# Attribute Controllable Beautiful Caucasian Face Generation by Aesthetics Driven Reinforcement Learning

Xin Jin
Beijing Electronic Science and
Technology Institute

Shu Zhao
Beijing Electronic Science and
Technology Institute

Le Zhang*
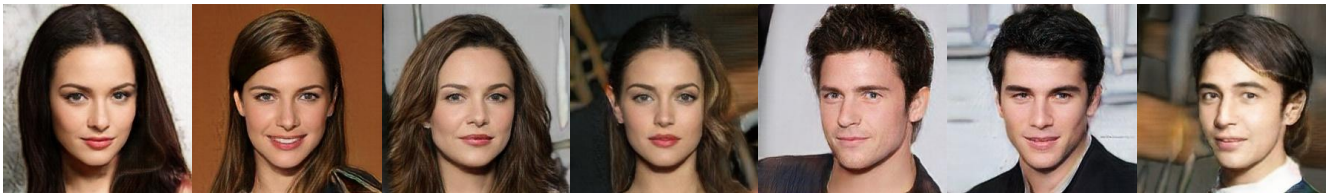Beijing Electronic Science and
Technology Institute

Xin Zhao
Beijing Electronic Science and
Technology Institute

Qiang Deng
Beijing Electronic Science and
Technology Institute

Chaoen Xiao
Beijing Electronic Science and
Technology Institute

*EigenGAN*



*Our Approach*

Figure 1: Top: the original faces generated by EigenGAN [6]. Bottom: aesthetically improved ones by aesthetics-driven reinforcement learning over semantic attributes of the generator.

## ABSTRACT

In recent years, image generation has made great strides in improving the quality of images, producing high-fidelity ones. Also, quite recently, there are architecture designs, which enable GAN to unsupervisedly learn the semantic attributes represented in different layers. However, there is still a lack of research on generating face images more consistent with human aesthetics. Based on Eigen-GAN [He et al., ICCV 2021], we build the techniques of reinforcement learning into the generator of EigenGAN. The agent tries to figure out how to alter the semantic attributes of the generated human faces towards more preferable ones. To accomplish this, we trained an aesthetics scoring model that can conduct facial beauty prediction. We also can utilize this scoring model to analyze the correlation between face attributes and aesthetics scores. Empirically, using off-the-shelf techniques from reinforcement learning would not work well. So instead, we present a new variant incorporating the ingredients emerging in the reinforcement learning communities in recent years. Compared to the original generated images, the adjusted ones show clear distinctions concerning various attributes. Experimental results using the MindSpore, show the effectiveness of the proposed method. Altered facial images are commonly more attractive, with significantly improved aesthetic levels.

## CCS CONCEPTS

• **Applied computing** → *Media arts.*

## KEYWORDS

image aesthetics, face generation, reinforcement learning

*Corresponding author email: lezhang.thu@gmail.com

# 1 INTRODUCTION

In recent years, Generative Adversarial Networks (GANs) [4] show promising performance on image synthesis. A parallel line of research tries to learn interpretable GAN generators, which can discover explicit dimensions controlling the semantic attributes. InfoGAN in [1] learns disentangled representations by decomposing the conventional single unstructured noise vector into incompressible noise and a set of structured latent variables. Motivated by style transfer literature, Karras et al. proposed StyleGAN in [8], enabling separations of high-level attributes like pose and identity. Most recently, He et al. proposed EigenGAN [6], which for each generator layer directly embeds in one linear subspace model with an orthogonal basis. During the training, EigenGAN automatically learns a set of "eigen-dimensions" corresponding to interpretable semantic attributes. So until now, people have several ways to manipulate the semantics within the GANs, or said in other words, semantically edit them. One problem followed is how to enhance the generated face images positively.

In this paper, we incorporate the techniques of reinforcement learning (RL) into the attribute values searching in EigenGAN. The first requirement of the RL agent is the design of the reward function. The objective is to enhance the images to the ones more aesthetically attractive. So for this, we trained an aesthetics score network based on SCUT-FBP5500 released in [12]. By this, we give a multi-attribute model using MindSpore[15], which can conduct automatic Caucasian face beautification for the generated images, as shown in Fig. 1. The user can select different facial attributes in the interaction section to change the generated image.

# 2 RELATED WORK

**Reinforcement Learning in Aesthetics.** Li et al. proposed an RL-based framework named Fast A3RL (Aesthetics-Aware Adversarial) [10, 11], which dedicates to the task of image cropping. This task requires the agent to extract a well-composed region from the input image. Li et al. first formulated the cropping as a decision-making process. Next, they introduced an RL-based cropping network, which contains an actor-critic architecture [17]. Although Fast A3RL is also an application of RL in aesthetics, the work proposed here differs in several aspects. On one side, Fast A3RL relies on a specially designed reward function, which involves human-designed heuristics besides an aesthetics score network. The solution discussed here, however, only requires a straightforward regression over SCUT-FBP5500 [12] to give facial beauty predictions. The solution does not need any hand-craft features. On the other side, RL techniques utilized in Fast A3RL are conventional policy gradient methods. So it requires two networks, i.e., a policy network and a value network. Whereas we follow the approach taken in the literature on image captioning [18, 19, 26], and use a self-critical baseline. This form not only simplifies the network design but also liberates the agent from learning accurate value estimations.

**Facial Attractiveness Enhancement.** There are other facial image beautification techniques proposed in recent years. Diamant et al. proposed Beholder-GAN [3], which is a variant of PGGAN (Progressive Growing) [7] and CGAN (Conditional) [16]. Diamant et al. first trained a GAN based on beauty scores. Then for real images, the latent vectors and the beauty scores are recovered using GAN inversion techniques. To make input images more attractive, one only needs to increase the beauty scores recovered. Although effective, Beholder-GAN struggles to preserve the facial identity of the input image. In [27], Zhou and Xiao proposed an approach based on InterFaceGAN [21], and treated the "beauty" as a binary semantic. We note that in InterFaceGAN, the main assumption is "for any binary semantic (eg, male v.s. female), there exists a hyperplane in the latent space serving as the separation boundary." Although intuitively implausible (as beauty is not obvious binary semantic as other semantics like the presence of eyeglasses or not), Zhou and Xiao managed to validate the existence of such a beauty hyper-plane. Like InterFaceGAN, they can now semantically edit this beauty semantic of input images, generating aesthetically more attractive ones. Besides, by inserting an identity preserving loss, they proposed an enhanced version of Beholder-GAN. Nonetheless, as presented (and illustrated) in their paper, this enhanced version only solved identity-preserving weakness to some extend. We note that for these approaches, GANs are still acting like a black box. Whereas, by contrast, we seek for precise controlling of explicit semantics. The pre-trained model in [21] is not released, so the comparison to it is not conducted in the experiment.

# 3 EIGENGAN + RL

In this section, we detail the building blocks for incorporating the policy gradient algorithm into the generator of EigenGAN. We first briefly review the preliminaries of RL and the architecture of the generator of EigenGAN. We depict it as part of Fig. 2.

## 3.1 Preliminaries for RL

In RL, the agent constantly interacts with the environment, and this process is described as a Markov Decision Process (MDP). We define the MDP as $(\mathcal{S}, \mathcal{A}, P, r, \rho_0, \gamma)$, where $\mathcal{S}$ and $\mathcal{A}$ are the state and action space resp., $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$ is the state transition probability, $r : \mathcal{S} \rightarrow \mathbb{R}$ is the reward function, $\rho_0$ is the distribution of the initial state $s_0$, and $\gamma$ is the discounting factor satisfying $0 \leq \gamma \leq 1$.

The objective is to learn a stochastic policy $\pi : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$, such that when taking actions $\mathbf{a}_t \sim \pi(\mathbf{a}_t|\mathbf{s}_t)$, the expected reward of the trajectories is maximized, i.e.,

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{\mathbf{s}_0, \mathbf{a}_0, \ldots} \left[ \gamma^t r(\mathbf{s}_t) \right], \tag{1}$$

where $\mathbf{s}_0 \sim \rho_0(\mathbf{s}_0)$, $\mathbf{a}_t \sim \pi(\mathbf{a}_t|\mathbf{s}_t)$, $\mathbf{s}_{t+1} \sim P(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$.

In this paper, the circumstance considered is a simplified version of the one above. The discounting factor $\gamma$ equals 1. Also, no rewards exist for intermediate states. The agent only gets a reward for the terminal state. The reward value is the aesthetic score of the image after a series of adjustments.

## 3.2 Generator of EigenGAN

The controlling parameters of the generator consist of not only the conventionally noise input $\epsilon \in \mathcal{N}(0, \mathbf{I})$, but also a sequence of latent variables $\mathbf{z}_i \in \mathbb{R}^q$, which are similarly sampled from $\mathcal{N}_q(0, \mathbf{I})$ (for EigenGAN, $q$ is set to 6). Six blocks of transposed convolution layers constitute the main part of the neural architecture. Strides of
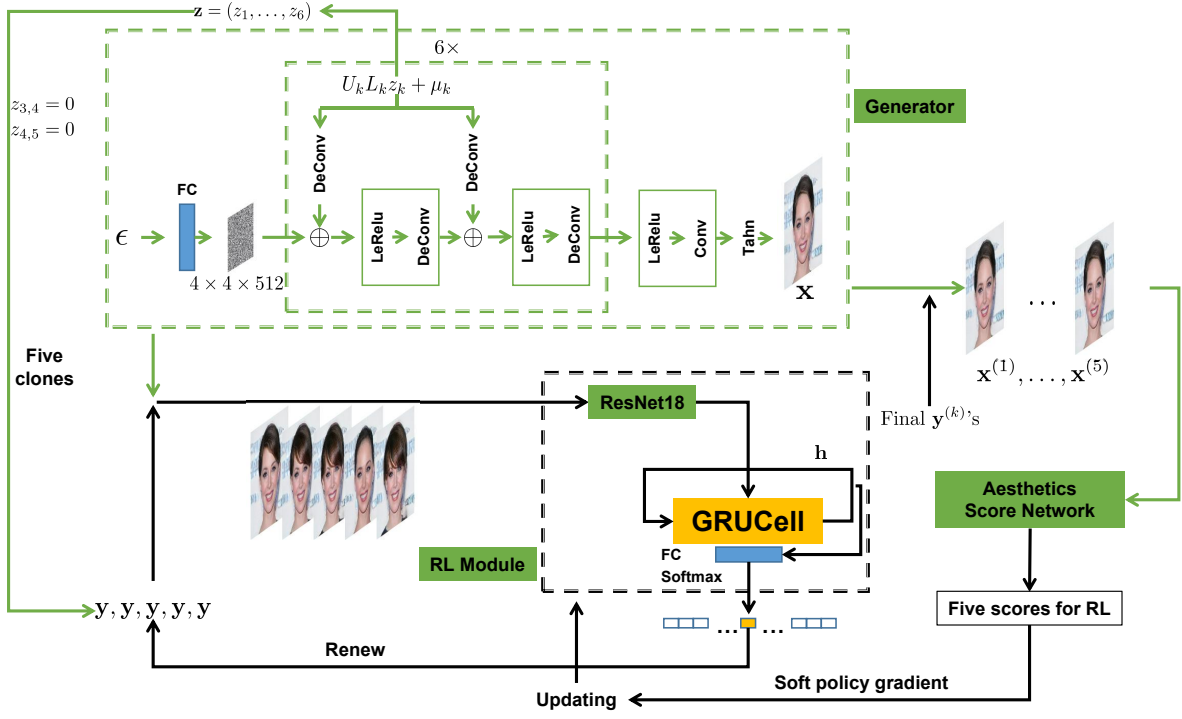
**Figure 2: Illustration of the training procedure over two semantics $z_{3,4}$ and $z_{4,5}$. After setting $z_{3,4}$ and $z_{4,5}$ to zero, by cloning z, we have five identical y's. We generate five images by feeding y's into the generator. We use ResNet18 to extract the features of the five images, and input them into a GRUCell. Next, the hidden state of GRUCell is input into a fully connected layer, and a softmax layer then gives the action distribution for choosing values for $z_{3,4}$. We renew these newly chosen values into y's. Again, by feeding these updated y's into the generator, we have five new fresh images. The procedure repeats, and we get the sampling values for $z_{4,5}$. After updating the values for $z_{4,5}$ into y's, we get the final vectors $y^{(k)}$'s. We again feed $y^{(k)}$'s into the generator to get the final five images corresponding to five randomly sampled trajectories. The aesthetics score network will score them and return five scores. These scores provide an informative baseline for the soft policy gradient. The RL module is next updated to raise the probabilities for the trajectories with higher scores. Finally, for the inference, the agent will not sample actions after the softmax layer. In contrast, it greedily chooses the most probable one.**

the two transposed convolutions are 2 and 1 resp. Hence, the size of the image is scaled by a factor of 2 when traveling through one such block. Within the block, the randomly sampled point (due to the randomness of $z_i$) from the subspace model $S_i = (\mathbf{U}_i, \mathbf{L}_i, \mu_i)$ is embedded into the image by element-wise addition. For the notations, $\mathbf{U}_i$ is the orthonormal basis, $\mathbf{L}_i$ is a diagonal matrix indicating the "importance," and $\mu_i$ is the origin of the subspace.

For the input part, it is a fully connected layer mapping $\epsilon$ into a $4 \times 4 \times 512$ image. Correspondingly for the output, it is passed into a convolutional layer, getting a generated $256 \times 256$ image of 3 channels.

## 3.3 Learn to set semantic attributes

For EigenGAN, dimensions from latent variables $\mathbf{z}_i$ correspond to interpretable or controllable semantic attributes. As an example, the fourth dimension of $\mathbf{z}_3$ is related to the control of the bangs.

The sixth dimension of $\mathbf{z}_5$, for a specific pre-trained generator, corresponds to the lipstick color. So as we move along these dimensions, the generator produces a sequence of images with continuous changes corresponding to the relevant semantic attributes. Six layers in EigenGAN brings us six such $\mathbf{z}$'s. Recall every $\mathbf{z}_i \in \mathbb{R}^q$, and $q = 6$. We only adopt a subset of these controllable semantic attributes, as the others are a bit entangled or drastic, like the first dimension of $\mathbf{z}_3$ is possible to be unsupervisedly trained to change the age or even the gender of the generated face. The premise of the adjustments is to keep the unaltered version still recognizable. Let $C$ be the set of selected controllable semantic attributes. We only discuss how to let the RL agent learn to choose the preferred values. We defer the details of the RL algorithm to the next section.

Formally as shown in 2 (for conciseness, only two controllable dimensions are involved), we first set the dimensions corresponding to the attributes in $C$ as zero. Any other "sensible" values (EigenGAN samples these values according to a standard normal distribution) can also be okay, as the agent will set or override these values

anyway. The agent learns how to select over this dimension during the training process. After these setting zero operations, we concatenate these altered $z_i$'s into one single vector. Let it be $\mathbf{y}$. We clone the single vector five times. The value of five caters to the five trajectories or decision sequences sampled by the RL agent. Intuitively, we train the agent over these sampled trajectories. The agent then gradually learns to select the preferred one.

As depicted in Fig. 2, these five (identical) $\mathbf{y}$'s will be passed into the generator. Accompanied with the same $\epsilon$, these five $\mathbf{y}$'s induce five (identical) images. Remind that for a frozen pre-trained Eigen-GAN generator, passing in the same $\epsilon$ and $\mathbf{z}_i$'s deterministic-ally generates the same image. The next step is the conventional feature extraction process. In this paper, we directly use a pre-trained ResNet18 model, for which we replace the last fully connected layer with a counterpart of output dimension 512. The value of 512 is consistent with the length of $\epsilon$. We also choose this value as the default dimension of latent vectors. We do not freeze the other layers of ResNet18. By contrast, we finetune these weights in the RL training.

The extracted features vectors are the input for a gated recurrent unit (GRU) cell, of which the size of the hidden state is 512. With the embedding of the current state, we feed the hidden states of the GRU cell into a fully connected module. With a softmax layer, the output of the RL module indicates the action probability distributions. The agent then samples an action. For example in Fig. 2, it will sample a value for $z_{3,4}$. Recall that there are five identical images for the first step. So we will sample five (possibly different) values for $z_{3,4}$.

Next, one obtains five renewed $\mathbf{y}$'s by setting the newly sampled values for $z_{3,4}$. So these $\mathbf{y}$'s can start to diverge from each other. Next, the process is repeated, i.e., feature extracting, fully connected mapping over the updated hidden state of GRU cell, sampling from distributions given by the softmax layer. The only difference is to randomly sample the values for $z_{4,5}$, not $z_{3,4}$.

Consider the simplified case in which we only need to set two attributes $z_{3,4}$ and $z_{4,5}$. At this very moment, the final $\mathbf{y}^{(k)}$'s with the same $\epsilon$ determine the final generated images. These five (probably different) images are input into the aesthetics score network, generating five scores for RL training.

## 3.4 Soft policy gradient + self-critical

The standard objective in reinforcement learning is the expected sum of rewards $\sum_t \mathbb{E}_{(\mathbf{s}_t, \mathbf{a}_t) \sim \rho_\pi}[r(\mathbf{s}_t, \mathbf{a}_t)]$. Most RL algorithms are working on the conventional notion of deterministic optimality. That is, the optimal policy we are searching for is always deterministic. And this holds at least for the circumstances of full observability [5]. However, intuitively the conventional objective does not consider the importance of stochasticity for exploration. A sufficient amount of randomness is critical for searching for near-optimal policies, preventing premature convergence to suboptimal ones. One way of keeping randomness is including an entropy regularization term (referred to as entropy bonus) in the loss function, like in [17, 20]. Even so, this is not a systematic way of ensuring the stochasticity of the policy. In that way, the entropy term serves as a regularizer, not as one of the objectives of the RL agent.

In this paper, we instead follows the framework of maximum entropy reinforcement learning (see e.g. [28]), which aims for a policy $\pi^*$ as follows:

$$\pi^* = \arg\max_\pi \sum_t \mathbb{E}_{(\mathbf{s}_t, \mathbf{a}_t) \sim \rho_\pi}[r(\mathbf{s}_t, \mathbf{a}_t) + \alpha \mathcal{H}(\pi(\cdot|\mathbf{s}_t)], \quad (2)$$

where $\mathcal{H}(\pi(\cdot|\mathbf{s}_t))$ is the entropy, and $\alpha$ is a hyper-parameter controlling the relative importance of $\mathcal{H}(\pi(\cdot|\mathbf{s}_t))$ in the objective.

The next step is to get the gradient of this objective in 2 w.r.t. the policy parameters. In [5], Haarnoja et al.implicitly gave the following form of the gradient (see Eqs. (27) and (28) in Appendix B of the arXiv version), and this form is explicitly stated in [22]. The result is as follows:

THEOREM 3.1 (SOFT POLICY GRADIENT [5, 22, 23]). *Suppose that the policy $\pi(\cdot|\mathbf{s})$ is differentiable w.r.t. its parameters $\theta$. Then,*

$$\nabla_\theta J_{\mathrm{PG}}(\pi_\theta)$$
$$= \mathbb{E}_{(\mathbf{s}_t, \mathbf{a}_t) \sim \rho_{\pi_\theta}} \left[ \left( \hat{Q}_{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t) - b(\mathbf{s}_t) \right. \right.$$
$$\left. \left. -\alpha(1 + \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t)) \right) \nabla_\theta \log \pi_\theta(\mathbf{a}_t|\mathbf{s}_t) \right], \quad (3)$$

*where $\alpha$ is the temperature hyper-parameter controlling the the extent of exploring, $\hat{Q}_{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t)$ is an empirical estimate of the Q-value of the policy, and $b(\mathbf{s}_t)$ is a state-dependent baseline, which can be any function as long as it does not vary with the actions.*

With the theorem above, we follow the approach proposed in [19], i.e., incorporating the self-critical way of training into the soft policy gradient updates. More concretely, $\hat{Q}_{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t)$'s are replaced with Monte Carlo estimations over single trajectories. These estimations correspond to the final five aesthetics scores given by the aesthetics score network. This disposes the need for estimations via separate neural networks for predicting $\hat{Q}_{\pi_\theta}(\mathbf{s}_t, \mathbf{a}_t)$'s. For $b(\mathbf{s}_t)$ in 3.1, it is set as the average of the five scores. This baseline deviates slightly from [19] (which uses the baseline score got by the greedy policy under current $\pi_\theta$) but experimentally it is more effective as shown in [14].

In a word, the main difference from conventional self-critical training is applying the way of self-critical on the soft (not the vanilla) policy gradient. In the experiment section, we show this change is critical to ensure sufficient exploration, and thus this guarantees better performance.

## 3.5 Aesthetics score network

We designed a network dedicated to scoring the aesthetics of human faces. We use the lightweight module ECA (Efficient Channel Attention) [25], which acts as a plug-and-play block, and we apply it to the proposed network. We adopt EfficientNet-B4 [24] as a pretrained model to extract 1,792 features of the image. Then a convolution operation is performed with a kernel size of 3. We normalize the features before using the ReLU activation function. Next, it is input to the ECA module to obtain a 4-dimensional vector, and we flatten this vector into a 1-dimensional vector after activation and adaptive average pooling. After this, it is input into the regression network, which returns the aesthetics score.

We trained this aesthetic face scoring model above using the SCUT-FBP5500 dataset [12]. The SCUT-FBP5500 dataset contains a total of 5,500 frontal faces and their beauty scores, including 2,000

Asian females, 2,000 Asian males, 750 Caucasian females, and 750 Caucasian males. The batch size is 64 for the training, and the initial learning rate is 1e-4. The training consists of 10 epochs. For validation, we remap the prediction score of our model to [1, 5], which is the beauty score range in SCUT-FBP5500. The RMSE (root-mean-square error) of our model is 0.3058, which is better than the best result of 0.3325 presented in the paper of SCUT-FBP5500 [12]. The latter adopts a CNN-based ResNeXt-50 architecture.

## 4 EXPERIMENTS

All the RL models presented in this section use a batch size of 16. Since the RL agent samples five random trajectories, the literal batch size is 80 for each gradient updating step. By fixing the random seed, we maintain the same validation set across all comparisons. Identical to the experiments conducted in the original paper of EigenGAN, for the Adam optimizer [9, 13], we use a learning rate of 1e-5 with no $L_2$ regularization over the parameters. For the GRU cell, the size of the hidden state is 512, which is the same value for the output of the ResNet18 module. The ResNet18 module is a pre-trained version on ImageNet [2]. We set the hyperparameter $\alpha$ in 3 as 0.01 for soft policy gradient.

For the action space, for each semantic attribute, we discretize the range [−4.5, 4.5], taking equally spaced 17 points. It works well, although one could use continuous values or split into more bins. Nonetheless, we observe no significant improvement with more fine-grained splits, partially because the generated images are insensitive to small fluctuations of attribute values. The final models are versions with 20,000 updating iterations, and we report the models with the highest scores on the validation set (please refer to Fig. 3).

### 4.1 Attribute selection

EigenGAN discovers various facial semantic attributes learned in each layer of subspace, and we wish to analyze their correlations with the aesthetic quality. We use Spearman's rank correlation coefficient or Spearman's $\rho$, which measures the dependence between two variables by performing linear correlation analysis on the ranks of two variables. For the analysis here, the aesthetics score range is [0, 1]. The searching range of semantic attributes is [−4.5, 4.5]. We discretize it with step size 0.5. We calculate Spearman's $\rho$ for 30 randomly generated pictures. Correlation coefficients greater than 0.8 indicate high correlation; We interpret those with values less than 0.3 as irrelevant. In addition, we perform a significance test over these coefficients. Hypotheses as to this test are $H_0 : \rho = 0$ and $H_1 : \rho \neq 0$. We set the threshold of indicating significant difference as 0.05 for the $p$-values. We list the results in Table 1.

Lastly, we have to choose the dimensions with better decoupling effects, i.e., one only observes continuous changes of a *single* semantic attribute when traversing the coefficients of those eigen-dimensions. This step is necessary mainly because, in Eigen-GAN, different eigen-dimensions might control the same semantic. For example, changes in $z_{5,2}$ and $z_{5,3}$ both affect the hair color. To break the ties, we adopt a greedy policy by choosing the more relevant one. To further narrow the scope, only the attributes with strong correlation ($|\rho| > 0.3$) and also being significant ($p$-value<

**Table 1: Correlation test results of 30 pictures randomly generated by EigenGAN. The notation $z_{i,j}$ indicates the $j^{th}$ dimension of the $i^{th}$ layer. When the $p$-value is less than $10^{-4}$, we treat it as 0.0. For repeated attributes, we choose the most relevant one. The final selected ones are marked in bold.**

| D | Attribute | Spearman's $\rho$ | $p$-value |
|---|---|---|---|
| $z_{2,5}$ | **Body Pose** | 0.418 | 0.0 |
| $z_{3,4}$ | **Bangs** | -0.857 | 0.0 |
| $z_{3,5}$ | Hair Style | -0.125 | $2.875 \times 10^{-3}$ |
| $z_{3,6}$ | Body Side | 0.188 | 0.0 |
| $z_{4,1}$ | Pose | 0.328 | 0.0 |
| $z_{4,2}$ | Background | -0.978 | 0.0 |
| $z_{4,3}$ | Pose | 0.027 | 0.1551 |
| $z_{4,4}$ | **Lighting** | -0.561 | 0.0 |
| $z_{4,5}$ | Smiling | 0.786 | 0.0 |
| $z_{4,6}$ | Face Shape | -0.720 | 0.0 |
| $z_{5,1}$ | Background | -0.198 | 0.0 |
| $z_{5,2}$ | Hair Color | -0.164 | 0.0 |
| $z_{5,3}$ | Hair Color | 0.685 | 0.0 |
| $z_{5,4}$ | Lighting | -0.061 | 0.1484 |
| $z_{5,5}$ | **Gaze** | 0.585 | 0.0 |
| $z_{5,6}$ | **Lipstick Color** | 0.471 | 0.0 |

0.01) are selected. We further filter the candidate set by observing whether they preserve the facial identities or not. After this step, we choose only five semantics, as marked boldly in Table 1.

### 4.2 Effect of soft policy gradient on entropy

In 3.4, we emphasize the role of soft policy gradient in ensuring the higher entropy of $\pi_\theta$. Without the soft policy gradient, the main phenomenon observed is the quick diminishing of entropy, which indicates the possibility of exploration of diverse choices is gradually becoming zero. For RL agents, this might induce premature convergence to suboptimal deterministic policies. To simplify the observation, we only consider the case of setting only one dimension, i.e., $z_{3,4}$. Thus it is a single-step environment now. The difficulty of achieving higher scores by changing only $z_{3,4}$ justifies the choice of this dimension. Fig. 4(a) depicts the changing of entropy w.r.t. training iterations. The initial entropy value is slightly higher than 2.82. As can be observed, if following vanilla policy gradient updates, the average entropy of distributions defined by $\pi_\theta(\cdot|\mathbf{s}_t)$ decreases to values less than 0.3, whereas, by using soft policy gradient, this value is kept still above 2.3. Higher entropy values indicate the fact that there are still possibilities for exploration. It helps the agents to learn and perform better finally.

### 4.3 Improved performance with soft policy gradient and more attributes

In this section, we extend the former one-step environment to a counterpart of two steps, i.e., $z_{3,4}$ and $z_{4,5}$. So one can observe the improved performance brought by considering more attributes adjustable. Meanwhile, we also show the results on using soft policy gradient or not. Intuitively higher entropy enables the possibility
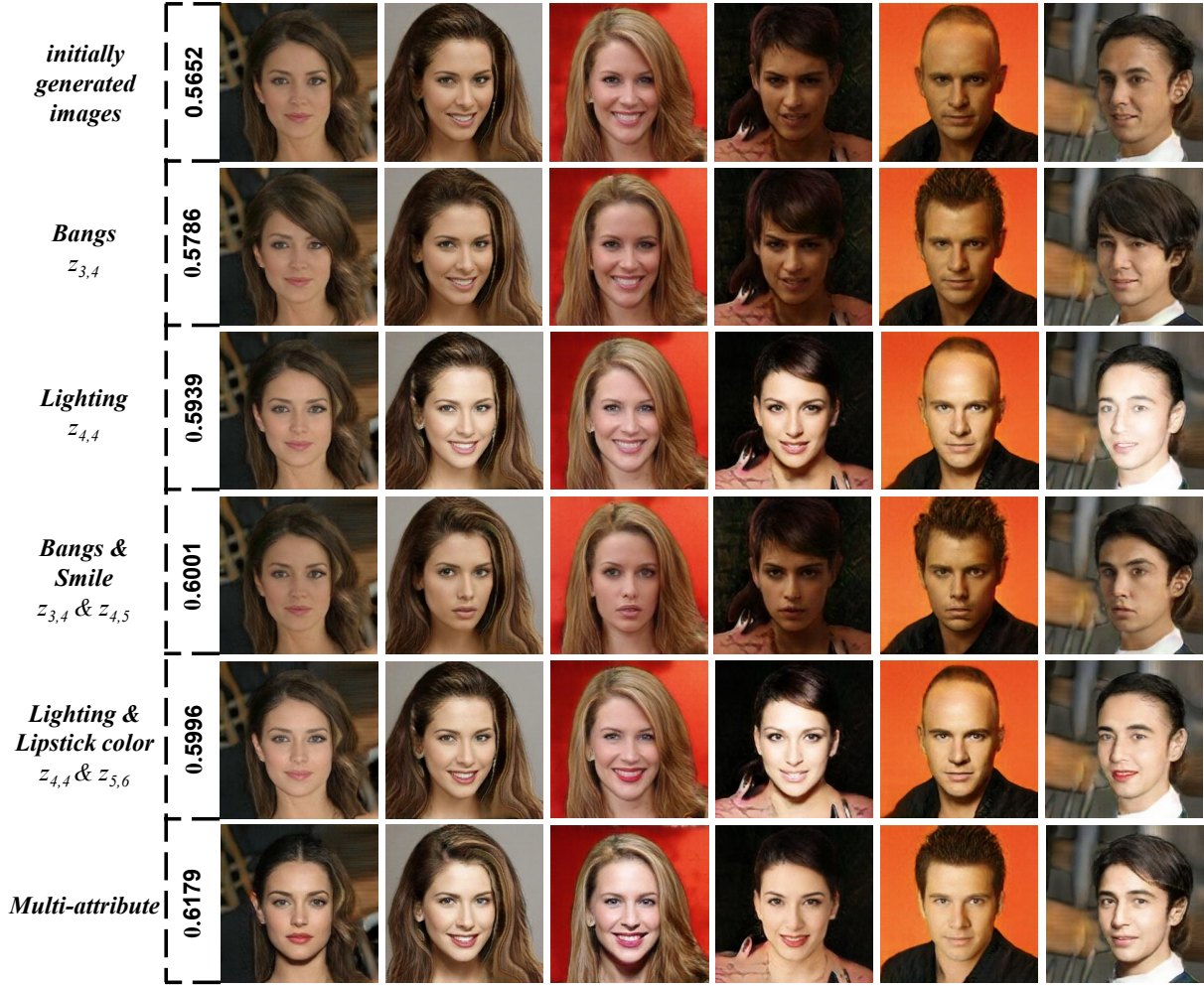
**Figure 3: A qualitative comparison of aesthetic improvement with different sets of attributes. Six rows are listed, which indicates six distinct suits of semantics available. Also, next to each row of illustrated pictures, we type vertically the average scores of 1,000 randomly generated test images.**

of still exploring the environment and thus getting higher scores. We verify this intuition in the experiment. We note higher entropy of $\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)$ does not necessarily mean the agent cannot perform optimally. By contrast, as in the inference phase, we only select $\mathbf{a}^* = \arg\max_{\mathbf{a}_t} \pi_\theta(\cdot|\mathbf{s}_t)$, so as long as preferred actions still dominate the probability mass, intuitively the agent still performs optimally. We adopt a randomly generated set of 80 images for validation. The original average score of these images is 0.54530, shown in Fig. 4(b) for iteration 0. Consider the increase of scores by adjusting one more attribute. If using vanilla policy gradient, for one-step setting, the validation score is improved to 0.55991; for two-steps, this value is 0.58539. Meanwhile, for soft policy gradient, these two values are 0.56649 and 0.60051 resp. So this indicates higher scores with more attributes adjustable. Also, most of the time, the curves concerning soft policy gradient are above the curves of vanilla policy gradient, no matter the environment is one-step or two-steps.

This observation verifies the advantage of the soft version of policy gradient.

## 4.4 Experimental analysis of combinations of attributes

We evaluate five variants of the proposed method, called $M_b$ (*bangs*), $M_l$ (*lighting*), $M_{bs}$ (*bangs & smile*), $M_{ll}$ (*lighting & lipstick color*), and $M_{aes}$ (*joint control over full five attributes*). Fig. 3 shows the qualitative comparison of the face images obtained by these five variants, which are tested over a total of 1,000 images, and six of which are presented as an illustration. Most images changed by these variants are distinct from the original ones. Moreover, they are all with promoted average aesthetics scores. We use FID to evaluate the quality of these images. The FID value of the image generated by $M_{aes}$ is 47.11, and the FID value of the initially generated image is 41.07, indicating that the attribute adjustment
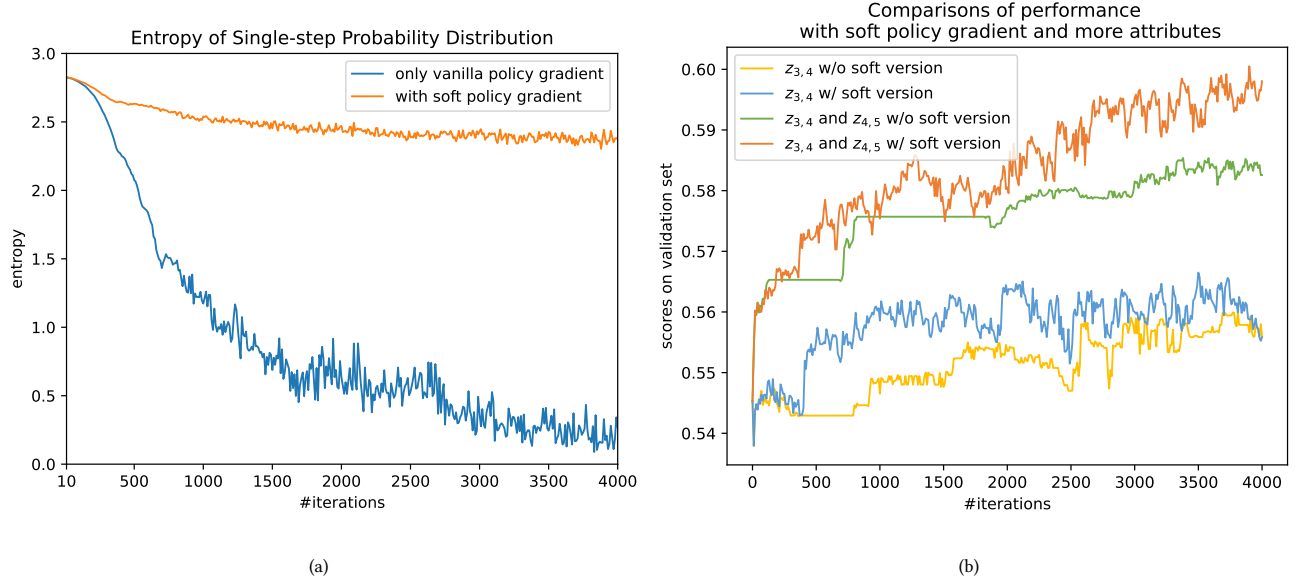
Figure 4: (a) Changing of entropy w.r.t. the number of iterations. (b) Performance w.r.t. the number of iterations when adjusting more attributes or using soft policy gradient.
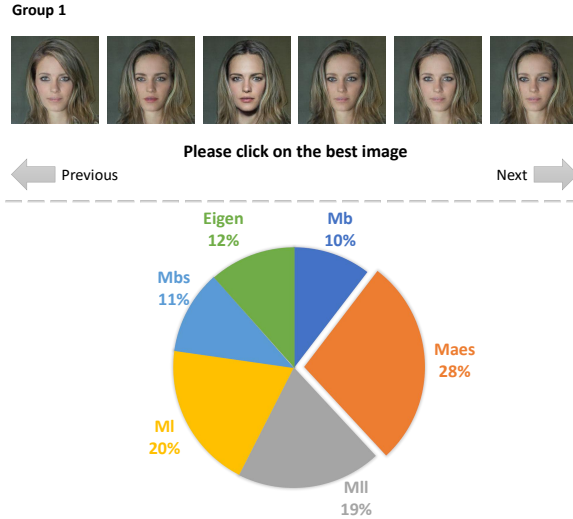
Figure 5: Top: Schematic of subjective experiment. Bottom: Participants' selection statistics for five variants of the proposed model. By fixing the latent variables, these five variants, namely $M_{\mathbf{b}}$, $M_{\mathbf{l}}$, $M_{\mathbf{bs}}$, $M_{\mathbf{ll}}$, and $M_{\mathbf{aes}}$, generate 100 groups, 600 images in total. Participants chose the best-looking among each group. $M_{\mathbf{aes}}$ receives the highest votes.

does not make the quality better, just makes them more beautiful. Also, Fig. 3 shows that controlling the brightness achieves a higher average score than altering the bangs, which indicates that the enhancement differs with distinct semantics. In the fourth row, we observe mouthes of the faces change significantly, from bright

smiles to slightly pursed mouthes. Intuitively, as grinning faces have been scarce in SCUT-FBP5500, this might give misleading reward signals to the RL agent. In addition, as Asian faces dominate SCUT-FBP5500 up to 72%, the aesthetics score network tends to prefer darker hair colors, and hence we remove this semantic in $M_{\text{aes}}$. Also, we observe $M_{\text{aes}}$ alters the images in a way, not via a straightforward composing of the changes in each separate single-step model. For example, for the first column, the image generated by $M_{\text{aes}}$ is different from several other variant models in terms of bangs, lighting, lipstick color, body pose, and gaze. For $M_{\text{aes}}$, we observe that for lighting, it tends to choose the butterfly light, which highlights the characters and reduces the environmental impact. Some changes are not in line with common perceptions, such as some male faces choosing reddish lipstick colors, partly because the generator lacks gender perception.

## 4.5 Subjective analysis

Because of the subjectivity of image aesthetics, humans show diverse characteristics in their aesthetic thinking. Even on commonly viewed attractive ones, people might differ subtly in their preferences. We, therefore, conduct a subjective comparative experiment on the five variants presented in the previous section. We recruit eight users to participate in the subjective study. All users do not know the experiment contents in advance. In the subjective study, we compared the face images obtained by EigenGAN and our five variants, namely $M_{\text{b}}$, $M_{\text{l}}$, $M_{\text{bs}}$, $M_{\text{ll}}$, and $M_{\text{aes}}$. We show 100 groups of images to all participants. The images generated by the same latent variables are a group, and we display each group of images in random order. The user selects the best image by clicking the screen without limiting the selection time. Fig. 5 shows that $M_{\text{aes}}$

has obtained the most votes, which indicates its effective controlling of various attributes of the face.

## 5 CONCLUSIONS

In this paper, by applying a novel variant of the ingredients in RL, we propose an RL-based model which can learn to precisely control the semantics underlying the generator of EigenGAN. We experimentally show the resulting model can alter the images in an aesthetically more attractive way. In future work, we will verify the efficacy of the proposed method on real-world image editing. A straightforward way for this task is to first recover the latent codes in EigenGAN corresponding to the input images.

**Limitations.** The work presented in this paper largely depends on how well EigenGAN can discover disentangled semantics and can reveal these interpretable factors explicitly in the generators. However, we have to manually observe the continuous changes by traversing the coefficient of specific eigen-dimensions. This step helps us filter out these semantics (such as gender, age, race, etc.) that drastically alter the identities of the original images. Another drawback of the method here is the quality of RL policy largely depends on the facial beauty prediction network, this might misleading the RL agent.

## ACKNOWLEDGMENTS

## REFERENCES

[1] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. 2016. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*. Curran Associates Inc., Red Hook, NY, USA, 2180–2188.

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 248–255.

[3] Nir Diamant, Dean Zadok, Chaim Baskin, Eli Schwartz, and Alexander M. Bronstein. 2019. Beholder-GAN: Generation and Beautification of Facial Images with Conditioning on Their Beauty Level. *CoRR* abs/1902.02593 (2019). arXiv:1902.02593 http://arxiv.org/abs/1902.02593

[4] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative Adversarial Nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2* (Montreal, Canada) *(NIPS'14)*. MIT Press, Cambridge, MA, USA, 2672–2680.

[5] Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. 2017. Reinforcement learning with deep energy-based policies. In *International Conference on Machine Learning*. PMLR, 1352–1361.

[6] Zhenliang He, Meina Kan, and Shiguang Shan. 2021. EigenGAN: Layer-Wise Eigen-Learning for GANs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, Montreal, QC, Canada, 14408–14417.

[7] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. 2018. Progressive Growing of GANs for Improved Quality, Stability, and Variation. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net. https://openreview.net/forum?id=Hk99zCeAb

[8] Tero Karras, Samuli Laine, and Timo Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE, Long Beach, CA, USA, 4401–4410.

[9] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1412.6980

[10] Debang Li, Huikai Wu, Junge Zhang, and Kaiqi Huang. 2018. A2-RL: Aesthetics Aware Reinforcement Learning for Image Cropping. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8193–8201. https://doi.org/10.1109/CVPR.2018.00855

[11] Debang Li, Huikai Wu, Junge Zhang, and Kaiqi Huang. 2019. Fast A3RL: Aesthetics-Aware Adversarial Reinforcement Learning for Image Cropping. *IEEE Transactions on Image Processing* 28, 10 (2019), 5105–5120. https://doi.org/10.1109/TIP.2019.2914360

[12] Lingyu Liang, Luojun Lin, Lianwen Jin, Duorui Xie, and Mengru Li. 2018. Scut-fbp5500: A diverse benchmark dataset for multi-paradigm facial beauty prediction. In *2018 24th International Conference on Pattern Recognition (ICPR)*. IEEE, Beijing, China, 1598–1603.

[13] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net. https://openreview.net/forum?id=Bkg6RiCqY7

[14] Ruotian Luo. 2020. A Better Variant of Self-Critical Sequence Training. *CoRR* abs/2003.09971 (2020). arXiv:2003.09971 https://arxiv.org/abs/2003.09971

[15] MindSpore. 2022. https://www.mindspore.cn/.

[16] Mehdi Mirza and Simon Osindero. 2014. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784* (2014).

[17] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. 2016. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*. PMLR, 1928–1937.

[18] Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. 2020. X-linear attention networks for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10971–10980.

[19] Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-Critical Sequence Training for Image Captioning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Honolulu, HI, USA, 1179–1195.

[20] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[21] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. 2020. Interpreting the Latent Space of GANs for Semantic Face Editing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 9240–9249. https://doi.org/10.1109/CVPR42600.2020.00926

[22] Wenjie Shi, Shiji Song, and Cheng Wu. 2019. Soft Policy Gradient Method for Maximum Entropy Deep Reinforcement Learning.. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. 3425–3431.

[23] Richard S. Sutton and Andrew G. Barto. 2018. *Reinforcement Learning: An Introduction.* A Bradford Book, Cambridge, MA, USA.

[24] Mingxing Tan and Quoc Le. 2019. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In *Proceedings of the 36th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 97)*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.). PMLR, Long Beach, California, USA, 6105–6114. https://proceedings.mlr.press/v97/tan19a.html

[25] Qilong Wang, Banggu Wu, Pengfei Zhu, Peihua Li, Wangmeng Zuo, and Qinghua Hu. 2020. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Seattle, WA, USA, 11531–11539. https://doi.org/10.1109/CVPR42600.2020.01155

[26] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. 2021. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5579–5588.

[27] Yuhongze Zhou and Qinjie Xiao. 2020. GAN-Based Facial Attractiveness Enhancement. *CoRR* abs/2006.02766 (2020). arXiv:2006.02766 https://arxiv.org/abs/2006.02766

[28] Brian D Ziebart. 2010. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy.* Carnegie Mellon University, Pittsburgh, PA 15213.