DyStyle: Dynamic Neural Network
for Multi-Attribute-Conditioned Style Editings

Bingchuan Li*   Shaofei Cai*   Wei Liu   Peng Zhang

Qian He   Miao Hua   Zili Yi†

ByteDance Ltd, Beijing, China

{libingchuan, caishaofei, liuwei.jikun, zhangpeng.ucas, heqian, huamiao, yizili}@bytedance.com

Abstract

The semantic controllability of StyleGAN is enhanced by unremitting research. Although the existing weak supervision methods work well in manipulating the style codes along one attribute, the accuracy of manipulating multiple attributes is neglected. Multi-attribute representations are prone to entanglement in the StyleGAN latent space, while sequential editing leads to error accumulation. To address these limitations, we design a Dynamic Style Manipulation Network (DyStyle) whose structure and parameters vary by input samples, to perform nonlinear and adaptive manipulation of latent codes for flexible and precise attribute control. In order to efficient and stable optimization of the DyStyle network, we propose a Dynamic Multi-Attribute Contrastive Learning (DmaCL) method: including dynamic multi-attribute contrastor and dynamic multi-attribute contrastive loss, which simultaneously disentangle a variety of attributes from the generative image and latent space of model. As a result, our approach demonstrates fine-grained disentangled edits along multiple numeric and binary attributes. Qualitative and quantitative comparisons with existing style manipulation methods verify the superiority of our method in terms of the multi-attribute control accuracy and identity preservation without compromising photorealism.

1. Introduction

Recent development in Generative Adversarial Networks (GANs) has provided a new paradigm for realistic image generation. As one of the most celebrated GAN frameworks, StyleGAN and a series of upgraded works [13, 14, 15], can produce diverse and high fidelity images with unmatched photorealism. Due to the introduction of scale-disentangled latent space, StyleGAN provides pos-
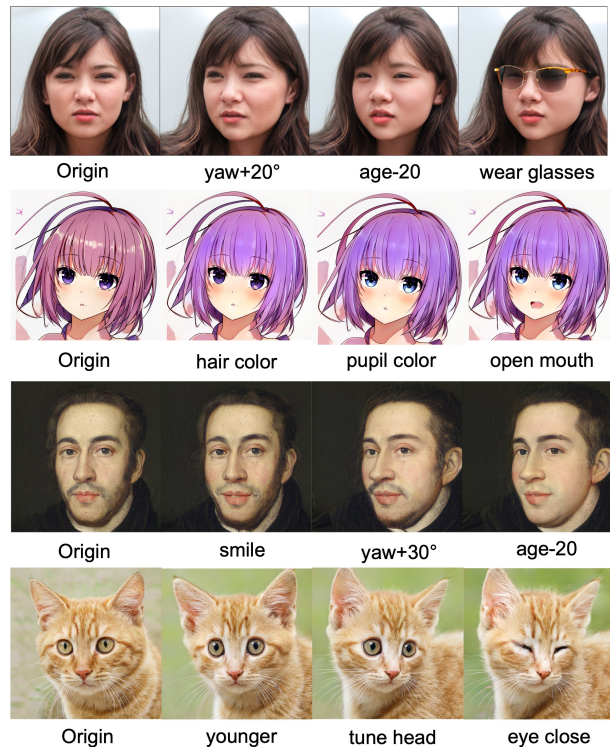

Figure 1: The multi-attribute-conditioned image editing results with our approach, achieved by manipulating the latent codes of four different pre-trained StyleGAN2 models.

sibilities for flexible and controllable manipulation of attributes.

Thus, one stream of research is to manipulate the latent codes of unconditional GANs, without retraining the generator. By operating the code to walk in the StyleGAN latent space, resulting in continuous changes of different degrees of a single attribute. Manipulating individual attribute of generated images has yielded satisfactory results [22, 5, 25]. However, performing edits along one attribute may lead to

---

*Co-first author
†Corresponding author

unexpected changes in other semantics due to entanglement between different semantics in the latent space. Prior methods always produce semantic confusion when jointly manipulating multiple attributes. We analyzed several reasons: 1. The static structure of single-attribute sequential editing is prone to error accumulation. E.g. InterfaceGAN [22], the identity similarity drops sharply as the number of edited attributes increases; 2. The static structure of multi-attribute parallel [32, 5] input fails to adapt to random combinations of attributes during inference. Since the structure and parameters are fixed after training, all attributes must be calculated forward regardless of whether they are selected or not; 3. Training parallel static structures requires simultaneously optimizing all attributes in one forward pass, which makes learning disentangled latent space more difficult.

We argue that the style editing network should be able to adapt to wide varieties of attribute configurations, and when training the style editing network for multiple attribute manipulation, any biases of the distribution of attribute configurations would easily result in systematic control errors. Take realistic portrait editing as an example, for one case we only change the hair color of the portrait. For another case, we change both the hair color and age of the portrait. We wish the two attribute configurations are sampled with equal probability and are both well handled by the style editing network. We made two efforts to address this issue, we evenly sample the attribute configurations during training instead of using static set of training samples. To make this possible, we employ pre-trained knowledge networks to provide "on-the-fly" supervisions rather than using static labels. In addition, we employ a dynamic style editing network consisting of multiple experts, each of which is responsible for the manipulation of one attribute. We dynamically activate a subset of the experts based on whether the corresponding attributes are edited or not. That means, the structure and parameters of the style editing network vary by different input samples. Experiments show that the Dynamic Style Manipulation Network (DyStyle) that processes each sample with data-dependent architectures and parameters can well adapt to various types of attribute configurations.

Although dynamic networks are usually more flexible than static networks, they also require appropriate optimization methods to play its performance. Therefore, we propose dynamic multi-attribute contrastive learning (DmaCL) methods cooperate with better optimization of DyStyle. In general, we perform contrastive constraints from the image space and latent space of model. For generated images, we employ a variety of pre-trained classifications or regressions to capture related attribute features. Dynamic multi-attribute contrastor are proposed to ensure the other attributes of the image are unchanged when optimizing activated attributes. Meanwhile, we propose dynamic multi-

attribute contrastive loss to constrain the embeddings extracted by the expert network. It can perceive and discriminate the correlation and difference between the activated attributes to ensure disentanglement during training. To make the training of the DyStyle network easier, we adopt a novel easy-to-hard training procedure in which the DyStyle network is trained for editing a single attribute at a time, and then trained for jointly manipulating multiple randomly-sampled attributes. Generally speaking, our contributions include:

- A Dynamic Style Manipulation Network (DyStyle) is carefully designed to perform multi-attribute-conditioned editing of the StyleGAN latent codes, implementing adaptability to wide varieties of attribute configurations.

- We propose the Dynamic Multi-Attribute Contrasive Learning (DmaCL) method, which is applied to fully optimize the dynamic neural network to realize the disentanglement of image space and latent space.

- Comprehensive evaluations on various datasets (realistic faces, comics, artistic portraits, animal faces) demonstrate improved attribute control accuracy and better identity preservation of our approach over existing static architectures. The improvements are more significant when jointly manipulating the styles along multiple attributes.

2. Ralated Work

2.1. Unconditional GANs

Generative Adversarial Network (GAN) is first introduced by Goodfellow et al. [8], and has been one of the most active fields in deep neural networks. One research direction is to improve the GAN architectures, loss functions, and training dynamics for improved quality, diversity and stability of training. In terms of the GAN architectures, DCGAN [20], ProgressiveGAN [12], BigGAN [6] and StyleGAN series [13, 14, 29, 30] architectures are the top known architectures in history of development. We build our work on StyleGAN2 [29], as it achieves the best photorealism.

2.2. Attribute Conditional GANs

Conditional GANs [33] have given rise to many image manipulation applications. Unlike unconditional GANs that take random noises as input, conditional GANs take meaningful priors (e.g., attribute [32, 5], text [18, 40, 25], 3D model parameters [23], sketches [41, 39], label maps [34, 38]) as input and synthesizes relevant images, thus offering users a certain level of control.

Some of unsupervised methods [28, 37] uses Principal Component Analysis (PCA) to learn the most important directions. Although these methods can find some principal
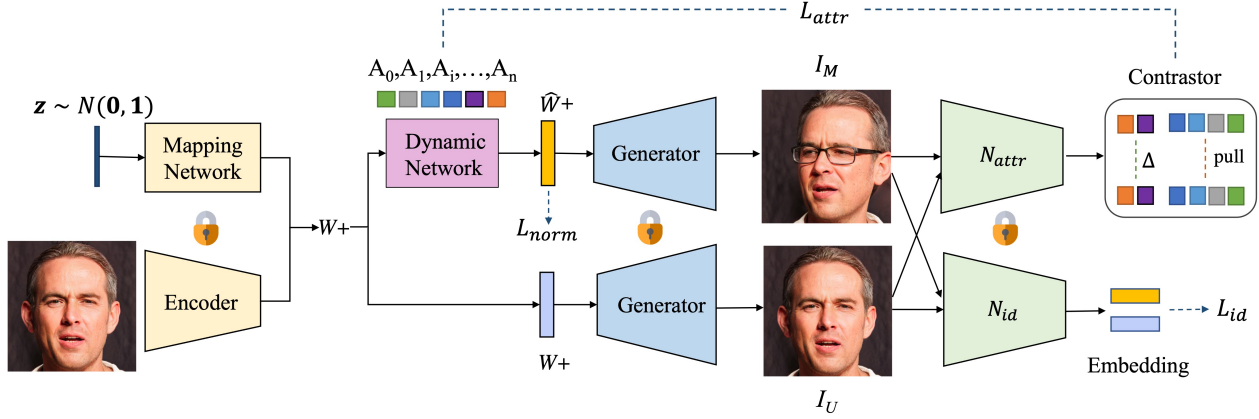
Figure 2: The framework and training losses of our multi-attribute-conditioned style editing approach. The Dynamic Network is trained for attribute-conditioned image editing while the StyleGAN2 Generator, the Style Mapping Network and the Encoder are held fixed.

components with clear semantics, they need to be artificially defined. Unlike unsupervised methods, supervised attribute conditional GANs usually associate attribute labels to GAN latent space to manipulate image generation. InterFaceGAN [22] uses labeled data to learn a SVM to discover the separation plane and the directions of certain attributes. StyleRig [23] employs a 3D morphable face models (3DMMs) [1], and a rigging network to map 3DMM's semantic parameters to StyleGAN's input. Although StyleRig generates very nice results for the manipulation of head pose and illumination, the detailed control of other facial attributes did not work. StyleFlow [5] seeks continuous and nonlinear normalizing flows in the latent space conditioned by attribute features. However, the training of StyleFlow does not explicitly enforce the control accuracy of attributes, and the assumption of normal distribution with respect to any attribute configuration does not always hold due to data biases. StyleCLIP [18] introduces text-aware CLIP [35] to generative models, thereby enriching manipulable semantics. ISFGAN [32] proposes an implicit style function to straightforwardly achieve multi-modal and multi-domain image-to-image translation from pre-trained unconditional generators.

3. Method

3.1. Framework

As shown in Figure 2, our method manipulates the extended latent code $\mathbf{W}+$ using a Dynamic Style Manipulation Network (DyStyle). The extended latent code $\mathbf{W}+$ consists of 18 different 512-dimensional vectors, one for each input layer of StyleGAN2 generator [14]. The $\mathbf{W}+$ can be either mapped from a random Gaussian noise vector $\mathbf{z} \sim N(\mathbf{0}, \mathbf{1})$ with the Style Mapping Network of StyleGAN2 [14] or embedded from a real photograph with the image-

to-style encoder of pSp [21] or E4E [24]. The DyStyle network takes an attribute specification and $\mathbf{W}+$ as inputs and predicts a manipulated latent code $\hat{\mathbf{W}}+$. The attribute specification **Attr** is made up of a set of attribute values specified by the user, defining the appearance of the desired image. The attribute set, in our experimental setting, includes numeric attributes (e.g., yaw, pitch of the head pose, age) and binary attributes (e.g., glasses, smile, black hair, mustache, close eye, open mouth). The number of attributes can be expanded without modifying the framework.

The manipulated latent code $\hat{\mathbf{W}}+$ is fed to the Style-GAN2 generator to generate the corresponding manipulated image $\mathbf{I}_M$. In the meantime, the original latent code $\mathbf{W}+$ is mapped to the untouched image $\mathbf{I}_U$. $\mathbf{I}_M$ is expected to maintain the identity of $\mathbf{I}_U$, while matching the target attribute specification. Dynamic multi-attribute contrastor here is utilized to learn the disentanglement between activated and inactive attributes. Such constraint is enforced with the pre-trained attribute predictors N_{attr} and a pre-trained identity recognition model N_{id}.

3.2. Dynamic network architecture

The architecture of DyStyle is shown in Figure 3. As shown, the DyStyle network manipulates each $\mathbf{W}_l+$ code separately, by taking the attribute configuration and $\mathbf{W}_l+$ as input and predicting the proxy code P which is further used to linearly modulate the $\mathbf{W}_l+$ code itself. By conditioning the proxy code jointly upon the attribute configuration and $\mathbf{W}_l+$, rather than the attribute configuration solely, the network is able to predict the proxy code adaptively for each input case of $\mathbf{W}+$, rather than generate a uniform modulation parameters for all cases of $\mathbf{W}+$. The DyStyle network employs multiple experts to process the attributes separately before they fused with cross-attention and element-wise addition. The proxy codes extracted by activated expert are
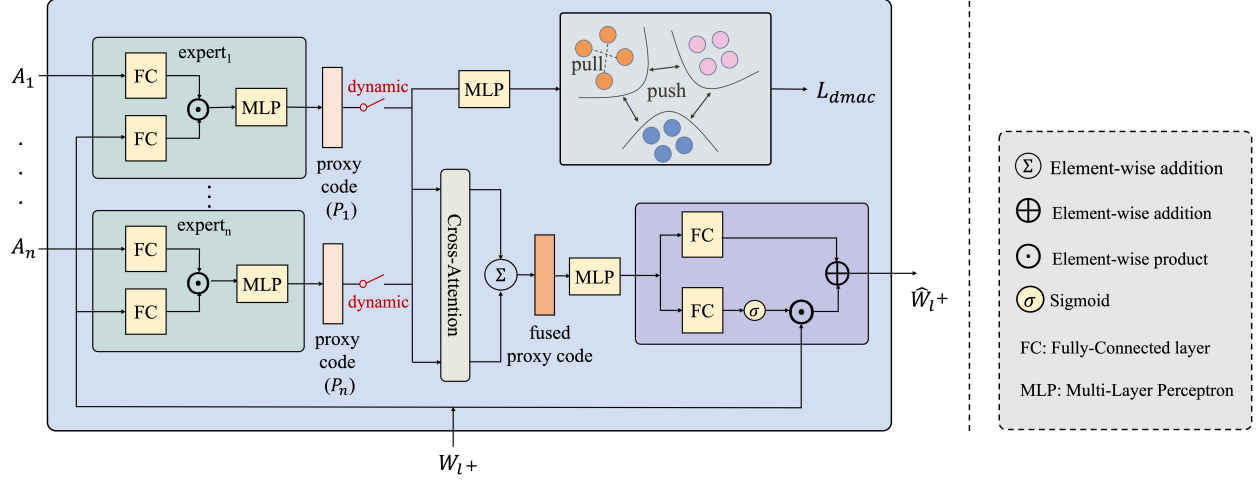
Figure 3: The architecture of our Dynamic Style Manipulation Network (DyStyle). $\mathbf{W}_l+$ ($l \in \{0, 1, ..., 17\}$) is one of the 18 vectors of $\mathbf{W}+$. Multiple experts are employed, in which each expert is responsible for the processing of one attribute before they are joined. Whether an expert is activated is based on whether the corresponding attribute is intended for editing or not.

encoded into a unified latent space, so that the dynamic multi-attribute contrastive loss is performed to constrain the same attribute aggregation and different attributes to separate from each other. Cross-attention allows different experts to communicate with each other and enables an expert adapt to the unexpected influences caused by the edits of other attributes. This argument is well supported by the ablation studies: see the Sec. 6.2 for more details. Addtionally, the cross-attention module is well suited for a variable number of attributes, which is an important feature of our dynamic architecture.

We join the features after attribute-specific processing by cross-attention and element-wise addition. Such a design favors disentangled attribute editing and improved control precision. The cross attention is computed as Eq. 1.

$$P_i = \sum_j V_j \odot \frac{\exp(Q_i \cdot K_j)}{\sum_j \exp(Q_i \cdot K_j)} \quad (1)$$

where Q_i, K_i, V_i $i \in \{1, 2, ..., n\}$ are the query, key and value vectors computed from the proxy code P_i with an FC layer respectively. As an extreme case, when $n = 1$, the output of the cross-attention $P_i = V_i$.

3.3. Dynamic multi-attribute contrastive learning

The DyStyle network is trained with an objective consisting of 4 types of losses, which is defined as

$$L = \alpha_{attr} L_{attr} + \alpha_{dmac} L_{dmac} + \alpha_{id} L_{id} + \alpha_{norm} L_{norm} \quad (2)$$

where L_{attr} is the attribute loss for various attributes (e.g., pose, age, black hair, glasses, smile), which are controlled by a dynamic multi-attribute contrastor and used to enforce the consistency of target attributes and measured attributes of the manipulated image $\mathbf{I}_M$. L_{dmac} is designated to disentangle the activated attributes in the latent space of the

model. L_{id} is the identity loss intended to preserve the identity of the original image, while L_{norm} is the normalization loss discouraging degradation of image quality. α_{attr}, α_{dmac}, α_{id} and α_{norm} are the corresponding coefficients for each loss term.

3.3.1 Dynamic multi-attribute contrastor

Similar to some weakly supervised methods [3, 18], we employ pre-trained models to extract attribute information. Since the training mode of the framework relies on dynamic attribute input, the goal of dynamic multi-attribute contrastor is to ensure the generated image after manipulated contains activated attributes, while the inactive attributes remain consistent with randomly generated images. The form of L_{attr} that relies on a set of contrastors differs for numeric attributes and binary ones. We design specific losses for different types of pre-trained estimators.

Specifically, the contrastive loss for numeric attribute A^k is defined as

$$L_{attr}^{A^k} = \begin{cases} |A_M^k - A_U^k|, \text{if } \Delta_{A^k}^{gt} \text{ is none} \\ \max(|A_M^k - A_U^k - \Delta_{A^k}^{gt}| - T_{A^k}, 0), \text{otherwise} \end{cases} \quad (3)$$

where A_M^k, A_U^k are attribute value of Face $\mathbf{I}_M$, $\mathbf{I}_U$ measured by a pre-trained attribute estimation network N_{A^k}. $\Delta_{A^k}^{gt}$ is the ground-truth pose variation of $\mathbf{I}_M$ to $\mathbf{I}_U$, specified at the input of DyStyle. If $\Delta_{A^k}^{gt}$ is none, it means A^k is inactive. T_{A^k} are constant thresholds, which is set to 3 for yaw and pitch, and 5 for age.

For binary attributes, the input attribute values are either 0 or 1, and they represent the status of the target attributes (1 means "with" and 0 means "without"). We employ a pre-trained multi-task or multi-class network (N_{A^k}) to predict

all the binary attributes of an image, thus $\mathbf{A}_M^k = N_{A^k}(\mathbf{I}_M)$, $\mathbf{A}_U^k = N_{A^k}(\mathbf{I}_U)$.

The attribute loss of the binary attributes is written as

$$L_{attr}^{A^k} = \begin{cases} 1 - \frac{\mathbf{emb}_M^{A^k}}{\|\mathbf{emb}_M^{A^k}\|} \cdot \frac{\mathbf{emb}_U^{A^k}}{\|\mathbf{emb}_U^{A^k}\|}, \text{if } \mathbf{A}_{gt}^k == \mathbf{A}_U^k \text{ or } \mathbf{A}_{gt}^k \text{ is None} \\ -\sum_{A^k} [A_{gt}^k \log A_M^k + (1 - A_{gt}^k) \log(1 - A_M^k)], \text{otherwise} \end{cases}$$
(4)

where $\mathbf{emb}_M^{A^k}$ (or $\mathbf{emb}_U^{A^k}$) is the activation of the second last layer of the pre-trained multi-attribute predictor N_{A^k} given $\mathbf{I}_M$ (or $\mathbf{I}_U$) as input. $\mathbf{A}_{gt}^k$ is the ground-truth (or target) attribute specification. The similarity score of $\mathbf{emb}_M^{A^k}$ and $\mathbf{emb}_U^{A^k}$ is enforced to 1 when no edits are intended. Further, the cross-entropy loss is calculated separately for each binary attribute, and finally summed up.

The original face $\mathbf{I}_U$ and the manipulated face $\mathbf{I}_M$ are expected to have the same identity. Therefore, the identity loss are computed as

$$L_{id} = 1 - cos\_similarity(\mathbf{emb}_M^{id}, \mathbf{emb}_U^{id}) \qquad (5)$$

where $\mathbf{emb}_M^{id}$ (or $\mathbf{emb}_U^{id}$) is the feature embedding of the face in $\mathbf{I}_M$ (or $\mathbf{I}_U$), extracted with a pre-trained face recognition model. When the editing targets are not realistic faces, the conventional face recognizer does not serve as an effective identity representation. In this cases (e.g., comics or animal faces), we employ LPIPS loss [27] as the identity loss.

Practically, we choose $\alpha_{yaw} = 0.05$, $\alpha_{pitch} = 0.05$, $\alpha_{age} = 0.02$, $\alpha_{A^k} = 1.0$ (if A^k is binary attribute), $\alpha_{id} = 1.0$ and $\alpha_{norm} = 0.001$ in our experiments. Not that the hyperparameters are adjusted to scale the different types of losses to the same magnitude.

3.3.2 Dynamic multi-attribute contrastive loss

The proposed dynamic model will randomly select attributes for activation in a training iteration. The dynamic multi-attribute contrastor can capture the difference between the active and inactive attributes during training. However, a variety of attributes come from different pre-trained estimators. It is difficult for the contrastor to disentangle the attributes that are activated at the same time. Therefore, we apply contrastive loss L_{dmac} on the simultaneously activated attributes in the latent space of the model. The principle of the L_{dmac} is to ensure that the editing directions of a single attribute is fully gathered, while increasing difference between the editing directions of multiple attributes as much as possible.

As shown in Figure 3, attributes are independently encoded by experts as proxy codes, which contains the direction of attribute changes. In order to perform contrastive learning, activated proxy codes are mapped into the unified latent space. Here, the encoder simply applies a Mul-

tilayer Perceptron (MLP). We calculate the attribute auto-correlation and cross-correlation respectively. For auto-correlation, we calculate it on the batch dimension which the similarity measured between attributes by dot product. The formula is

$$I_{ac}^k = \sum_{i,j \in N_b, i \neq j} S_i^k \cdot S_j^k \qquad (6)$$

where N_b is the training batch size, and S_i^k represents the i-th dimention of the k-th attribute code in the unified latent space. For the cross-correlation coefficient, the attribute codes other than itself are regarded as negative samples, so the formula is

$$I_{cc}^k = \sum_{q \in N_k, q \neq k} S^k \cdot S^q \qquad (7)$$

where N_k represents that k attributes are activated by the current dynamic network. The form of our function L_{dmac} follows InfoNCE [17], which is calculated as

$$L_{dmac} = \sum_{k \in N_k} -\log \frac{\exp\left(I_{ac}^k\right)}{\exp\left(I_{ac}^k\right) + \exp\left(I_{cc}^k\right)} \qquad (8)$$

Note that each activation attribute is traversed in a single training iteration to calculate the loss sum. The form of L_{dmac} can be easily realized using softmax cross-entropy loss.

For $\hat{\mathbf{W}}_l+$ output by the dynamic network, the normalization loss is defined as

$$L_{norm} = \sum_l \|(\hat{\mathbf{W}}_l+) - \mathbf{W}_{avg}\| \qquad (9)$$

where $\mathbf{W}_{avg}$ is the statistic center of the $\mathbf{W}$ space of the pre-trained StyleGAN2 generator [14], and $\hat{\mathbf{W}}_l+$ is the manipulated style vector corresponding to the l-th layer. As discussed in [21], being closer to $\mathbf{W}_{avg}$ means higher expected quality of the generated image.

3.3.3 Two-stage training strategy

The DyStyle is trained with randomly sampled $\mathbf{W}+$ codes with the Style Mapping Network and evenly-sampled attribute configurations. The training is conducted by following a two-phase procedure. In this first stage, the network is trained for single-attribute manipulation, by randomly choosing an attribute for editing and evenly sample the target attribute value. That means, only an expert (or branch) for the edited attribute is activated at a time in this phase. Note that the loss is not changed. This phase allows each expert to focus on one attribute at a time and get used to easy editing cases. In the second stage, the DyStyle network is trained to adapt to situations when multiple attributes are manipulated jointly. In this phase, a combination of attribute set are randomly sampled and set as the
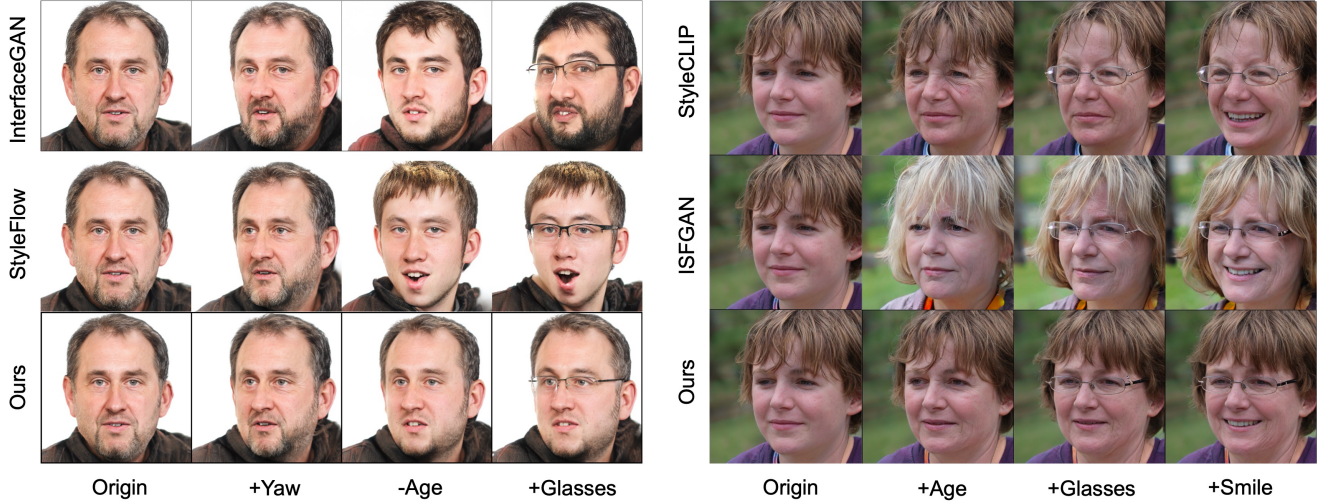
Figure 4: Comparisons of our method and competing methods in terms of multiple-attribute manipulation. As shown, the compared methods saw significant identity variation and semantic entanglement.

input of the DyStyle network, so that the experts learn to communicate with each other and adapt to more complex attribute configurations.

4. Experimental Results

4.1. Experiment setups

We experimented our style manipulation network on four pre-trained StyleGAN2 models, the one trained on FFHQ dataset [13] for realistic face generation, the one trained on MetFace [15] for artistic face synthesis, that trained on AFHQ [7] for animal face synthesis and the one trained on comics dataset [2] for comic face generation. The expanded experimental setups are reflected in the supplementary materials.

Various attribute manipulation results of our approach are presented in Figure 1. Some more attribute-controlled image generation results for each datasets are presented in Figure 15 (realistic faces), Figure 16 (artistic faces), Figure 18 (Comics) and Figure 19 (animal faces) in the supplementary materials. Some more high-resolution (1024×1024) realistic face editing results by our approach are presented in Figure 17 in the supplementary materials. With the image-to-style encoder provided in [21], we also conducted attribute-conditioned editing of real photos and present the results in Figure 20 in the supplementary materials.

4.2. Comparisons

To verify the effectiveness of the proposed method, we compared our approach with the state-of-the-art style manipulation methods including InterFaceGAN [22], Style-Flow [5], StyleCLIP [18] and ISFGAN [32]. To assure fair comparisons, these methods are designated to manipulate

the latent space of the same pre-trained StyleGAN2 model which was trained on FFHQ dataset [13]. According to the common attributes of the official open source of different methods, the comparisons only account for the editing of four attributes (yaw, age, glasses and smile). StyleCLIP [18] and ISFGAN [32] are compared separately due to their lack of ability to edit poses. As the separation plane of Inter-FaceGAN only specifies the directions of attribute control, we toughly estimated the physical meaning of scales based on a few labeled examples. Note that StyleCLIP [18] does not participate in quantitative evaluation, as CLIP-encoded [35] textual information cannot be quantified consistently across scales.

With the same set of 5000 attribute-configuration-and-$\mathbf{W}+$ pairs as test set, we conduct quantitative evaluations on these methods. Specifically, we evaluate the accuracy (or precision) of attribute control, preservation of identity and image quality respectively with well-defined metrics. To evaluate the precision of attribute control, we employ the pre-trained attribute predictors to predict the attribute labels of manipulated images and then compare them with target labels. For assure fairness, we employ a different set of attribute predictors that are excluded from those used for training. Specifically, the pose estimator is the official pre-trained model from [26], and the age estimator is officially provided by [4]. The glasses and smile classifier is a multi-task ResNet50 classifier [9] trained by ourselves with CelebA dataset [16]. With the predicted labels, we compute the Mean Absolute Error (MAE) of yaw and age, and the classification accuracy of "glasses" and "smile" attributes. As for identity preservation, we calculate the average cosine similarity score of manipulated faces and original ones: see Table 1. In terms of image quality, we evaluate the dis-

Table 1: Quantitative comparisons of different attribute-conditioned style editing approached in terms of the identity preservation and attribute control accuracy. Note that we compute Mean Absolute Error for numeric attributes (e.g., yaw, age) and Classification Acc. for binary attributes (e.g., glasses, smile). The identity similarity score is between the original face and the manipulated.

| attribute type	single-attribute editing				multi-attribute editing	
	yaw	age	glasses	smile	glasses+smile	yaw+glasses
method	Identity Similarity Score(↑)					
InterFaceGAN ([22])	0.78 ± 0.05	0.82 ± 0.06	0.84 ± 0.12	0.95 ± 0.05	0.74 ± 0.05	0.68 ± 0.15
StyleFlow ([5])	0.82 ± 0.07	0.86 ± 0.08	0.85 ± 0.1	0.96 ± 0.05	0.83 ± 0.12	0.78 ± 0.12
ISFGAN [32]	-	0.85 ± 0.13	0.88 ± 0.03	0.96 ± 0.03	0.85 ± 0.08	-
Ours (w/o L_{id})	0.85 ± 0.04	0.86 ± 0.03	0.87 ± 0.08	0.97 ± 0.02	0.85 ± 0.12	0.82 ± 0.24
Ours (w/ L_{id})	$\mathbf{0.95} \pm 0.05$	$\mathbf{0.89} \pm 0.08$	$\mathbf{0.90} \pm 0.09$	$\mathbf{0.98} \pm 0.1$	$\mathbf{0.87} \pm 0.1$	$\mathbf{0.85} \pm 0.09$
method	Attribute Control Accuracy						
	yaw(↓)	age (↓)	smile (↑)	glasses (↑)	glasses(↑)	smile(↑)	yaw(↓)	glasses(↑)
InterFaceGAN ([22])	12.95	13.50	0.894	0.93	0.832	0.877	15.33	0.826
StyleFlow ([5])	6.41	**12.78**	0.975	0.981	0.944	0.921	8.58	0.925
ISFGAN [32]	-	13.97	0.963	0.985	0.935	0.901	-	-
Ours	**6.33**	13.77	**0.976**	**0.988**	**0.963**	**0.955**	**7.26**	**0.961**
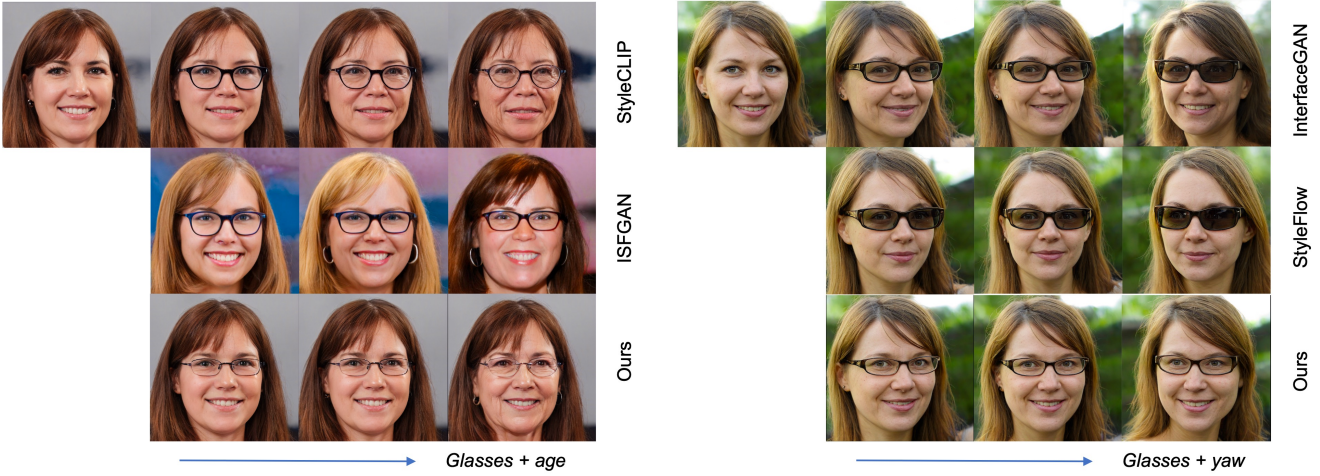
Figure 5: Fixing a specific property while continuously manipulating other properties. Glasses are chose to fix due to their instability. As shown, our method stably manipulates pose and age attributes while minimizing changes in glasses shape and color. Furthermore, our method clearly has the best consistency with the original image for unedited attributes, such as smiling expressions.

tance between the distribution of manipulated images and that of real images (FFHQ dataset) with the Fréchet In-

Table 2: The FID comparisons between manipulated faces I_M.

Methods	FID(↓)			
	age	glasses	smile	avg
InterFaceGAN [22]	64.32	62.66	57.91	61.63
StyleFlow [5]	53.5	51.85	51.34	52.23
StyleCLIP [18]	51.23	47.45	47.54	48.74
ISFGAN [32]	52.87	51.3	47.78	50.65
Ours	47.73	42.69	41.55	**43.98**

ception Distance (FID) [10]: see Table 2. As shown, our model exhibits higher control precision of all attributes except for age. When joint manipulating multiple attributes, the control precision and identity similarity scores of compared methods deteriorate significantly while our method performs consistently well. In addition, we cancel the identity loss as the ablation experiment, in order to ensure consistency with the evaluation methods, see Table 1. Although the quantitative index is slightly lower than using the identity constraint, it still has the best similarity compared with other approaches.

4.3. Extended ablations studies

We qualitatively and quantitatively analyze and compare ways of multi-attribute fusion (cross-attention instead of
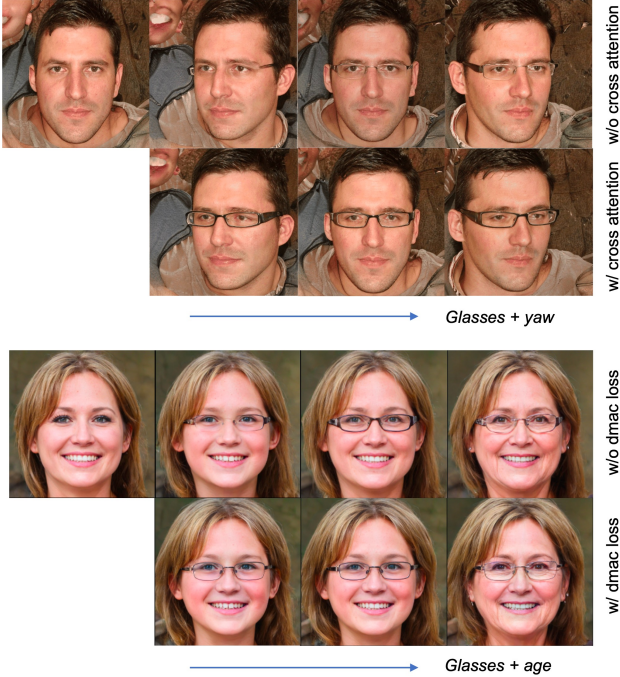
Figure 6: Qualitatively demonstrate the stability of binary attribute under the condition of continuous editing of numerical attributes.

Table 3: The evaluation of ablation controls the accuracy of individual attribute of glasses, as well as the accuracy of the attribute of glasses under the condition of editing yaw or age.

Model	Attribute Control Accuracy($\uparrow$)		
	glasses	glasses|yaw	glasses|age
w/o CA & L_{dmac}	0.943	0.921	0.919
w/o CA	0.975	0.947	0.936
w/o L_{dmac}	0.951	0.932	0.924
full	**0.988**	**0.965**	**0.96**

MLP) and the performance of L_{dmac} loss on multi-attribute joint editing. As shown in Figure 10 and Table 3, both tricks independently contribute to the stability of multi-attribute editing.

To verify the effectiveness of the proposed dynamic architecture and two-stage training procedure, we prepared two validation datasets separately for multi-attribute-conditioned realistic (FFHQ) editing and comic face editing. We started three trainings, which include the static architecture (all branches are activated regardless of the input as in Figure 3) trained for joint multi-attribute editing, the dynamic architecture trained with the two-stage training procedure, and that trained for multi-attribute editing only (single-stage). We visualize how the validation losses

(L_{id}, L_{attr}, $L_{attr}^{A_i}$ as defined in the method section) change against the number of training steps. As shown in Figure 8 in the supplementary materials, for both experiments, the static architecture cannot converge as well as the dynamic architecture, implying its invulnerability in adapting to various kinds of attribute configurations. As for the dynamic architecture, the single-stage training procedure is highly unstable and achieves worse identity preservation and average control accuracy. Visual comparisons are illustrated in Figures 9, Figure 10 in the supplementary materials. More ablation studies on the architecture features and training techniques are presented in the supplementary materials.
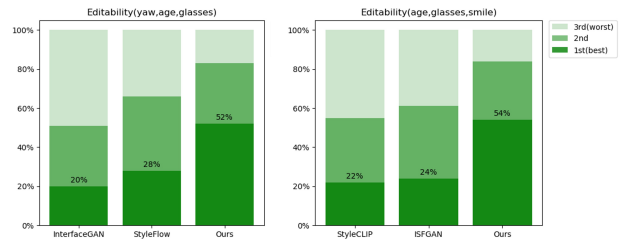
4.4. User Study

Figure 7: User study of jointly manipulating multi-attributes, corresponding to qualitative comparisons.

To perceptually evaluate multi-attribute editing performance, we conducted a user study. We evaluate the sequential editing results of two sets of multiple attributes: the first set is yaw, age and glasses, where the presence of pose attribute allows us to compare only InterfaceGAN [22] and StyleFlow [5]; the second set is age, glasses, smile, we evaluate StyleCLIP [18] and ISFGAN [32]. We collected 600 votes from 20 participants to evaluate each set of 30 images. In conclusion, our method received more than half of the first approvals.

5. Conclusion

In this paper, we propose a dynamic neural network that enables nonlinear and adaptive style manipulations for multi-attribute conditioned image generation. Additionally, our dynamic multi-attribute contrastive learning method effectively solves the entanglement problem of multi-attribute joint editing. Compared with other static style manipulation approaches, our model exhibits higher average precision of attribute-control and improved competency of identity preservation. When manipulating multiple attributes, the superiority of our approach becomes more significant. Future work should focus on more fine-grained division of model latent space in order to ensure the stable joint editing of more attributes.

References

[1] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *SIGGRAPH*, pages 187–194, 1999.

[2] Gwern Branwen. Danbooru2019: A large-scale crowd-sourced and tagged anime illustration dataset. 2015.

[3] Alaluf et al. Only a matter of style: Age transformation using a style-based regression model. *TOG*, 40(4), 2021.

[4] Alaluf et al. Only a matter of style: Age transformation using a style-based regression model. *arXiv preprint*, 2021.

[5] Abdal et al. Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows. *TOG*, 40(3):1–21, 2021.

[6] Brock et al. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint*, 2018.

[7] Choi et al. Stargan v2: Diverse image synthesis for multiple domains. In *CVPR*, pages 8188–8197, 2020.

[8] Goodfellow et al. Generative adversarial networks. *arXiv preprint*, 2014.

[9] He et al. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[10] Heusel et al. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *arXiv preprint*, 2017.

[11] Huang et al. Curricularface: adaptive curriculum learning loss for deep face recognition. In *CVPR*, pages 5901–5910, 2020.

[12] Karras et al. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint*, 2017.

[13] Karras et al. A style-based generator architecture for generative adversarial networks. In *CVPR*, pages 4401–4410, 2019.

[14] Karras et al. Analyzing and improving the image quality of stylegan. In *CVPR*, pages 8110–8119, 2020.

[15] Karras et al. Training generative adversarial networks with limited data. *arXiv preprint*, 2020.

[16] Liu et al. Deep learning face attributes in the wild. In *ICCV*, December 2015.

[17] Oord et al. Representation learning with contrastive predictive coding. *arXiv preprint*, 2018.

[18] Patashnik et al. Styleclip: Text-driven manipulation of stylegan imagery. *arXiv preprint*, 2021.

[19] Rothe et al. Dex: Deep expectation of apparent age from a single image. In *ICCV workshops*, pages 10–15, 2015.

[20] Radford et al. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint*, 2015.

[21] Richardson et al. Encoding in style: a stylegan encoder for image-to-image translation. *arXiv preprint*, 2020.

[22] Shen et al. Interfacegan: Interpreting the disentangled face representation learned by gans. *TPAMI*, 2020.

[23] Tewari et al. Stylerig: Rigging stylegan for 3d control over portrait images. In *CVPR*, pages 6142–6151, 2020.

[24] Tov et al. Designing an encoder for stylegan image manipulation. *TOG*, 40(4):1–14, 2021.

[25] Wang et al. Hijack-gan: Unintended-use of pretrained, black-box gans. In *CVPR*, pages 7872–7881, 2021.

[26] Yang et al. Ssr-net: A compact soft stagewise regression network for age estimation. In *IJCAI*, volume 5, page 7, 2018.

[27] Zhang et al. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595, 2018.

[28] Erik et al. Härkönen. Ganspace: Discovering interpretable gan controls. *arXiv preprint arXiv:2004.02546*, 2020.

[29] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *Proc. NeurIPS*, 2020.

[30] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. In *Proc. NeurIPS*, 2021.

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

[32] Yahui Liu, Yajing Chen, Linchao Bao, Nicu Sebe, Bruno Lepri, and Marco De Nadai. Isf-gan: An implicit style function for high-resolution image-to-image translation. *arXiv preprint arXiv:2109.12492*, 2021.

[33] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint*, 2014.

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

[35] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, and J. Clark. Learning transferable visual models from natural language supervision. 2021.

[36] Nataniel et al. Ruiz. Fine-grained head pose estimation without keypoints. In *CVPR workshops*, pages 2074–2083, 2018.

[37] Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in gans. *arXiv preprint*, 2020.

[38] Yichun Shi, Xiao Yang, Yangyue Wan, and Xiaohui Shen. Semanticstylegan: Learning compositional generative priors for controllable image synthesis and editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11254–11264, 2022.

[39] Sheng-Yu Wang, David Bau, and Jun-Yan Zhu. Sketch your own gan. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14050–14060, 2021.

[40] Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. Tedigan: Text-guided diverse face image generation and manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2256–2265, 2021.

[41] Yu Zeng, Zhe Lin, and Vishal M Patel. Sketchedit: Mask-free local image manipulation with partial sketches. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5951–5961, 2022.