6. Appendix

6.1. Extended experiment setups

As the training of our style manipulation network does not require any external images, the training samples are basically randomly-sampled attribute-configuration-and-$\mathbf{W}+$ pairs, which are generated on the fly during training. Specifically, the $\mathbf{W}+$ vectors are mapped from randomly-sampled gaussian noise vector $\mathbf{z} \sim N(\mathbf{0}, \mathbf{1})$ (truncated with 0.7) with the style mapping network of StyleGAN2 [14], and the attribute configurations are produced by evenly sampling the attribute values along each attribute space (e.g., $age \sim U[-30, 30]$, $smile \sim U\{0, 1\}$). The detailed attribute settings in our experiments for different datasets are listed in Table 4.

As shown in Figure 2, we employ a couple of pre-trained attribute prediction networks to supervise the training. Specifically, we employ the official pre-trained HopeNet model [36] implemented in PyTorch for pose estimation. The age estimator used for training is the pre-trained age regressor implemented in PyTorch [19]. We employ the official pre-trained CircularFace model [11] for identity embedding extraction of realistic/artistic faces. The official pre-trained VGG-19 model for comic/animal identity embedding extraction. The multi-task binary attribute classifier is a ResNet34 [9] trained by ourselves on the CelebA dataset [16] (for realistic/artistic faces), AFHQ (for animal faces), and comics dataset [2] (for comic face).

Our style manipulation network is implemented in PyTorch 1.6. It is trained with batch size of 8 on a single Tesla V100 GPU. It is optimized using Adam optimizer [31] with $\beta_1 = 0.5$ and $\beta_2 = 0.99$, and the learning rate is fixed at 10^{-4}. In all experiments, our model is trained for 50,000 steps for single-attribute manipulation (Stage I) and then 100,000 additional epochs for for multi-attribute editing (Stage II).

6.2. Extended ablations studies

Other than the dynamic architecture, L_{dmac} constraint and two-stage training procedure, we conducted more ablation studies on other architecture features and training techniques used in our approach.

Relative numeric attribute In our design, the numeric attribute (e.g., age, yaw, pitch) values on which the style manipulation network is conditioned represent the relative change of the attribute with respect to the original image, rather than the absolute attribute values. For example, if the user specified $yaw = +15$, it means increasing the yaw angle by 15° with respect to the original face (i.e., $\Delta yaw = +15$). Such relative feature for numeric attributes has proved to be more effective in encouraging precise edits. Figure 11 demonstrates if the relative attribute setting (only for numeric attributes) and the use of the contrastive losses is superior to the absolute attribute setting.

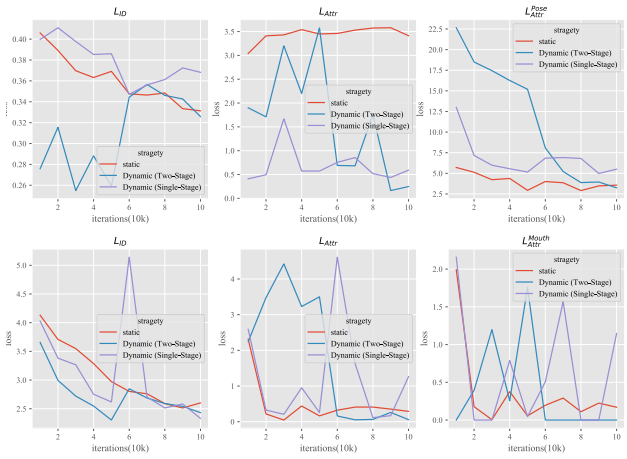

Figure 8: Comparisons of the dynamic architecture versus the static architecture, and the two-stage training versus single-stage training, experimented on FFHQ (**top**) and comic datasets (**bottom**). The validation losses demonstrate how well the model performs on the validation dataset in terms of the identity preservation and attribute control accuracy (the lower the better). As shown, as the two-stage training procedure trains for single-attribute editing in the first 50k iterations, its performance in terms of multi-attribute control accuracy at the beginning is not good. However, after 100k iterations, the two-stage training converges well and achieves the lowest L_{id} and L_{attr}.

Additional architectural details Figure 13 examines how our model benefits from the identity loss L_{id} and normalization loss L_{norm} as introduced in the method section. Figure 12 compares two variations of DyStyle architectures. In our architecture design, we condition the proxy codes on both the latent code $\mathbf{W}_l+$ and attribute specification to allow adaptive style modulation. Whereas, an alternative way is to condition the generation of proxy codes merely on the attributes. Figure 12 verifies the superiority of this feature.

6.3. Visual Comparisons between DyStyle and prior methods

We visually show some test results and demonstrate how the generated images vary by methods. As shown in Figure 4, 5, when jointly manipulating multiple attribute, the preservation of identity and control accuracy along each attribute of prior methods are problematic. Some incremental attribute editing results are demonstrated in Figure 14. As shown, StyleFlow [5] exhibits good attribute disentanglement and identity preservation as ours. Whereas, the control precision of yaw and the smoothness of change when controlling binary attributes (glasses and smile) is inferior to ours. As InterFaceGAN [22] performs linear editing of style codes, unwanted change of identity and other attributes are noticeable. StyleCLIP [18] shows good identity preservation in the process of attribute editing, but the semantic ac-

Table 4: The attribute settings in our experiments for different datasets.

	attribute name	type	value	detailed explanations
realistic/artistic face	yaw	numeric	(-30,30)	relative yaw change. "+20" means "increase yaw angle by 20°"
	pitch	numeric	(-30,30)	relative pitch change. "+20" means "increase pitch angle by 20°"
	age	numeric	(-30,30)	relative age change. "+20" means "become 20 years older"
	black-hair	binary	{0,1}	1 means having black hair
	mustache	binary	{0,1}	1 means having mustaache
	expressions	Multi-class binary	$\{0,1\}^7$	(smile, angry, disgust, fear, sad, surprise, neutral) these expressions are exclusive. 1 means having that expressions.
	glasses	binary	{0,1}	1 means having glasses
comic face	pupil color	Multi-class binary	$\{0,1\}^8$	(red, yellow, blue, green, brown, purple, black, white) these colors are exclusive. 1 means the pupil is of that color.
	hair color	Multi-class binary	$\{0,1\}^8$	(red, yellow, blue, green, brown, purple, black, white) these colors are exclusive. 1 means the hair is of that color.
	open mouth	binary	{0,1}	1 means open mouth
	hair style	Multi-class binary	$\{0,1\}^2$	(long, short). the hair styles are exclusive. 1 means the hair is of that style.
animal face	head pose	Multi-class numeric	{0,1,2}	{0: head turn left, 1: head facing front, 2: head turn right}
	young	binary	{0,1}	0 mens young, 1 means old
	open mouth	binary	{0,1}	0 means open mouth. 1 means shut off the mouth.
	close eye	binary	{0,1}	1 means close eye
	breed	Multi-class binary	$\{0,1\}^5$	the breed set vary by dog or cat. Breed types are exclusive. 1 means the cat (or dog) is of that breed.

curacy is unsatisfactory. Except for the accumulation of errors caused by static sequential editing, general CLIP [35] models may not accurately describe specific attributes. IS-FGAN [32] improves generative diversity but reduces stability due to the introduction of noise in the manipulation of latent codes. As all images are generated with the same StyleGAN2 generator, the degradation of image qualities is unnoticeable. Generally speaking, he performance gap in terms of single-attribute editing between existing methods and ours are not that noticeable as joint multi-attribute editing.

6.4. Visual experimental results

We present more attribute-controlled image generation results in Figure 15 (realistic faces), Figure 16 (artistic faces), Figure 18 (comic faces) and Figure 19 (animal faces). Some more high-resolution (1024×1024) realistic face editing results by our approach are presented in Figure 17, in the context of single-attribute manipulation and multi-attribute manipulation. Some more high-resolution (1024×1024) realistic face editing results by our approach are presented in Figure 17. With the image-to-style encoder provided in pSp [21], we also conducted attribute-conditioned editing of real photos and present the results in Figure 20.

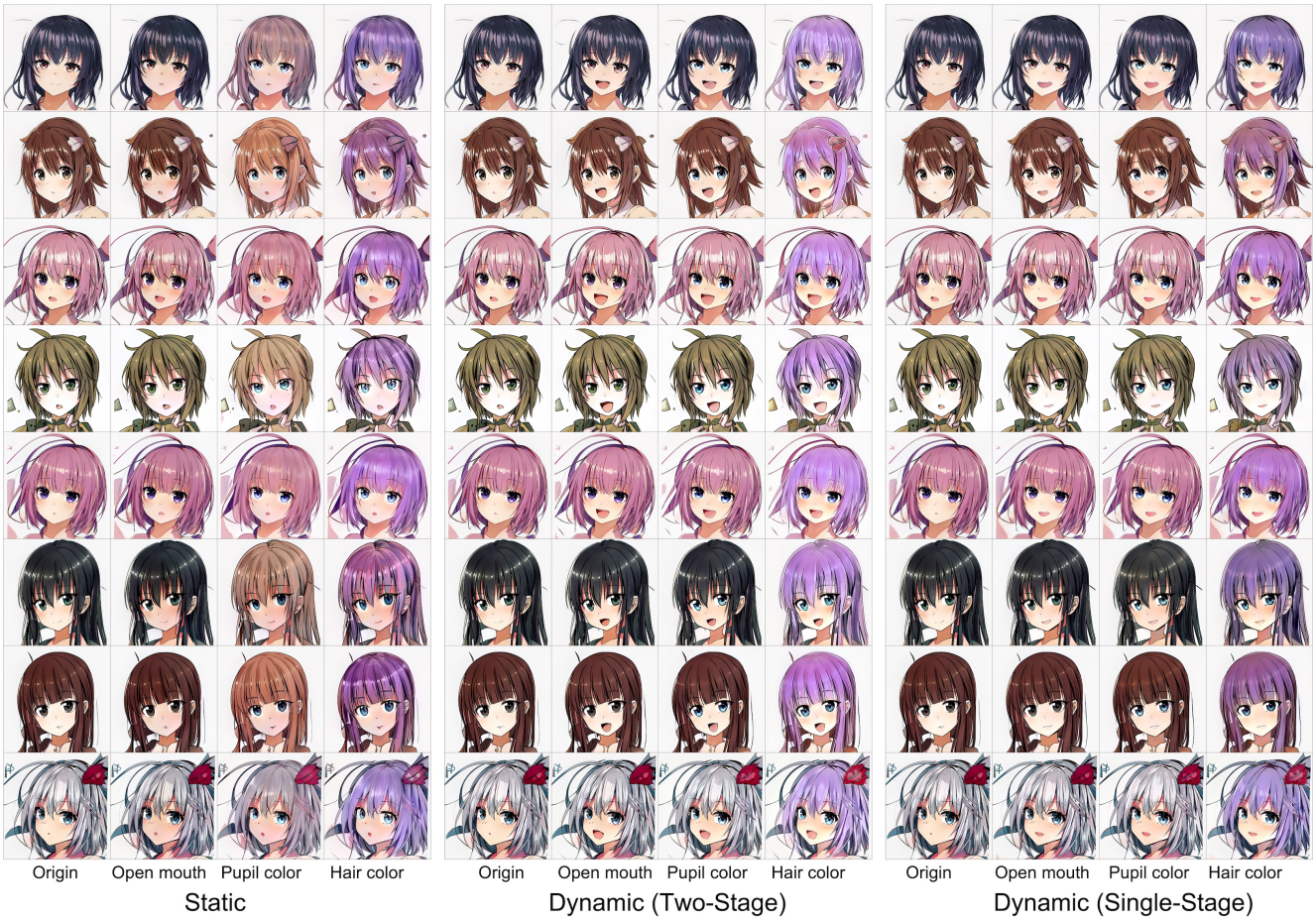

Figure 9: Qualitative comparisons of the dynamic architecture (**middle**) versus the static architecture (**left**), and the two-stage training (**middle**) versus single-stage training (**right**). As shown, static structures lead to noticeable color distortions, while single-stage dynamic structures yield weakly variable results.

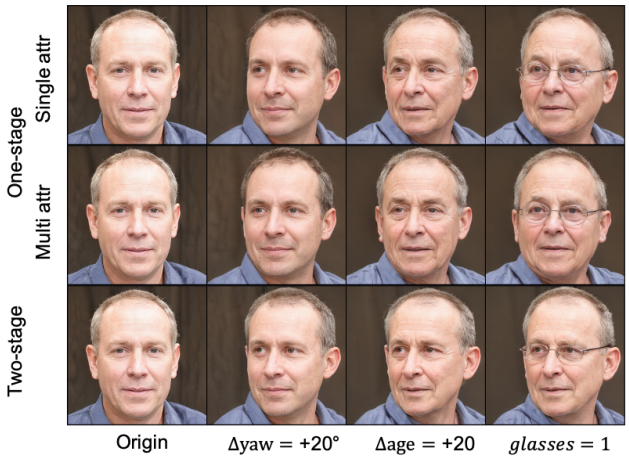

Figure 10: Ablation studies on the training strategy. We compare the results of two-stage training (single-attribute in Stage I and multi-attribute in Stage II, **(bottom)**) and one-stage training (single-attribute only **(top)**, multi-attribute only **(middle)**). As shown, the identity does not hold after multi-attribute editing **(top)** and the control of yaw is imprecise **(middle)**. The two-stage training strategy results in better editing results than one-stage training **(bottom)**.

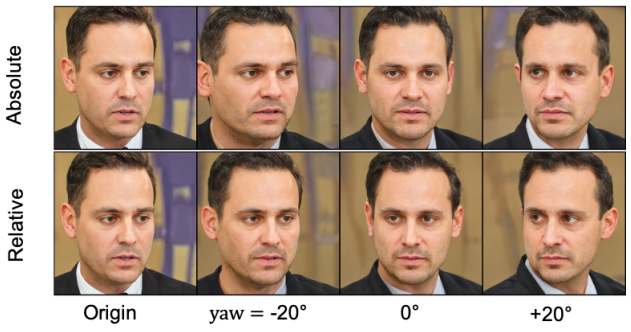

Figure 11: Comparisons of the relative attribute setting and the absolute setting. As shown, the absolute attribute setting results in unpleasant identity variation and imprecise control of head rotation along yaw.

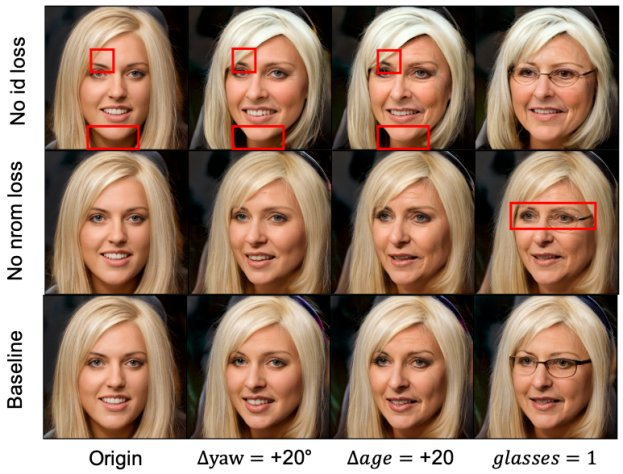

Figure 12: Ablation studies on the loss configuration. By removing the identity loss term or the normalization loss term from the full loss as in Eq 2, we retrain our model with the same hyperparameters. Without L_{id} loss, the identity variation tends to be more significant **(top)**. Without L_{norm}, the generated images are prone to fall into failure modes **(middle)**: see the regions highlighted with red boxes.

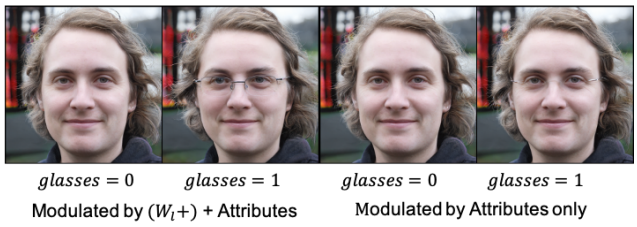

Figure 13: Ablation studies on the architecture design of the DyStyle. We compared the architecture conditioned the generation of proxy codes on the latent code and attributes (left), and that conditioned on attributes only (right). When changing the face from "no-glasses" to "with-glasses", the left model generates faithful attribute editing results while the right model is prone to fail.

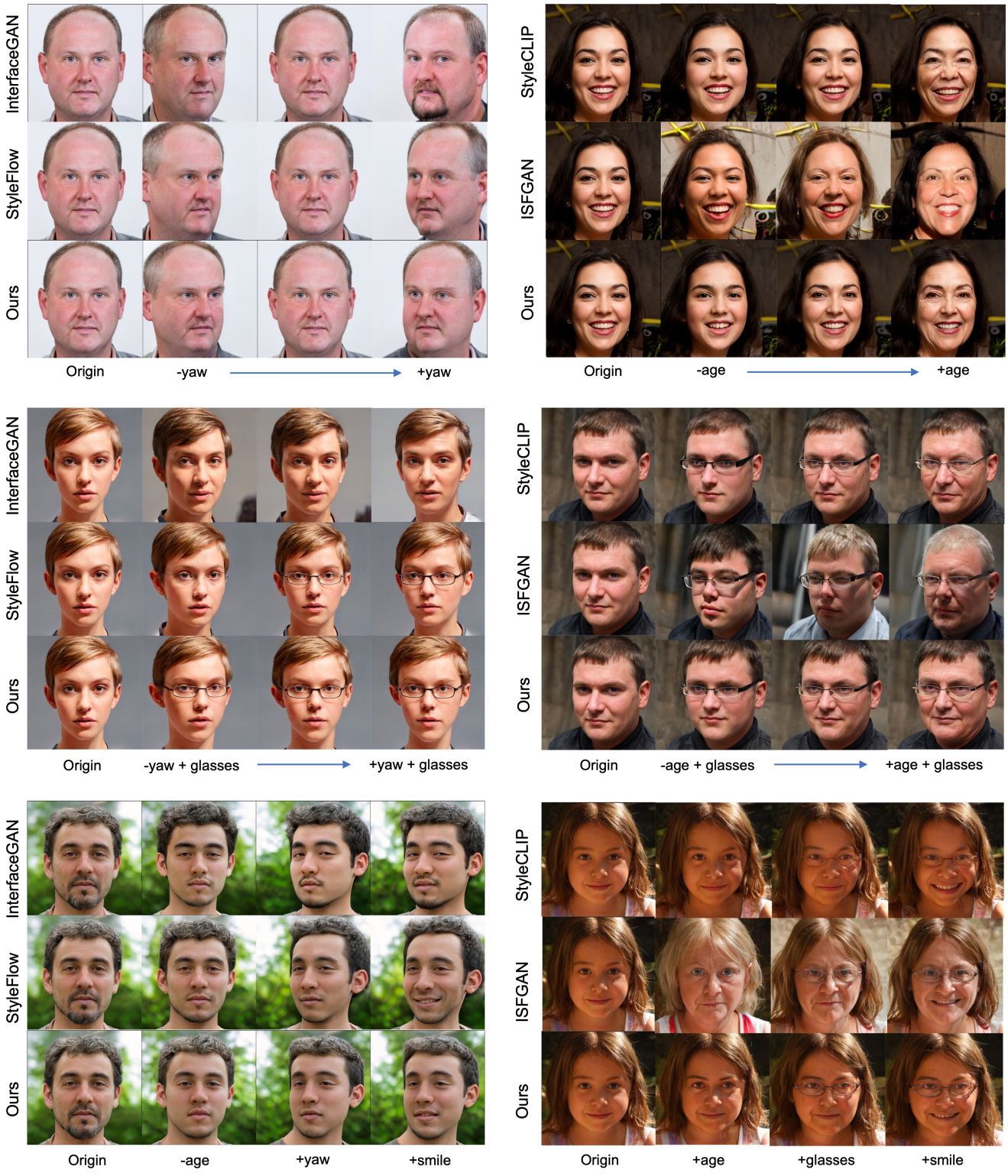

Figure 14: Comparisons of our approach and competing methods in terms of incremental attribute manipulation.

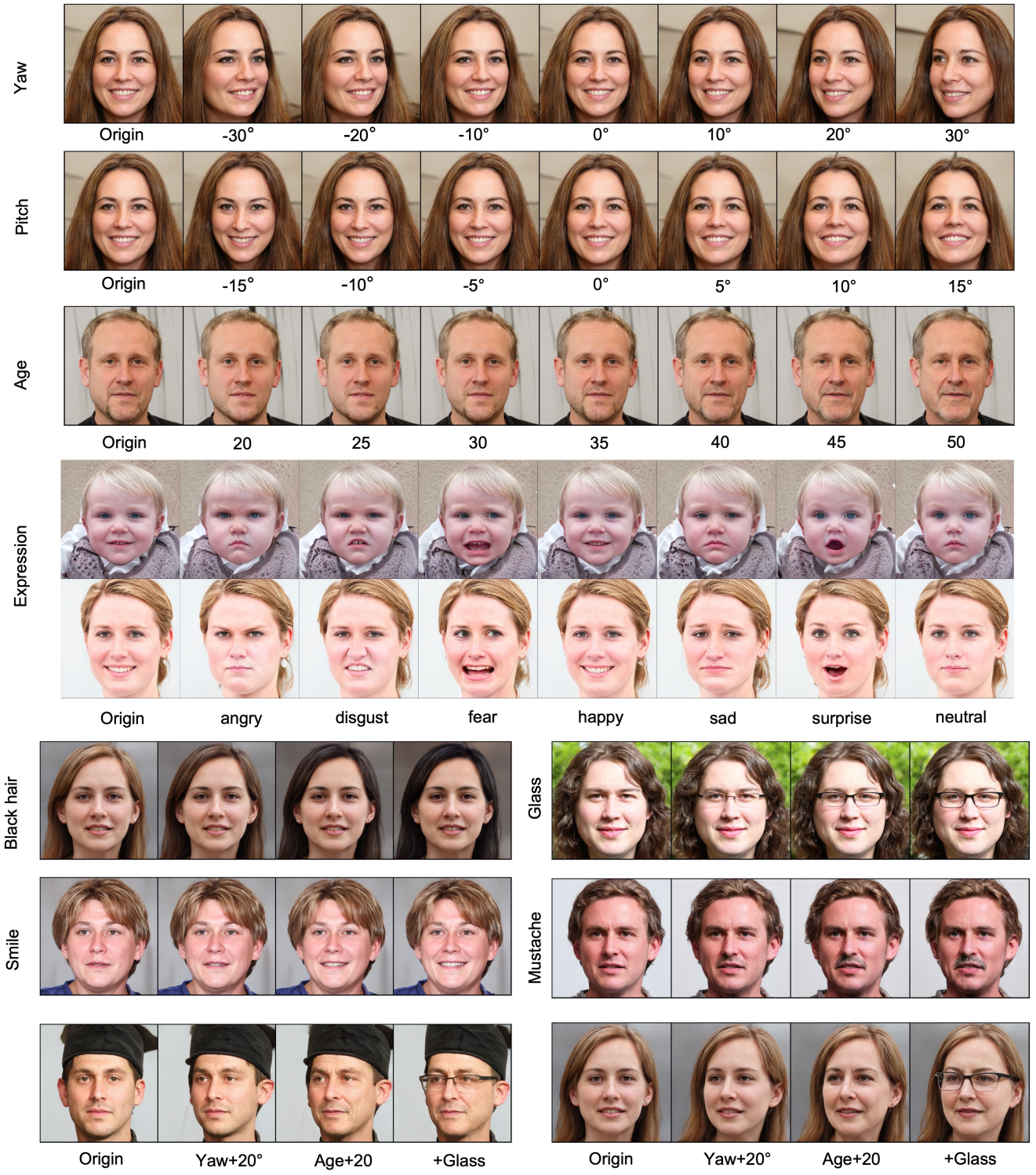

Figure 15: Results of single- and multi-attribute manipulation on realistic faces.

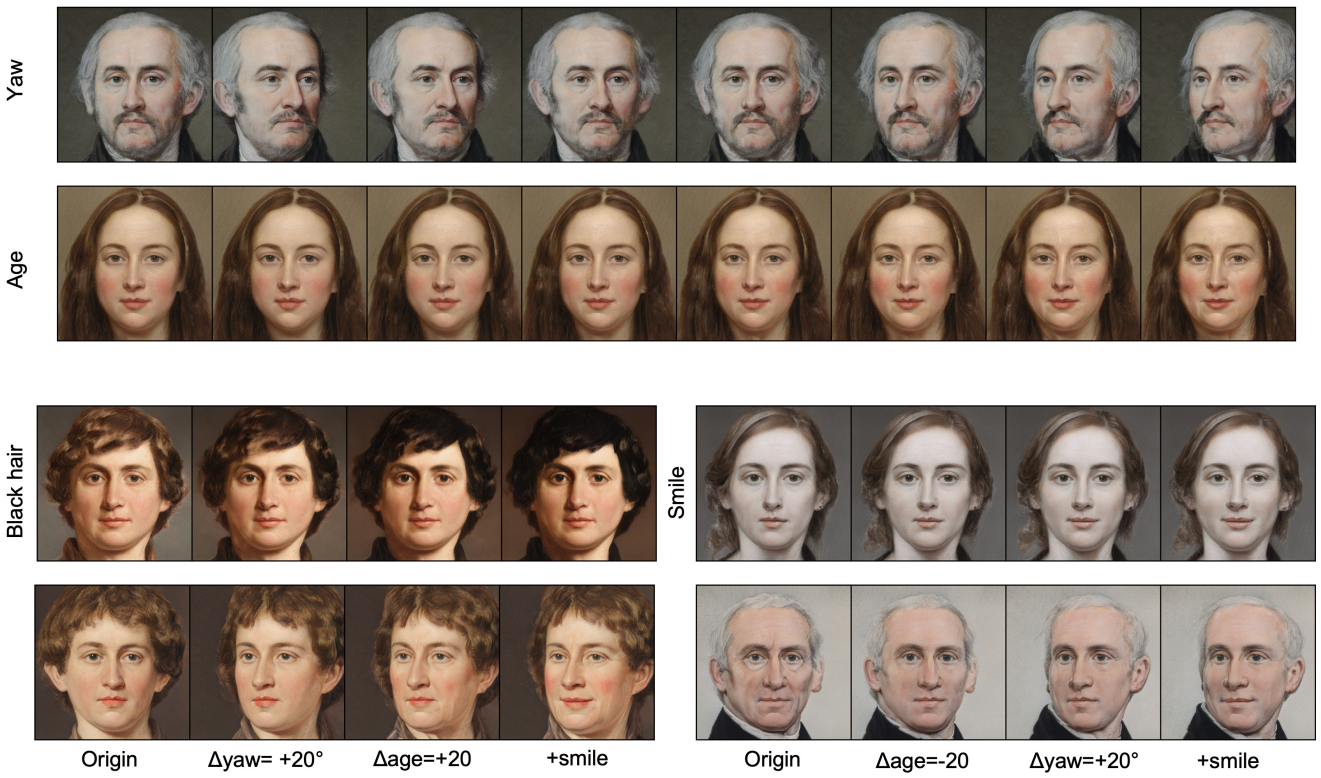

Figure 16: Results of single- and multi-attribute manipulation on artistic faces.

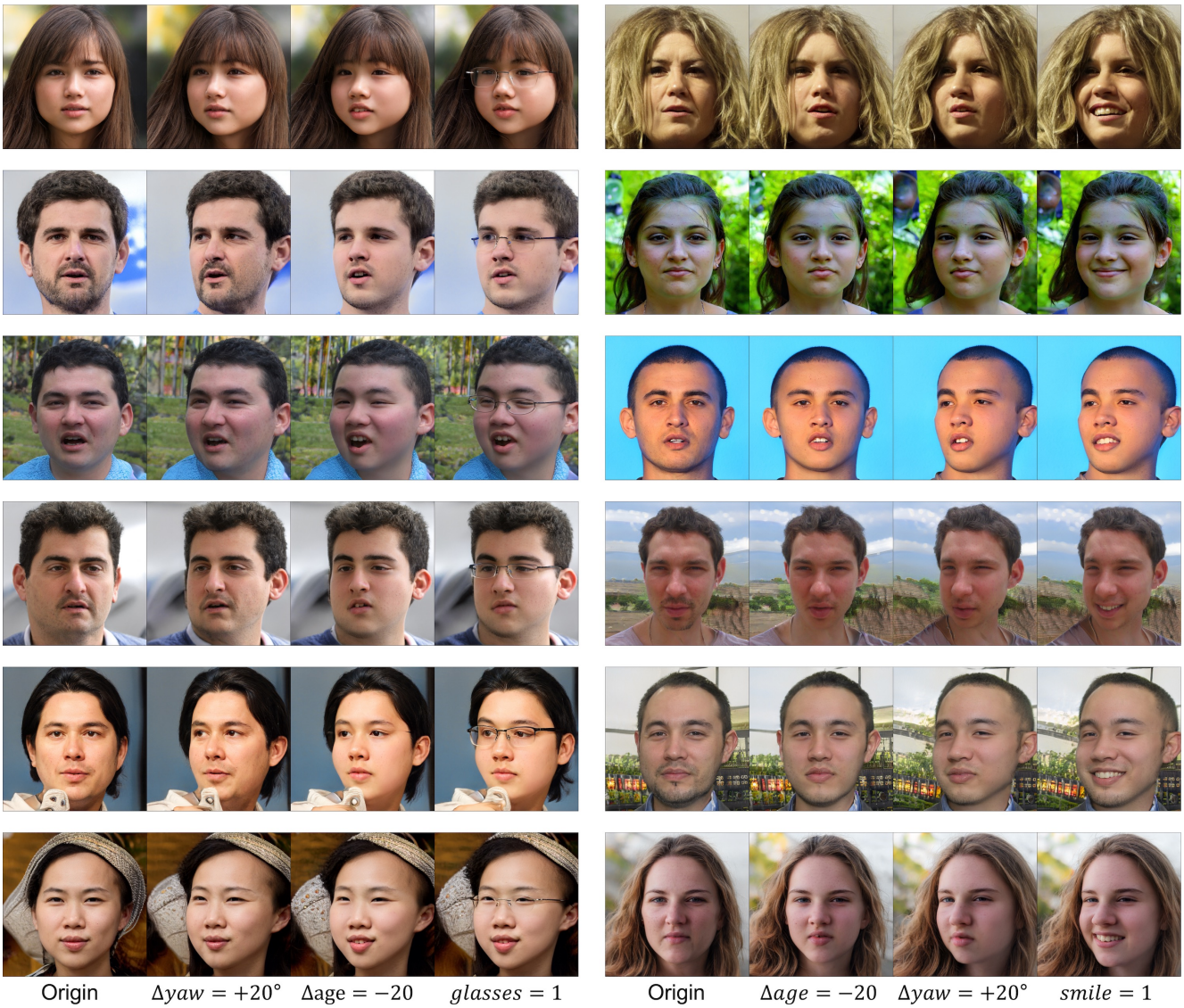

Figure 17: Results of multi-attribute manipulation on high-definition realistic faces (1024×1024).

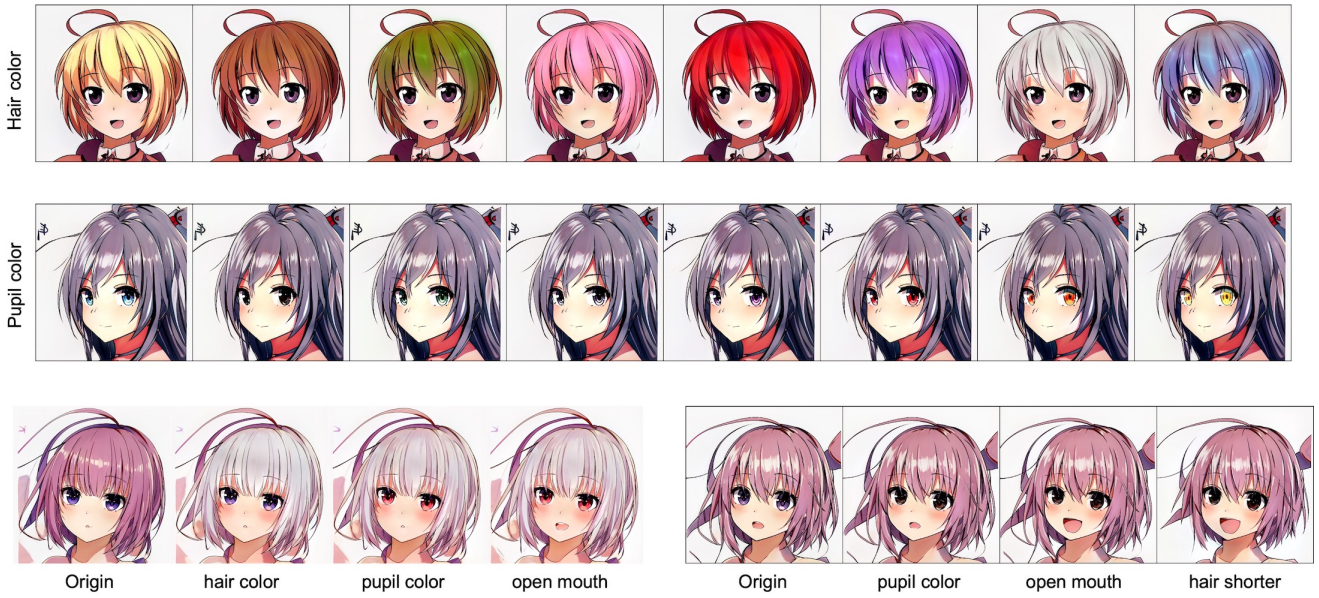

Figure 18: Results of single- and multi-attribute manipulation on comic faces.

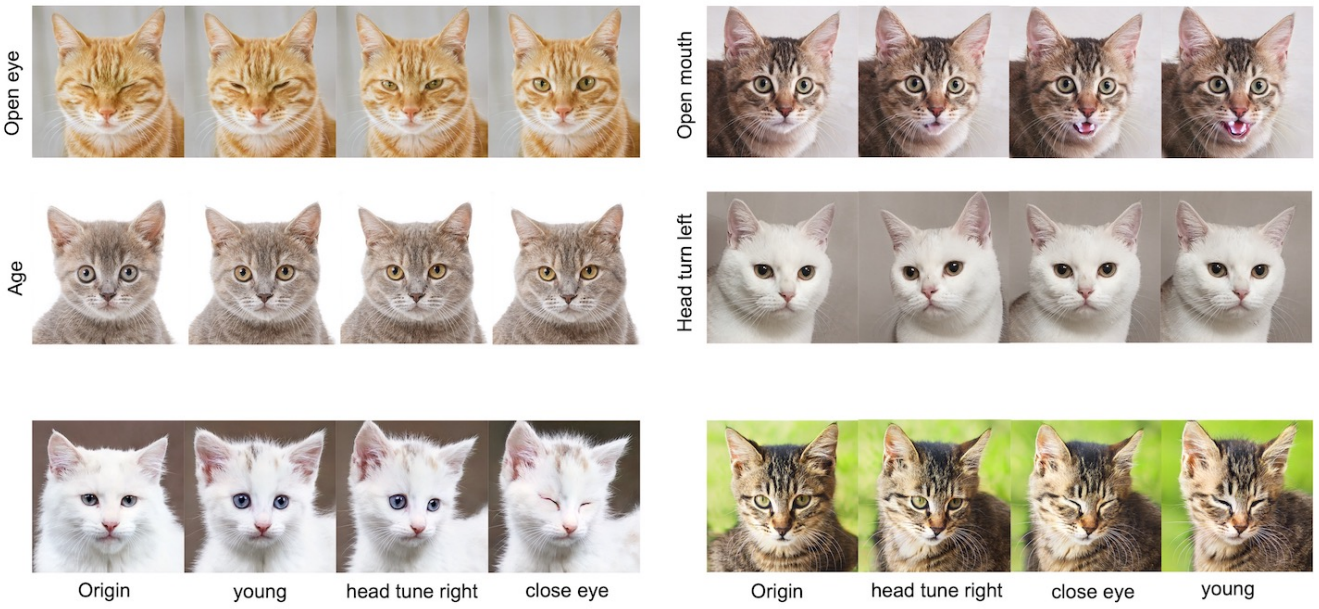

Figure 19: Results of single- and multi-attribute manipulation on animal faces.

Figure 20: Multi-attribute editing of real photographs that are reconstructed with pSp encoder.