A. Proof

A.1. Proof of Theorem 3.1

Theorem 3.1 *Given the objective of Eq. (4), the optimal policy $p^*_\theta(x_{t-1}|x_t, z)$ has the following expression:*

$$p^*_\theta(x_{t-1}|x_t,z) \propto \exp\{\frac{Q^*(x_{t-1},z)+\alpha_2 \log p_{pre}(x_{t-1}|x_t,z)}{\alpha_1+\alpha_2}\},$$

where $Q^*(x_{t-1}, z) = r(x_0,z) I_{\{t-1=0\}} + (\alpha_1+\alpha_2) \cdot \log \sum_{x_{t-2}} \exp\{\frac{Q^*(x_{t-2},z)+\alpha_2 \log p_{pre}(x_{t-2}|x_{t-1},z)}{\alpha_1+\alpha_2}\}$.

Proof. According to [2, 20], we have

$$p_{\theta_k}(x_{t-1}|x_t, z) \propto \exp\{\frac{Q_k(x_{t-1}, z) + \alpha_2 \log p_{pre}(x_{t-1}|x_t, z) + \alpha_3 \log p_{\theta_{k-1}}(x_{t-1}|x_t, z)}{\alpha_1 + \alpha_2 + \alpha_3}\},$$

where $Q_k(x_{t-1}, z) = r(x_0,z) I_{\{t-1=0\}} + (\alpha_1+\alpha_2+\alpha_3) \log \sum_{x_{t-2}} \exp\{\frac{Q_k(x_{t-2},z)+\alpha_2 \log p_{pre}(x_{t-2}|x_{t-1},z)+\alpha_3 \log p_{\theta_{k-1}}(x_{t-2}|x_{t-1},z)}{\alpha_1+\alpha_2+\alpha_3}\}$.

When $t = 1$, we have

$$\begin{aligned}
\log p_{\theta_k}(x_0|x_1, z) &\propto \frac{r(x_0,z) + \alpha_2 \log p_{pre}(x_0|x_1, z) + \alpha_3 \log p_{\theta_{k-1}}(x_0|x_1, z)}{\alpha_1 + \alpha_2 + \alpha_3}, \\
\log p_{\theta_{k-1}}(x_0|x_1, z) &\propto \frac{r(x_0,z) + \alpha_2 \log p_{pre}(x_0|x_1, z) + \alpha_3 \log p_{\theta_{k-2}}(x_0|x_1, z)}{\alpha_1 + \alpha_2 + \alpha_3}, \\
&\dots \\
\log p_{\theta_1}(x_0|x_1, z) &\propto \frac{r(x_0,z) + \alpha_2 \log p_{pre}(x_0|x_1, z) + \alpha_3 \log p_{\theta_0}(x_0|x_1, z)}{\alpha_1 + \alpha_2 + \alpha_3}.
\end{aligned} \tag{7}$$

After simplifying the formula above, when $k \to \infty$, we have

$$p^*_\theta(x_0|x_1, z) \propto \exp\{\frac{r(x_0,z) + \alpha_2 \log p_{pre}(x_0|x_1, z)}{\alpha_1 + \alpha_2}\}.$$

The convergence of $p_{\theta_k}(x_{t-1}|x_t, z)$ is linked to the convergence of $Q_k(x_{t-1}, z)$, and the convergence of $Q_k(x_{t-1}, z)$ further depends on that of $p_{\theta_k}(x_{t-2}|x_{t-1}, z)$ ultimately tying the convergence of $p_{\theta_k}(x_{t-1}|x_t, z)$ to $p_{\theta_k}(x_{t-2}|x_{t-1}, z)$. Due to the convergence of $p^*_\theta(x_0|x_1, z)$, we can obtain the convergence of $p_{\theta_k}(x_0|x_1, z)$. Following a similar way to Eq. (7), this conclusion can be readily established.

□

Lemma A.1. *Defined a policy $\pi(a|s) \propto \exp\{\alpha A(s, a)\}$, we get that the entropy $H(\pi)$ is monotonically decreasing with respect to α.*

Proof. Defined $Z = \sum_a \exp\{\alpha A(s, a)\}$, we have

$$H(\pi) = \sum_a \frac{\exp\{\alpha A(s, a)\}}{Z} (-\log \frac{\exp\{\alpha A(s, a)\}}{Z}).$$

Taking the derivative of $H(\pi)$ with respect to α, we have

$$\begin{aligned}
H'(\pi) = &- \frac{\sum_a (\exp\{\alpha A(s, a)\} A(s, a) + \exp\{\alpha A(s, a)\} \alpha A(s, a)^2) Z}{Z^2} \\
&+ \frac{\sum_a (\exp\{\alpha A(s, a)\} \alpha A(s, a)) (\sum_a (\exp\{\alpha A(s, a)\} A(s, a)))}{Z^2} \\
&+ \frac{\sum_a (\exp\{\alpha A(s, a)\} A(s, a))}{Z} \\
=&\alpha \frac{(\sum_a (\exp\{\alpha A(s, a)\} A(s, a)) (\sum_a (\exp\{\alpha A(s, a)\} A(s, a)) - (\sum_a (\exp\{\alpha A(s, a)\} A(s, a)^2) (\sum_a (\exp\{\alpha A(s, a)\}))}{Z^2} \\
=&\frac{1}{2} \sum_{a_i, a_j} (\exp\{\alpha A(s, a_i) + \alpha A(s, a_j)\} (2 A(s, a_i) A(s, a_j) - A(s, a_i)^2 - A(s, a_j)^2) \\
\leq&0.
\end{aligned}$$

□

A.2. Proof of Theorem 3.2

Theorem 3.2 *When $\hat{\alpha}_1 \leq \alpha_1$, we have $H(p^*_\theta(x_{t-1}|x_t, z); \hat{\alpha}_1, \alpha_2) \leq H(p^*_\theta(x_{t-1}|x_t, z); \alpha_1, \alpha_2)$. Specially, compared to not including an entropy term, that is $\alpha_1 = 0$, in the objective function of Eq. (4), adding an entropy term results in a higher entropy after the algorithm converges, that is $H(p^*_\theta(x_{t-1}|x_t, z); \alpha_1, \alpha_2) \geq H(p^*_\theta(x_{t-1}|x_t, z); 0, \alpha_2)$.*

Proof. Since

$$\exp\{\frac{Q^*(x_{t-1}, z) + \alpha_2 \log p_{pre}(x_{t-1}|x_t, z)}{\hat{\alpha}_1 + \alpha_2}\} \geq \exp\{\frac{Q^*(x_{t-1}, z) + \alpha_2 \log p_{pre}(x_{t-1}|x_t, z)}{\alpha_1 + \alpha_2}\}.$$

According to Lemma A.1, this conclusion is easy to draw.

Adding an entropy term of Eq. (4), from Theorem 3.1, we get

$$p^*_\theta(x_{t-1}|x_t, z) \propto \exp\{\frac{Q^*(x_{t-1}, z) + \alpha_2 \log p_{pre}(x_{t-1}|x_t, z)}{\alpha_1 + \alpha_2}\}.$$

Without entropy term of Eq. (4), we get

$$\hat{p}^*_\theta(x_{t-1}|x_t, z) \propto \exp\{\frac{Q^*(x_{t-1}, z) + \alpha_2 \log p_{pre}(x_{t-1}|x_t, z)}{\alpha_2}\}.$$

According to Lemma A.1, we have $H(p^*_\theta(x_{t-1}|x_t, z); \alpha_1, \alpha_2) \geq H(p^*_\theta(x_{t-1}|x_t, z); 0, \alpha_2)$. □

A.3. Proof of Theorem 3.3

Theorem 3.3 *If $C = \min_{x_{t-1}, y_{t-1}}[\log p_{pre}(y_{t-1}|x_t, z) - \log p_{pre}(x_{t-1}|x_t, z)]/[Q^*(x_{t-1}, z) - Q^*(y_{t-1}, z)] > 0$, the objective function contains only a KL term and no entropy term, that is $\alpha_1 = 0$, if $\frac{1}{C} \leq \hat{\alpha}_2 \leq \alpha_2$, we have $H(p^*_\theta(x_{t-1}|x_t, z); 0, \hat{\alpha}_2) \geq H(p^*_\theta(x_{t-1}|x_t, z); 0, \alpha_2)$.*

Proof. Let $a \triangleq x_{t-1}$, $\alpha'_2 = \frac{1}{\alpha_2}$ and define $Z = \sum_a \exp\{\frac{Q^*(a,z) + \alpha_2 \log p_{pre}(a|x_t,z)}{\alpha_2}\} = \sum_a \exp\{\alpha'_2 Q^*(a, z) + \log p_{pre}(a|x_t, z)\}$. Since

$$H(p^*_\theta(a|x_t, z); 0, \alpha'_2) = -\frac{1}{Z}\sum_a \exp\{\alpha'_2 Q^*(a, z) + \log p_{pre}(a|x_t, z)\}(\alpha'_2 Q^*(a, z) + \log p_{pre}(a|x_t, z)) + \log Z.$$

Taking the derivative of $H(p^*_\theta(a|x_t, z); 0, \alpha'_2)$ with respect to α'_2, we have

$$\begin{aligned}
&\frac{\partial H(p^*_\theta(a|x_t, z); 0, \alpha'_2)}{\partial \alpha'_2} \\
=& -\frac{1}{Z^2}\sum_a \exp\{\alpha'_2 Q^*(a, z) + \log p_{pre}(a|x_t, z)\}(\alpha'_2 Q^*(a, z) + \log p_{pre}(a|x_t, z))Q^*(a, z) * Z \\
&+ \frac{1}{Z^2}\sum_a \exp\{\alpha'_2 Q^*(a, z) + \log p_{pre}(a|x_t, z)\}(\alpha'_2 Q^*(a, z) + \log p_{pre}(a|x_t, z))\cdot \\
&\sum_a \exp\{\alpha'_2 Q^*(a, z) + \log p_{pre}(a|x_t, z)\}Q^*(a, z) \\
=& \frac{1}{2}\sum_{a_i, a_j} \exp\{\alpha'_2 Q^*(a_i, z) + \log p_{pre}(a_i|x_t, z)\}\exp\{\alpha'_2 Q^*(a_j, z) \log p_{pre}(a_j|x_t, z)\}\cdot \\
&(Q^*(a_i, z) - Q^*(a_j.z))^2(-\alpha'_2 + \frac{\log p_{pre}(a_i|x_t, z) - \log p_{pre}(a_j|x_t, z)}{Q^*(a_j, z) - Q^*(a_i.z)}).
\end{aligned}$$

Therefore, $H(p^*_\theta(a|x_t, z); 0, \alpha'_2)$ is monotonically inceasing with respect to $\alpha'_2 \in [0, C]$. Based on the relationship between α_2 and α'_2, it is straightforward to prove this conclusion. □

Algorithm 1: Algorithm of AdaEnt

```
Input: Prompt set: $\mathcal{P}$; Training epoch: $\mathcal{E}$; Denoising step: $\mathcal{T}$.
Initialize pretrained diffusion model $\epsilon_\theta$ and prediction model $f_\phi$;
for $e = 1$ to $\mathcal{E}$ do
    for Prompt $p$ in $\mathcal{P}$ do
        // generate sample trajectory of $p$ iteratively:
        $\{\mathbf{x}^p_{\mathcal{T}-1}, ..., \mathbf{x}^p_0\} = \{\mu(\mathbf{x}^p_{\mathcal{T}}, t) + \sigma_{\mathcal{T}}\mathbf{z}, ..., \mu(\mathbf{x}^p_1, 1) + \sigma_1\mathbf{z}\}$.
        // compute reward:
        $R = r(\mathbf{x}^p_0, \mathbf{z})$.
        for Timestep $t$ in reversed $\mathcal{T}$ do
            // perform one step of denoising:
            $\mathbf{x}^p_{t-1} = \mu(\mathbf{x}^p_t, t) + \sigma_t\mathbf{z}$.
            // predicting $\gamma$:
            $\gamma_{t-1} = f_\phi(\mathbf{x}^p_{t-1})$.
            // dynamic stop:
            if $\gamma_{t-1} \geq \Delta$ then
                break.
            // compute KL:
            $KL_{t-1} = \frac{d}{2}\left[\frac{d\sigma^2_{t-1} + \|\mu_\theta - \hat{\mu}\|^2}{d\hat{\sigma}^2_{t-1}} + \ln\frac{\hat{\sigma}^2_{t-1}}{\sigma^2_{t-1}}\right] - \frac{d}{2}$.
            // compute Entropy:
            $H_{t-1} = \frac{d}{2}\ln\left(2\pi e\sigma^2_{t-1}\right)$.
            optimize $\epsilon_\theta$ according to Eq. (5) and PPO.
        optimize $f_\phi$ according to Eq. (6).
Output: learned model parameter $\theta$.
```

B. AdaEnt Algorithm

The procedure of the AdaEnt method is shown in Algorithm 1.

C. Experiment Supplementary

C.1. Experiment Details

In this paper, all models are trained with usually 100 epochs and batch size 64, except using aesthetic score=7 as the terminal epoch in the computation efficiency experiment. We adopt Adam optimizer with $\beta_1 = 0$ and $\beta_2 = 0.99$ and the learning rate is set to 0.0003.

The image classifier $f(\cdot)$ is a simple multi-layer CNN network, with only four convolution blocks (kernels [3,3,3,1]) and a linear layer. $f(\cdot)$ is adaptively trained based on the current RL-finetuning samples $\mathbf{x}_t$, where we take $t \in [T-5, T]$ as negative samples and $t \in [0, 5]$ as positive samples. Denoising is terminated when $f(x_t)$ indicates that x_t is approximated to the clean image (*i.e.*, $\Delta < 1e^{-8}$).

C.2. Analysis of Computational Cost

As $f(\cdot)$ is lightweight, the introduced computation overhead is trivial. In Sec. 5.6, we have discussed the computational efficiency of AdaEnt. In Fig. 3(b), the denoising steps in training drops from 50 to less than 30, corresponding to 40% reduction. Together with it, the learning efficiency of AdaEnt is also optimized since much fewer total training steps are required for the same reward score, leading to reduced overall training time (Tab. 3 left). In Tab. 3 right, the overhead of entropy and adjustment is light (GFLOPS ↑ less than 1%).

C.3. Explanation of Chosen Metrics

In this study, we evaluate image quality and relevance using several widely adopted metrics, including **ImageReward**, **Inception Score (IS)**, **PickScore**, and **ClipScore**. Below, we provide a brief description of each metric, which substantially demonstrates that the results in Tab. 1 and Tab. 2 indicate the superior overall quality of our generated images.

- **ImageReward** [46]: ImageReward is a learning-based metric specifically designed to align image quality assessments with human preferences. It is trained on a large-scale dataset of human preference annotations, making it particularly effective in evaluating generative models and text-to-image synthesis. Compared to traditional metrics, ImageReward better reflects subjective image quality and semantic alignment.
- **Inception Score (IS)** [32]: IS is a commonly used metric for assessing the quality of generative models. It measures both the diversity and realism of generated images by evaluating the entropy of class labels predicted by an Inception network. Higher IS values indicate that the generated images contain meaningful and diverse content. However, IS has been criticized for its insensitivity to intra-class diversity and lack of direct alignment with human perception.
- **PickScore** [17]: PickScore is a deep learning-based metric designed to evaluate image-text alignment by predicting human preferences. It leverages a contrastive learning approach to determine how well an image corresponds to a given text prompt. PickScore demonstrates a strong correlation with human judgment and has been shown to outperform previous automatic metrics in evaluating image-text consistency.
- **ClipScore** [27]: ClipScore quantifies image-text alignment based on the CLIP model's cosine similarity between image and text embeddings. It serves as a fast and scalable approach to assess semantic relevance but may exhibit biases due to the CLIP model's pretraining data. While ClipScore is effective in measuring text-image alignment, it does not directly capture aesthetic quality or fine-grained visual details.

These metrics collectively provide a comprehensive evaluation of image generation performance, balancing aspects of realism, diversity, alignment, and human preference.

C.4. Discussion of Denoising Step Reduction

In training, omitting a few steps has a limited impact on the diffusion model. In inference, AdaEnt still generates promising images with superior quality. For demonstration, as shown in Fig. 9, the main content of generated images is determined in the early denoising stage, while most AdaEnt's truncations are conducted after 50%. It also indicates that the denoising is robust to the slight step variations. In Tab. 4, we further compare the inference results with dynamic truncation (Ours) and with fixed 32-step truncation (Baselines), where all models are trained to AesScore=7. The results prove the overall superior quality of images generated by AdaEnt.

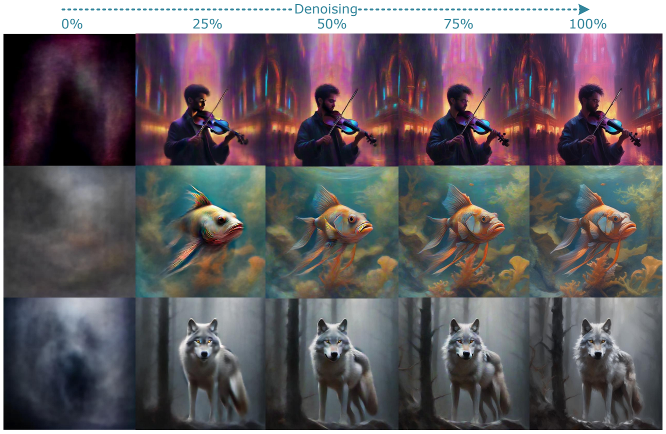

Figure 9. Generated images from different denoising stages.

Table 4. Overall Computation Efficiency of Baseline and Ours. TDS refers to total denoising steps. The left part and right part are the metrics of overall training and single-batch of denoising, respectively.

Method	Aes	CLIP	IS	PS
DDPO (32)	0.6675	0.2819	16.27	20.98
DPOK (32)	0.6537	0.2931	16.15	21.40
Ours (50)	0.7000	0.3010	20.58	21.41
Ours (ada.)	0.6739	0.2940	17.54	21.51

C.5. More Visual Impressions of Generated Images

Please refers to Fig. 10, 11, 12, and 13 along with their captions for more visual impressions and corresponding discussions.

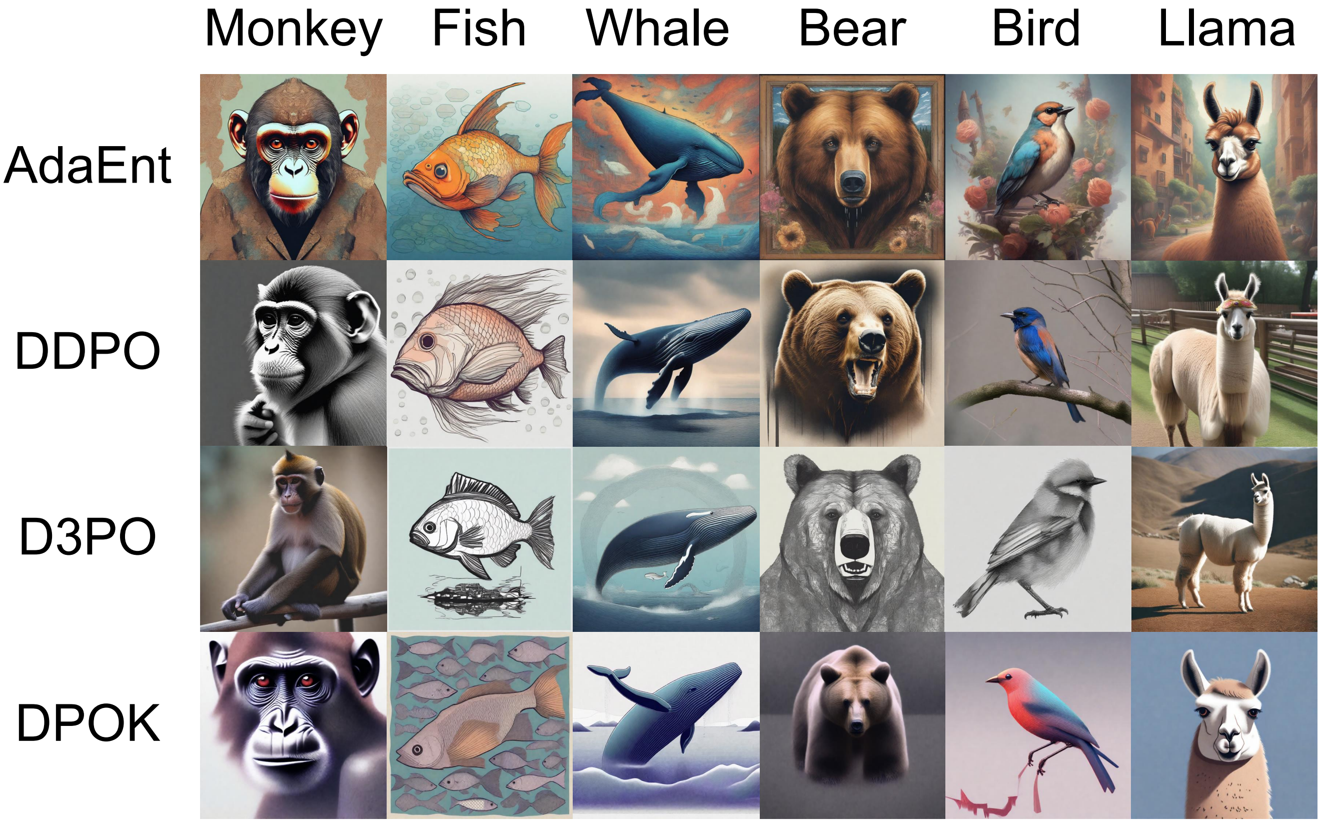

Figure 10. Visual comparison of models optimized for aesthetic style. Results are generated by SD-XL with simple-animal prompts trained on aesthetic scores. It can be observed that the proposed AdaEnt achieves the best aesthetic performance, with impressive artistic styles, rich colors, and fine textures.

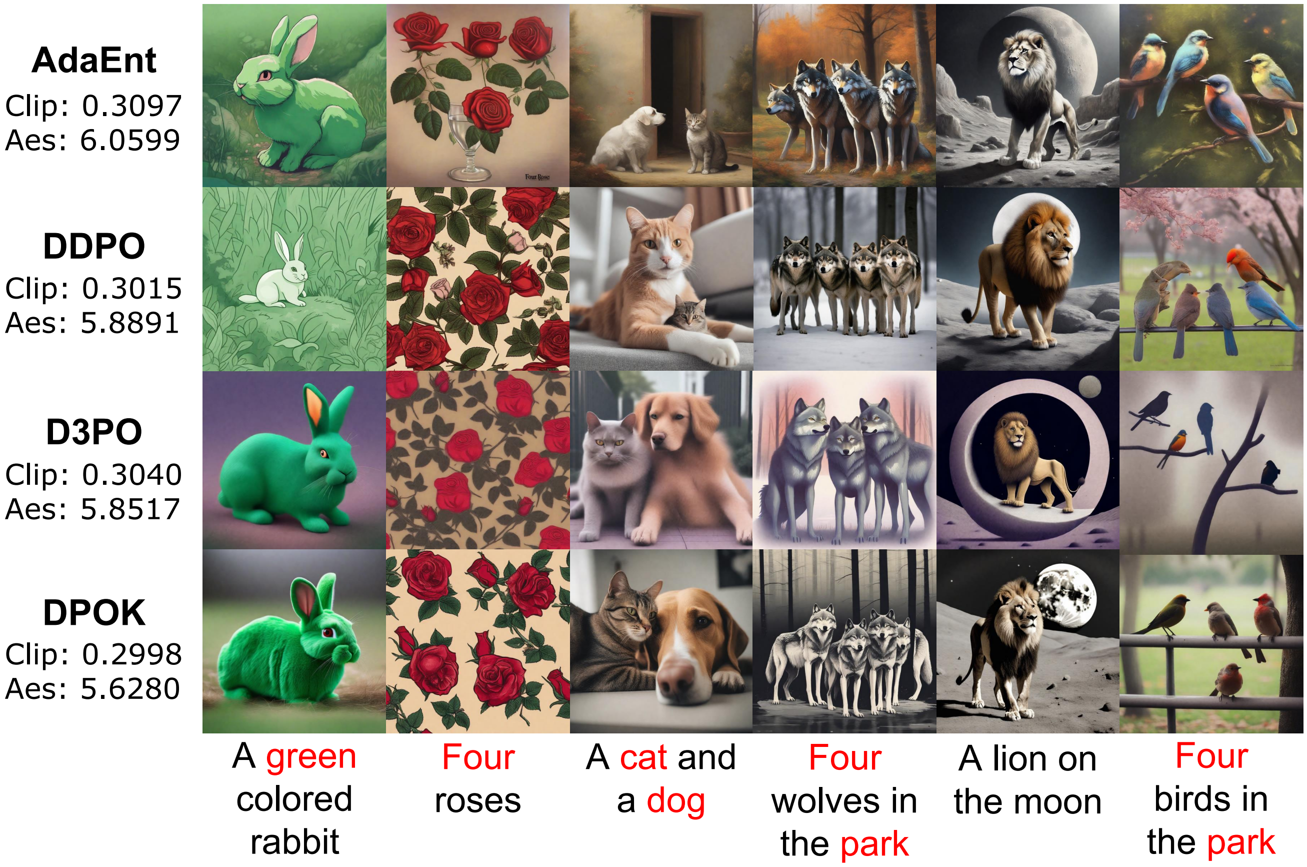

Figure 11. Visual comparison of models optimized for prompt-image alignment. We apply prompts with richer semantic-specific information to evaluate the prompt-image alignment performance. The metrics below each method's title are ClipScore and AestheticScore across these prompts with 30 random seeds, which could measure the aesthetic impressions and prompt-image alignment, respectively. Our method both qualitatively and quantitatively exhibits promising aesthetic performance with accurate alignment in color, number, composition, and location.

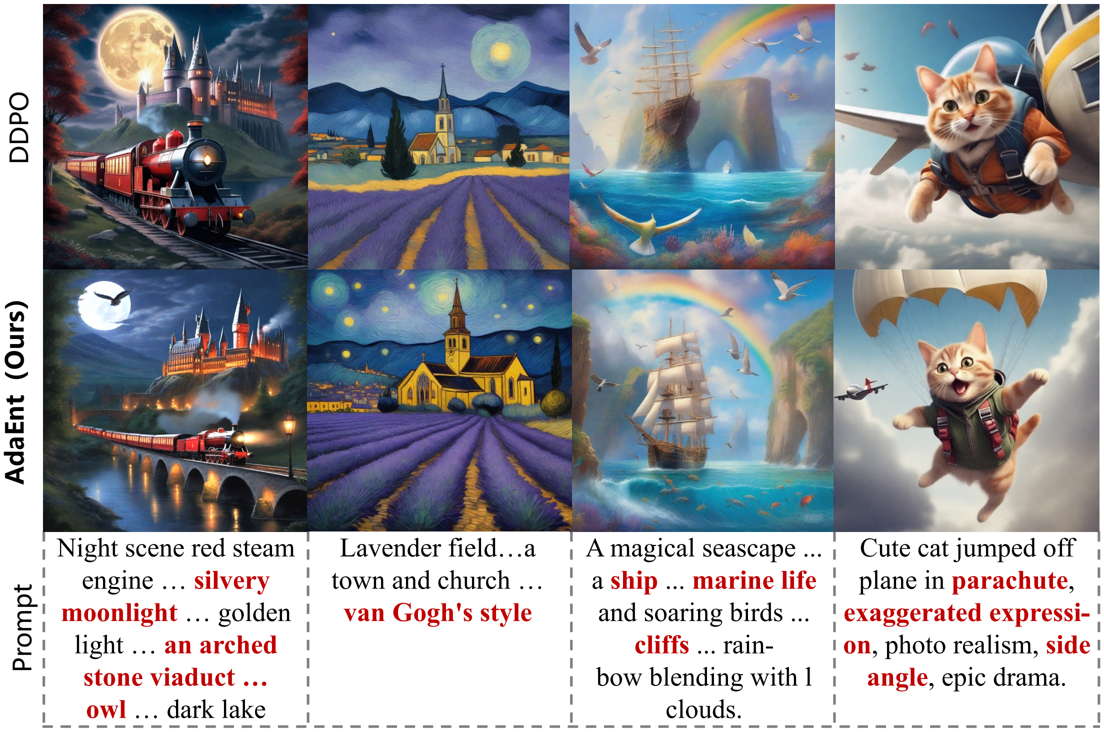
DDPO
AdaEnt (Ours)
Prompt
Night scene red steam engine … silvery moonlight … golden light … an arched stone viaduct … owl … dark lake
Lavender field…a town and church … van Gogh's style
A magical seascape ... a ship ... marine life and soaring birds ... cliffs ... rain-bow blending with l clouds.
Cute cat jumped off plane in parachute, exaggerated expressi-on, photo realism, side angle, epic drama.

Figure 12. Comparison of generated images from complex prompts after RL fine-tuning with Aesthetic Score on SD-XL. The disparity between prompts and baseline images is highlighted. Take the the second prompt as an example, while the aesthetic score of DDPO is higher than ours (7.2 versus 6.5), DDPO over-optimizes the reward and fails to generate the expected objects and styles.

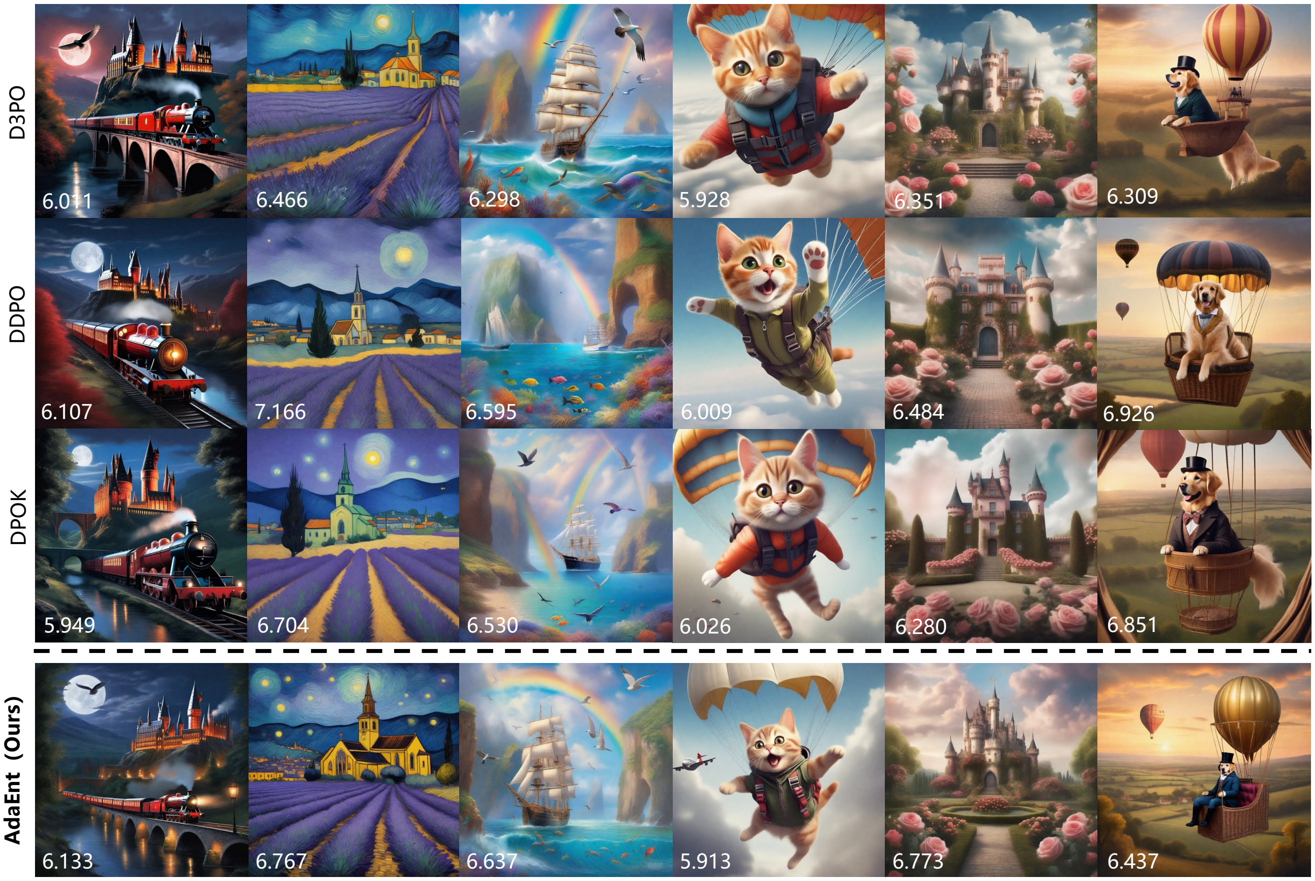

Figure 13. Visual comparison of our method and the baseline on complex prompts after training on the SD-XL model (training curves shown in Figure 6). The white numbers indicate the aesthetic scores of the generated images. It can be observed that although the baseline method achieves relatively high aesthetic scores for certain prompts, its generated results do not fully align semantically. In contrast, our method not only ensures precise semantic alignment but also attains higher aesthetic scores.