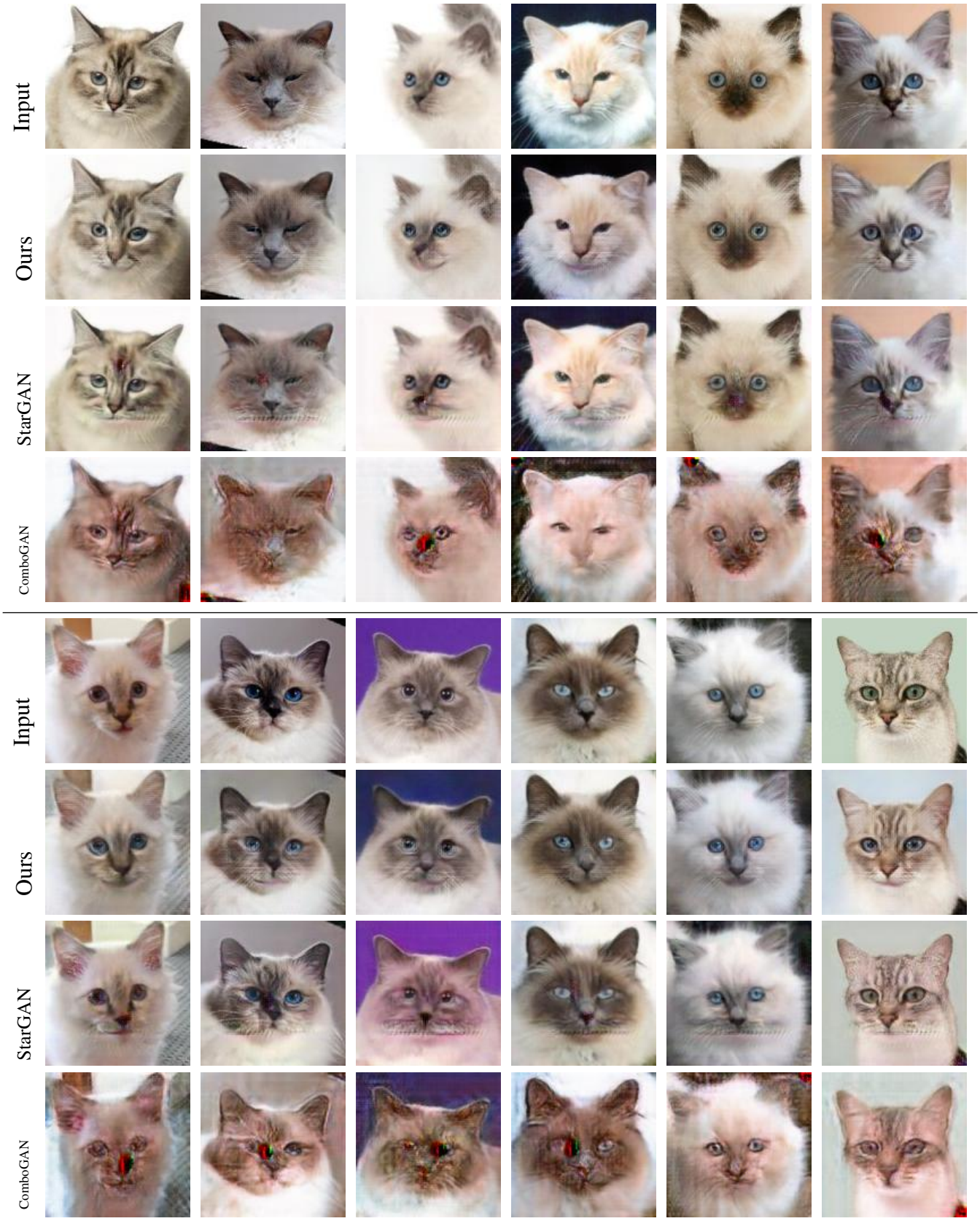


Figure 9. Results of cross-view expression manipulation based on StarGAN trained on CelebA.

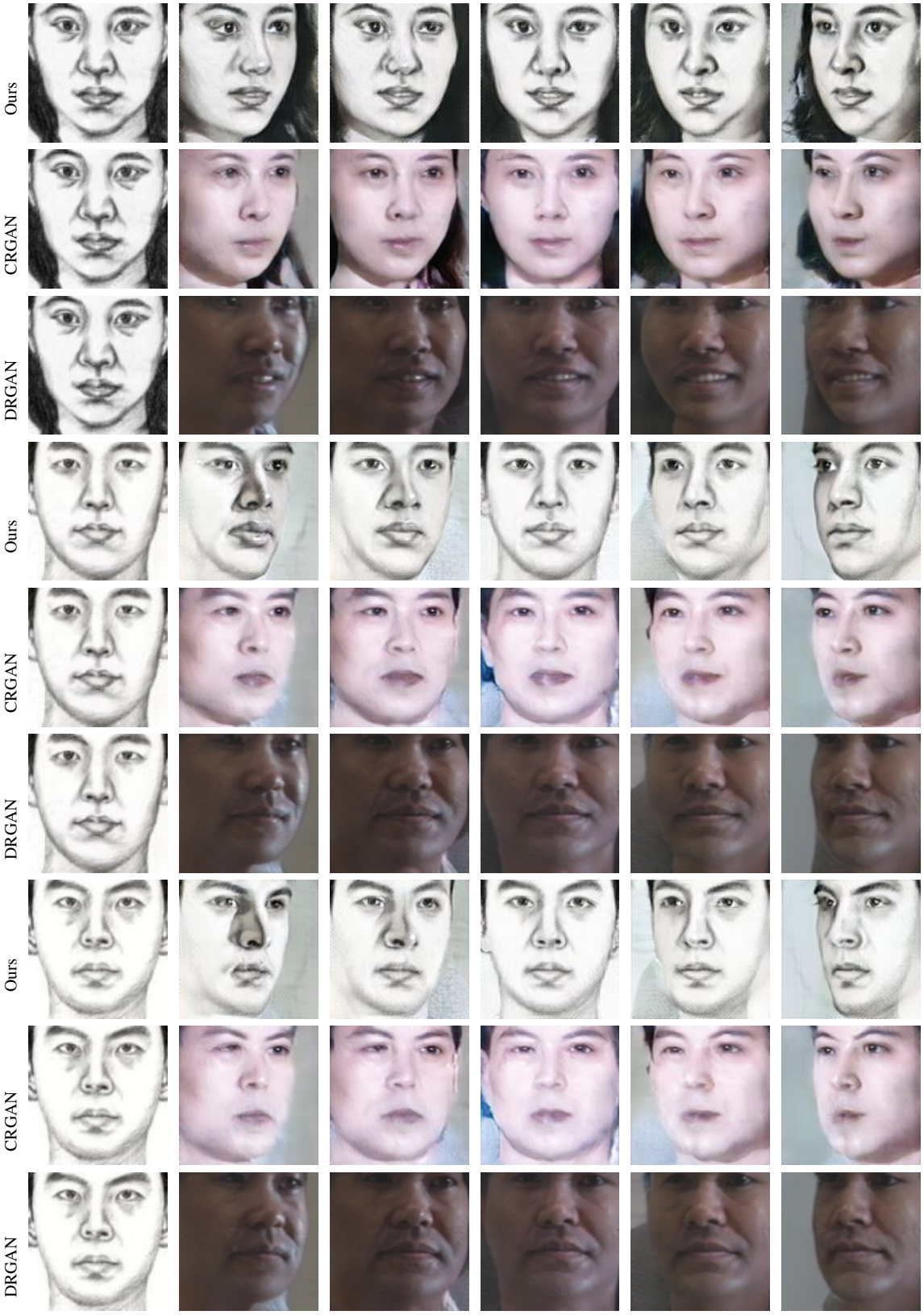


Figure 10. Additional results on cross-domain novel view synthesis.

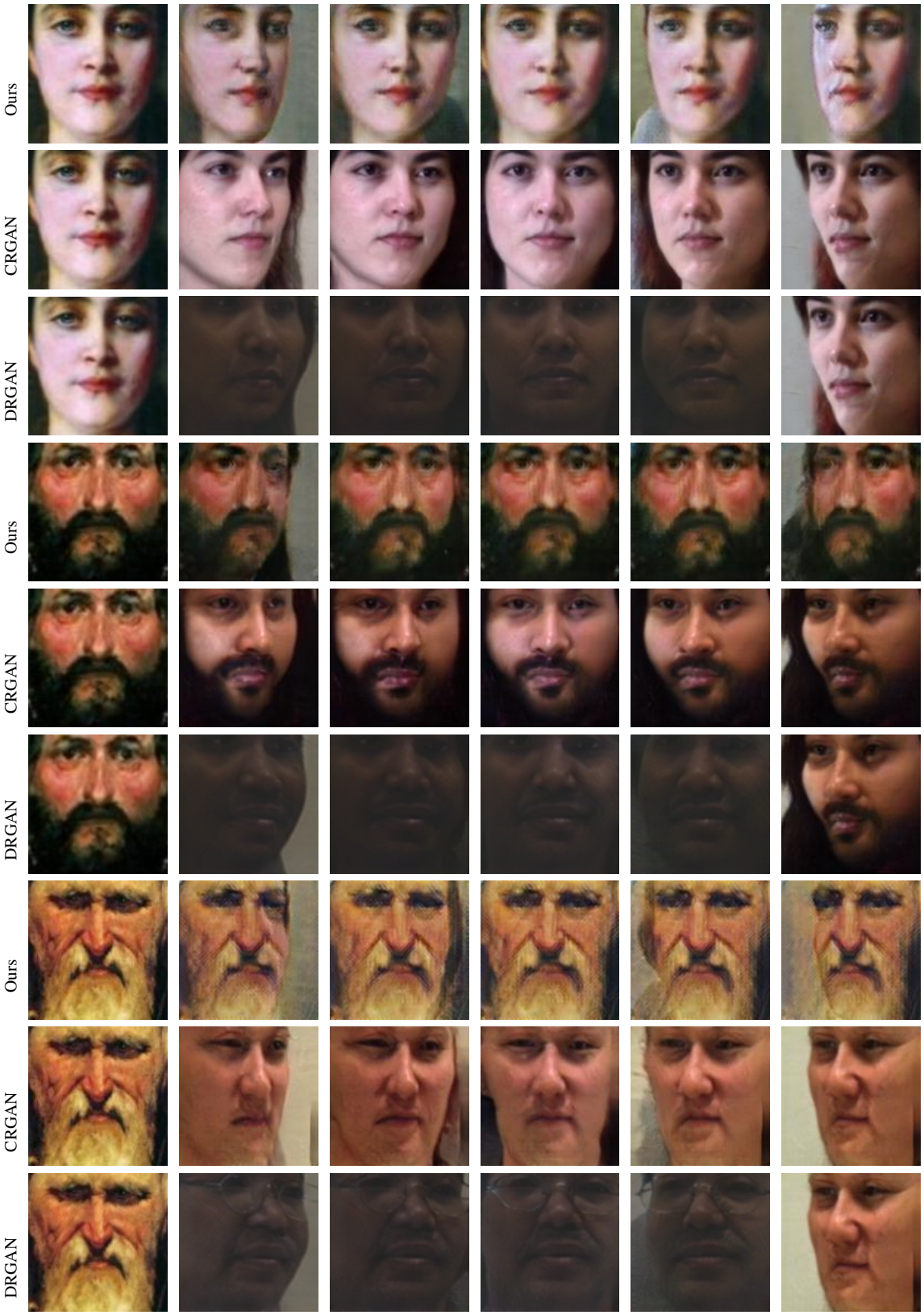


Figure 11. Additional results on cross-domain novel view synthesis.