Personalized Preference Fine-tuning of Diffusion Models

Supplementary Material

A. Direct Preference Optimization on Diffusion Models

Given dataset containing examples $(\mathbf{c}, \mathbf{x}_0^+, \mathbf{x}_0^-)$, we define $r(\mathbf{c}, \mathbf{x}_0)$ as the reward on image $\mathbf{x}_0$ given prompt $\mathbf{c}$. We would like to fine-tune a text-to-image models $p_\theta(\mathbf{x}_0|\mathbf{c})$ such that reward is maximized while keeping close to a reference model $p_{\text{ref}}(\mathbf{x}_0|\mathbf{c})$ in terms of KL-divergence as regularization :

$$\max_{p_\theta} \mathbb{E}_{\mathbf{c},\mathbf{x}_0}\left[r(\mathbf{c}, \mathbf{x}_0)\right] - \beta \mathbb{D}_{\text{KL}}\left[p_\theta(\mathbf{x}_0|\mathbf{c}) \| p_{\text{ref}}(\mathbf{x}_0|\mathbf{c})\right], \tag{8}$$

where β is a parameter controlling how much $p_\theta(\mathbf{x}_0|\mathbf{c})$ deviates from $p_{\text{ref}}(\mathbf{x}_0|\mathbf{c})$.

We introduce latent variables $\mathbf{x}_{1:T}$ and define $R(\mathbf{c}, \mathbf{x}_{0:T})$ as the reward on the whole diffusion chain, such that we can define $r(\mathbf{c}, \mathbf{x}_0) = \mathbb{E}_{p_\theta(\mathbf{x}_{1:T}|\mathbf{x}_0,\mathbf{c})}[R(\mathbf{c}, \mathbf{x}_{0:T})]$. Given Eq. (8), we have

$$\begin{aligned}
&\min_{p_\theta} -\mathbb{E}_{p_\theta(\mathbf{x}_0|\mathbf{c})}\left[r(\mathbf{c}, \mathbf{x}_0)/\beta\right] + \mathbb{D}_{\text{KL}}\left[p_\theta(\mathbf{x}_0|\mathbf{c}) || p_{\text{ref}}(\mathbf{x}_0|\mathbf{c})\right] \\
\leq &\min_{p_\theta} -\mathbb{E}_{p_\theta(\mathbf{x}_0|\mathbf{c})}[r(\mathbf{c}, \mathbf{x}_0)/\beta] + \mathbb{D}_{\text{KL}}\left[p_\theta(\mathbf{x}_{0:T}|\mathbf{c}) || p_{\text{ref}}(\mathbf{x}_{0:T}|\mathbf{c})\right] \\
= &\min_{p_\theta} -\mathbb{E}_{p_\theta(\mathbf{x}_{0:T}|\mathbf{c})}[R(\mathbf{c}, \mathbf{x}_{0:T})/\beta] + \mathbb{D}_{\text{KL}}\left[p_\theta(\mathbf{x}_{0:T}|\mathbf{c}) || p_{\text{ref}}(\mathbf{x}_{0:T}|\mathbf{c})\right] \\
= &\min_{p_\theta} \mathbb{E}_{p_\theta(\mathbf{x}_{0:T}|\mathbf{c})}\left(\log \frac{p_\theta(\mathbf{x}_{0:T}|\mathbf{c})}{p_{\text{ref}}(\mathbf{x}_{0:T}|\mathbf{c}) \exp(R(\mathbf{c}, \mathbf{x}_{0:T})/\beta)/Z(\mathbf{c})} - \log Z(\mathbf{c})\right) \\
= &\min_{p_\theta} \mathbb{D}_{\text{KL}}\left[p_\theta(\mathbf{x}_{0:T}|\mathbf{c}) \| p_{\text{ref}}(\mathbf{x}_{0:T}|\mathbf{c}) \exp(R(\mathbf{c}, \mathbf{x}_{0:T})/\beta)/Z(\mathbf{c})\right].
\end{aligned} \tag{9}$$

where $Z(\mathbf{c}) = \sum_{\mathbf{x}} p_{\text{ref}}(\mathbf{x}_{0:T}|\mathbf{c}) \exp\left(r(\mathbf{c}, \mathbf{x}_0)/\beta\right)$ is the partition function. The optimal $p_\theta^*(\mathbf{x}_{0:T}|\mathbf{c})$ of Equation (9) has a unique closed-form solution:

$$p_\theta^*(\mathbf{x}_{0:T}|\mathbf{c}) = p_{\text{ref}}(\mathbf{x}_{0:T}|\mathbf{c}) \exp(R(\mathbf{c}, \mathbf{x}_{0:T})/\beta)/Z(\mathbf{c}),$$

Therefore, we have the reparameterization of reward function

$$R(\mathbf{c}, \mathbf{x}_{0:T}) = \beta \log \frac{p_\theta^*(\mathbf{x}_{0:T}|\mathbf{c})}{p_{\text{ref}}(\mathbf{x}_{0:T}|\mathbf{c})} + \beta \log Z(\mathbf{c}).$$

Plug this into the definition of r, hence we have

$$r(\mathbf{c}, \mathbf{x}_0) = \beta \mathbb{E}_{p_\theta(\mathbf{x}_{1:T}|\mathbf{x}_0,\mathbf{c})}\left[\log \frac{p_\theta^*(\mathbf{x}_{0:T}|\mathbf{c})}{p_{\text{ref}}(\mathbf{x}_{0:T}|\mathbf{c})}\right] + \beta \log Z(\mathbf{c}).$$

Substituting this reward reparameterization into maximum likelihood objective of the Bradly-Terry model, the partition function cancels for image pairs, and we get a maximum likelihood objective defined on diffusion models, for a single pair $(\mathbf{c}, \mathbf{x}_0^+, \mathbf{x}_0^-)$:

$$L_{\text{Diffusion-DPO}}(\theta) = -\log \sigma\left(\beta \mathbb{E}_{\mathbf{x}_{1:T}^+, \mathbf{x}_{1:T}^-}\left[\log \frac{p_\theta(\mathbf{x}_{0:T}^+|\mathbf{c})}{p_{\text{ref}}(\mathbf{x}_{0:T}^+|\mathbf{c})} - \log \frac{p_\theta(\mathbf{x}_{0:T}^-|\mathbf{c})}{p_{\text{ref}}(\mathbf{x}_{0:T}^-|\mathbf{c})}\right]\right)$$

where $\mathbf{x}_{1:T}^+ \sim p_\theta(\mathbf{x}_{1:T}|\mathbf{x}_0^+, \mathbf{c})$ and $\mathbf{x}_{1:T}^- \sim p_\theta(\mathbf{x}_{1:T}|\mathbf{x}_0^-, \mathbf{c})$. Since sampling from $p_\theta(\mathbf{x}_{1:T}|\mathbf{x}_0, \mathbf{c})$ is intractable, we utilize the forward process $q(\mathbf{x}_{1:T}|\mathbf{x}_0)$ for approximation.

$$L_{\text{approx}}(\theta) = -\log \sigma\left(\beta \mathbb{E}_{\mathbf{x}_{1:T}^+, \mathbf{x}_{1:T}^-}\left[\log \frac{p_\theta(\mathbf{x}_{0:T}^+)}{p_{\text{ref}}(\mathbf{x}_{0:T}^+)} - \log \frac{p_\theta(\mathbf{x}_{0:T}^-)}{p_{\text{ref}}(\mathbf{x}_{0:T}^-)}\right]\right) \tag{10}$$

where $\mathbf{x}^+_{1:T} \sim q(\mathbf{x}_{1:T}|\mathbf{x}^+_0, \mathbf{c}), \mathbf{x}^-_{1:T} \sim q(\mathbf{x}_{1:T}|\mathbf{x}^-_0, \mathbf{c})$. Therefore,

$$
\begin{aligned}
L_{\text{approx}}(\theta) &= -\log\sigma\left(\beta\mathbb{E}_{\mathbf{x}^+_{1:T},\mathbf{x}^-_{1:T}}\left[\sum_{t=1}^{T}\log\frac{p_\theta(\mathbf{x}^+_{t-1}|\mathbf{x}^+_t)}{p_{\text{ref}}(\mathbf{x}^+_{t-1}|\mathbf{x}^+_t)} - \log\frac{p_\theta(\mathbf{x}^-_{t-1}|\mathbf{x}^-_t)}{p_{\text{ref}}(\mathbf{x}^-_{t-1}|\mathbf{x}_t)}\right]\right) \\
&= -\log\sigma\left(\beta\mathbb{E}_{\mathbf{x}^+_{1:T},\mathbf{x}^-_{1:T}} T\mathbb{E}_t\left[\log\frac{p_\theta(\mathbf{x}^+_{t-1}|\mathbf{x}^+_t)}{p_{\text{ref}}(\mathbf{x}^+_{t-1}|\mathbf{x}^+_t)} - \log\frac{p_\theta(\mathbf{x}^-_{t-1}|\mathbf{x}^-_t)}{p_{\text{ref}}(\mathbf{x}^-_{t-1}|\mathbf{x}_t)}\right]\right) \\
&= -\log\sigma\left(\beta T\mathbb{E}_{t,\mathbf{x}^+_{t-1,t},\mathbf{x}^-_{t-1,t}}\left[\log\frac{p_\theta(\mathbf{x}^+_{t-1}|\mathbf{x}^+_t)}{p_{\text{ref}}(\mathbf{x}^+_{t-1}|\mathbf{x}^+_t)} - \log\frac{p_\theta(\mathbf{x}^-_{t-1}|\mathbf{x}^-_t)}{p_{\text{ref}}(\mathbf{x}^-_{t-1}|\mathbf{x}_t)}\right]\right) \\
&= -\log\sigma\left(\beta T\mathbb{E}_{t,\mathbf{x}^+_t,\mathbf{x}^-_t,\mathbf{x}^+_{t-1},\mathbf{x}^-_{t-1}}\left[\log\frac{p_\theta(\mathbf{x}^+_{t-1}|\mathbf{x}^+_t)}{p_{\text{ref}}(\mathbf{x}^+_{t-1}|\mathbf{x}^+_t)} - \log\frac{p_\theta(\mathbf{x}^-_{t-1}|\mathbf{x}^-_t)}{p_{\text{ref}}(\mathbf{x}^-_{t-1}|\mathbf{x}_t)}\right]\right)
\end{aligned}
\tag{11}
$$

where $\mathbf{x}^+_t \sim q(\mathbf{x}_t|\mathbf{x}^+_0), \mathbf{x}^-_t \sim q(\mathbf{x}_t|\mathbf{x}^-_0)$ and $\mathbf{x}^+_{t-1} \sim q(\mathbf{x}_{t-1}|\mathbf{x}^+_t, \mathbf{x}^+_0), \mathbf{x}^-_{t-1} \sim q(\mathbf{x}_{t-1}|\mathbf{x}^-_t, \mathbf{x}^-_0)$. Since function $-\log\sigma$ is a convex function, by Jensen's inequality, we can push $\mathbb{E}_{t,\mathbf{x}^+_t,\mathbf{x}^-_t}$ to the outside of $-\log\sigma$ and get an upper bound, therefore we have

$$
\begin{aligned}
L_{\text{approx}}(\theta) \leq &-\mathbb{E}_{t,\mathbf{x}^+_t,\mathbf{x}^-_t}\log\sigma\left(\beta T\mathbb{E}_{\mathbf{x}^+_{t-1},\mathbf{x}^-_{t-1}}\left[\log\frac{p_\theta(\mathbf{x}^+_{t-1}|\mathbf{x}^+_t)}{p_{\text{ref}}(\mathbf{x}^+_{t-1}|\mathbf{x}^+_t)} - \log\frac{p_\theta(\mathbf{x}^-_{t-1}|\mathbf{x}^-_t)}{p_{\text{ref}}(\mathbf{x}^-_{t-1}|\mathbf{x}_t)}\right]\right) \\
= &-\mathbb{E}_{t,\mathbf{x}^+_t,\mathbf{x}^-_t}\log\sigma\Bigg(-\beta T\Big(\big(\mathbb{D}_{\text{KL}}[q(\mathbf{x}^+_{t-1}|\mathbf{x}^+_{0,t})\|p_\theta(\mathbf{x}^+_{t-1}|\mathbf{x}^+_t)] - \mathbb{D}_{\text{KL}}[q(\mathbf{x}^+_{t-1}|\mathbf{x}^+_{0,t})\|p_{\text{ref}}(\mathbf{x}^+_{t-1}|\mathbf{x}^+_t)]\big) \\
&\qquad - \big(\mathbb{D}_{\text{KL}}[q(\mathbf{x}^-_{t-1}|\mathbf{x}^-_{0,t})\|p_\theta(\mathbf{x}^-_{t-1}|\mathbf{x}^-_t)] - \mathbb{D}_{\text{KL}}[q(\mathbf{x}^-_{t-1}|\mathbf{x}^-_{0,t})\|p_{\text{ref}}(\mathbf{x}^-_{t-1}|\mathbf{x}^-_t)]\big)\Big)\Bigg)
\end{aligned}
$$

Using the Gaussian parameterization of the reverse process (Eq. (1)), the above loss simplifies to:

$$
\begin{aligned}
L_{\text{approx}}(\theta) = -\mathbb{E}_{\mathbf{c},\mathbf{x}^+_0,\mathbf{x}^-_0}\log\sigma\Bigg(&-\beta T\omega(\lambda_t)\Big(\|\epsilon^+ - \epsilon_\theta(\mathbf{x}^+_t, \mathbf{c}, t)\|_2^2 - \|\epsilon^+ - \epsilon_{\text{ref}}(\mathbf{x}^+_t, \mathbf{c}, t)\|_2^2 \\
&- \big(\|\epsilon^- - \epsilon_\theta(\mathbf{x}^-_t, \mathbf{c}, t)\|_2^2 - \|\epsilon^- - \epsilon_{\text{ref}}(\mathbf{x}^-_t, \mathbf{c}, t)\|_2^2\big)\Big)\Bigg)
\end{aligned}
$$

where $\epsilon^+, \epsilon^- \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $\mathbf{x}^+_t \sim q(\mathbf{x}^+_t|\mathbf{x}_0)$, $\mathbf{x}^-_t \sim q(\mathbf{x}^-_t|\mathbf{x}_0)$, $\lambda_t = \alpha_t^2/\sigma_t^2$ is a signal-to-noise ratio term.

B. Experiments Details

B.1. Diffusion Model Fine-tuning Details

For the experiments shown in Sec. 5.1, we optimize only the added cross-attention layers, with 150M trainable parameters in total. All models are trained using the AdamW optimizer with an effective batch size of 768 pairs, a learning rate of 1×10^{-5}, and a single training epoch. The hyperparameter β is tuned within the range $[0.1, 2]$. Training is conducted on the Pick-a-Pic training set, with the best β selected based on the averaged rewards of generated samples evaluated on the Pick-a-Pic validation set, which contains 500 unique captions. Results are reported on the Partipromt dataset containing 1632 captions. For each method (each row in Tab. 1), β is tuned independently. Experiments are conducted on H100 GPUs with 80GB of memory. Using 8 GPUs, each with a local batch size of 16, training for one epoch (approximately 1000 gradient update steps) takes about 2 hours.

For the personalized real user experiments shown in Sec. 5.2, we use the same training settings as in Sec. 5.1 except for a reduced learning rate of 3×10^{-6}. We train on Pick-a-Pic training set, find the best hyper-parameter on Pick-a-Pic validation set and report results on Partiprompt dataset. As there are no automatic evaluation metrics to evaluate alignment with individualized user preferences, we find the best β based on the highest PickScore on the Pick-a-Pic validation set. We then run Diffusion-DPO with the same β as baseline.

B.2. Additional Details for Generating User Embeddings

As mentioned in Sec. 4.2, features from pre-trained VLM, LLaVA-OneVision [20], are constructed from $N = 4$ few-shot examples. To elicit these features, we employ Zero-Shot Chain of Thought Prompting (COT) [17, 46] to allow the model to reason about the images in a preference pair as well as generate a user profile. The prompt used for this COT can be found in Tab. 2. We then extract an embedding from the VLM from the last hidden state of the Qwen 2 Language Model in the LLaVA OneVision architecture for the final token it generates for the User Profile (Step 5 in the COT Assistant Prompt in Tab. 2). For sampling, we use a temperature of 0.7 with nucleus sampling probability of 1.0 (no nucleus sampling). As seen in Fig. 2, the top-k accuracy of a learned classifier from this frozen embedding is high, significantly outperforming random chance, indicating that this embedding is expressive, able to distinguish users within the pick-a-pick dataset. We additionally store the generated user profile for the baselines where the user profile is appended to the caption as an augmentation for the fairest comparison.

B.3. Additional Details for Scoring

We similarly employ COT for scoring. We add an additional assistant prompt for User Preference Prediction as found in Tab. 2. Here, we employ a stronger VLM as a Judge, GPT 4o-mini. For consistency, we present each pair of images twice to the model. In particular, for two images A and B, we ask the VLM to judge the images in two different permutations: first A then B and first B then A. We omit comparisons where the model isn't consistent in scoring (i.e A chosen for both comparisons or B chosen for both comparisons). For sampling, we utilize a temperature of 1.0 and nucleus sampling probability of 1.0 (no nucleus sampling). With COT and consistency, we find that we can match the preferences from real users in the Pick-a-Pic v2 dataset [16] with 83% accuracy.

B.4. Additional Details for Evaluation and Dataset Construction

Due to the restriction of fewshot prompting, we require at least $N = 4$ examples per user. Therefore, we drop users in the Pick-a-Pic v2 dataset where the number of preference pairs that the user labels are below N. For classification, we subsample 300 users with the most preference pairs to allow for more examples in classification.

System Prompt
You are an expert in image aesthetics and have been asked to predict which image a user would prefer based on the examples provided.
COT Assistant Prompt
You will be shown a few examples of preferred and dispreferred images that a user has labeled. Here is Pair 1: Here is the caption: [Caption for Pair 1] Here is Image 1: [Image 1] Here is Image 2: [Image 2] Prediction of user preference: [1 or 2] [...] Here is Pair 4: Here is the caption: [Caption for Pair 1] Here is Image 1: [Image 1] Here is Image 2: [Image 2] Prediction of user preference: [1 or 2] 1. Describe each image in terms of style, visual quality, and image aesthetics. 2. Explain the differences between the two images in terms of style, visual quality, and image aesthetics. 3. After you have described all of the images, summarize the differences between the preferred and dispreferred images into a user profile. Format your response as follows for the four pairs of images: Pair 1: Image 1: [Description] Image 2: [Description] Differences: [Description] [...] Pair 4: Image 1: [Description] Image 2: [Description] Differences: [Description] User Profile: [Description]
Additional Assistant Prompt for User Preference Prediction
Finally, you are provided with a new pair of images, unlabeled by the user. Your task is to predict which image the user would prefer based on the previous examples you have seen. Format your response as follows: Prediction of user preference: [1 or 2]

Table 2. Instructions for Embedding Generation and User Preference Prediction

C. More Qualitative Examples

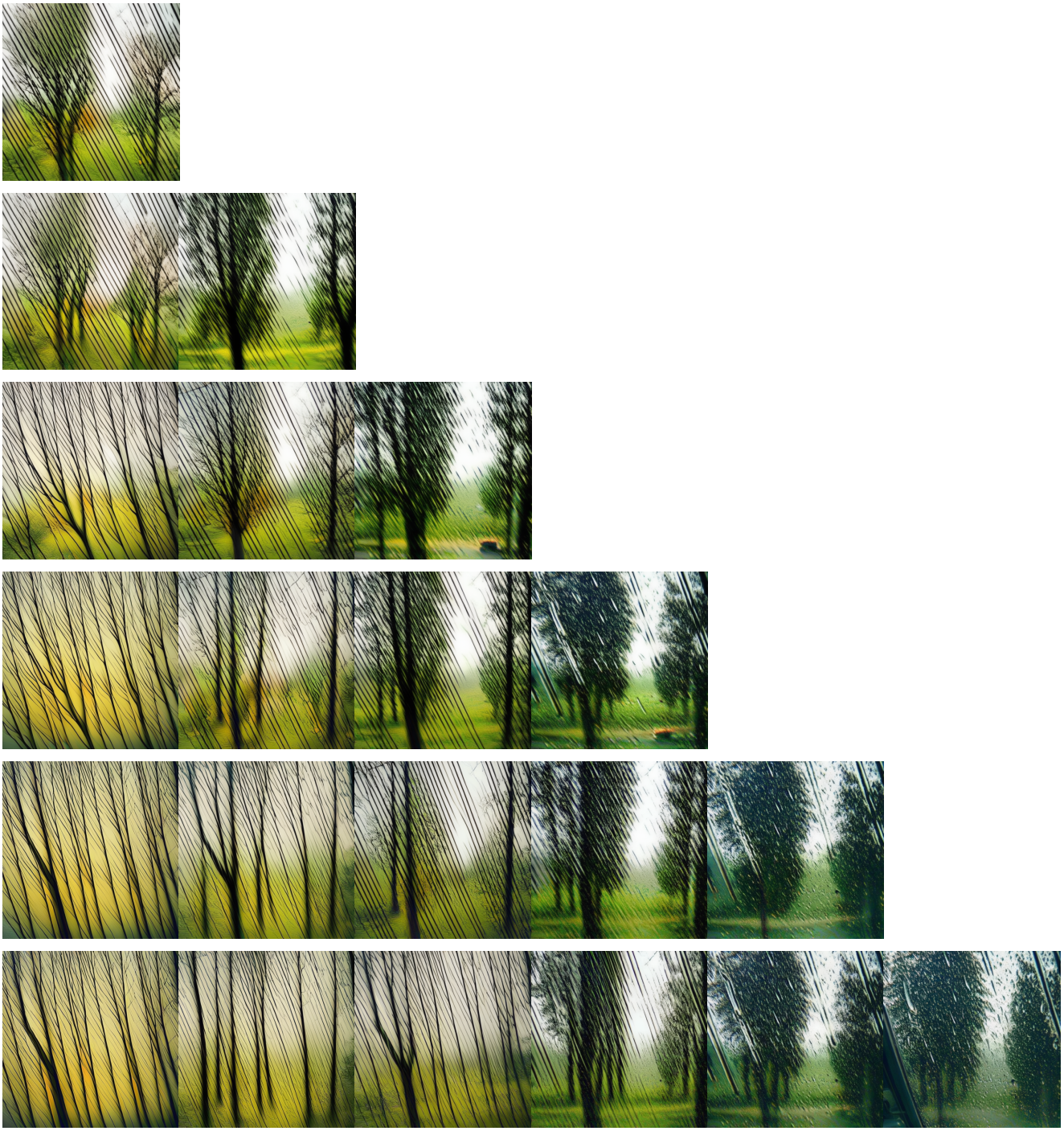
Similar to Fig. 4, the following figures show that PPD is able to interpolate among three distinct rewards during inference.

a cream-colored labradoodle wearing glasses and black beret teaching calculus at a blackboard

Figure 7

a black baseball hat with a flame decal on it

Figure 8



trees seen through a car window on a rainy day

Figure 9