Learning Preference Representations for Preference-Conditioned Image Generation

Wenyi Mo[1], **Tianyu Zhang**[2], **Ying Ba**[2],
Yalong Bai[2], **Ligong Han**[3,4], **Dimitris N. Metaxas**[1,✉]
[1]**Rutgers University** [2]**iN2X**
[3]**MIT-IBM Watson AI Lab** [4]**Red Hat AI Innovation**

Abstract

Modern image generative models produce high-quality, prompt-faithful images, yet the same prompt can correspond to different desired outputs for different users. We study *preference-conditioned image generation*: generating images that follow a target prompt while reflecting a user's latent visual preferences inferred from liked and disliked histories. This problem is challenging because real preference histories are sparse and costly to collect, user tastes mix semantic and stylistic factors that are difficult to verbalize, and diffusion generators are not naturally designed to condition on multi-image preference histories. We propose PREFGEN, a framework for learning structured preference representations and injecting them into diffusion-based generators. To address the supervision bottleneck, we construct a large-scale synthetic-agent preference dataset with dense, coherent liked/disliked histories, using it as scalable supervision for preference representation learning. We train a multimodal large language model with preference-oriented visual question answering and analyze its hidden states to separate two complementary signals: an intra-user embedding for liked-versus-disliked distinctions and a cross-history consistency embedding for stable user-level tendencies. To bridge MLLM representations and diffusion conditioning, we introduce a distributional alignment objective based on maximum mean discrepancy and inject the aligned preference signal through a lightweight cross-attention branch. We evaluate PREFGEN with a tiered protocol: PREFBENCH as a synthetic in-distribution diagnostic, leakage-free Pick-a-Pic transfer as a controlled real-user benchmark, and user-in-the-loop studies with participant-curated histories as human-facing evidence. Across these settings, PREFGEN improves preference alignment over strong personalization baselines while preserving prompt fidelity and competitive image quality.
Project Page: `https://prefgen.github.io`.

1 Introduction

Image generative models have advanced rapidly, achieving strong gains in visual realism, diversity, and prompt following [Esser et al., 2024, Labs et al., 2025, Wu et al., 2025, Wan et al., 2025]. As generation quality improves, a practical bottleneck is shifting from producing a plausible image to producing the image a particular user wants. The same prompt can admit many valid outputs: one user may prefer muted colors and minimal composition, while another may favor saturated tones, dense details, or a different rendering style. Such preferences are often difficult to specify exhaustively in language, so users still rely on repeated prompt editing [Hertz et al., 2023a, Mo et al., 2024], seed search [Zhou et al., 2025], and manual selection to obtain images that match their taste.

We study *preference-conditioned image generation*: given a target prompt and a user's liked/disliked generation histories, the model should infer the visual factors that distinguish preferred outputs from rejected ones and generate a prompt-faithful image aligned with that preference. This setting differs from single-reference style transfer. The goal is not to copy the appearance of one image, but to infer preference structure from multiple positive and negative examples and apply it to new prompts.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

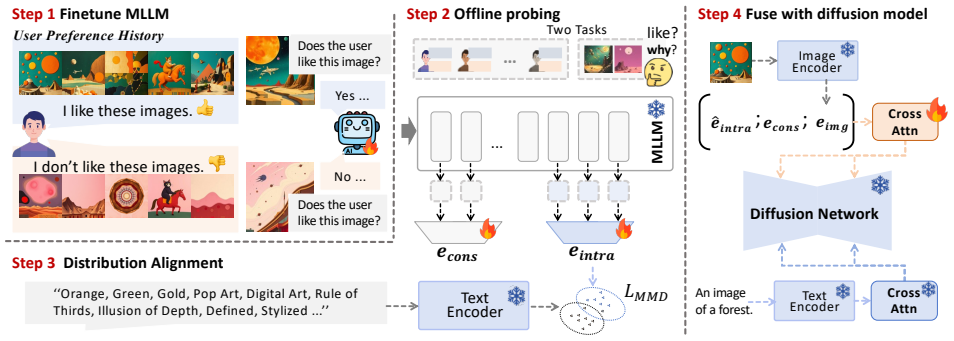


Figure 1: Overview of PREFGEN. We learn preference representations from liked/disliked histories using an MLLM, separate intra-user preference and cross-history user-consistency signals, align preference embeddings to the diffusion conditioning space with MMD, and inject the resulting signals through a lightweight cross-attention branch for preference-conditioned generation.

This problem remains challenging for three reasons. First, preference supervision is difficult to obtain at scale [Salehi et al., 2024]. Real user histories are often sparse, noisy, prompt-dependent, and inconsistently structured, making them insufficient for dense representation learning. Second, preferences are entangled with both semantic and stylistic factors: a liked image may differ from a disliked one in subject matter, composition, color palette, texture, or rendering style, and the relevant signal may vary across prompts. Third, even if a preference representation can be extracted from user histories, it is not naturally compatible with the conditioning interface of diffusion models [Song et al., 2024, Wang et al., 2025, Deng et al., 2025b]. Modern diffusion backbones are typically conditioned on text or image embeddings, whereas history-level preference embeddings learned by multimodal models may follow a different distribution. Naive injection can therefore be ignored by the generator, conflict with the target prompt, or cause semantic drift.

To address the supervision bottleneck, we construct a large-scale synthetic-agent preference corpus. Each agent provides coherent liked/disliked histories across prompts, enabling dense and controllable supervision for learning preference-sensitive representations. We do not treat synthetic agents as a replacement for real human taste distributions. Instead, we use them as structured training supervision and explicitly evaluate whether the learned representations transfer beyond the synthetic setting through real preference data and human-facing evaluation protocols.

Building on this corpus, we propose PREFGEN, a framework for preference-conditioned image generation. We first fine-tune a multimodal large language model (MLLM) [Li et al., 2024, Liu et al., 2024] with preference-oriented visual question answering over liked/disliked histories. Rather than compressing the MLLM hidden states into a single undifferentiated user code, we analyze them to extract two complementary signals. The first is an intra-user preference embedding, which captures liked-versus-disliked distinctions within a user's history. The second is a user-consistency embedding, which captures stable same-user tendencies across disjoint histories. A remaining challenge is to make these MLLM-derived preference signals usable by a diffusion generator. Directly injecting them into a text-conditioned diffusion backbone can be unstable because their distribution may not match the conditioning space expected by the generator. We therefore introduce a distributional alignment objective based on maximum mean discrepancy(MMD) [Gretton et al., 2012, Li et al., 2015] that maps preference embeddings toward the diffusion text-conditioning space at the distribution level. Compared with rigid point-wise alignment, this softer constraint improves compatibility while preserving user-specific variation. The aligned preference embedding, together with the user-consistency signal and image-level features, is injected through a lightweight cross-attention branch, enabling preference control while preserving prompt fidelity.

Our contributions are summarized as follows:

- We formulate preference-conditioned image generation from liked/disliked histories and construct a synthetic-agent preference corpus that provides supervision for preference representation learning.

- We propose a hierarchical MLLM-based representation that separates intra-user liked-versus-disliked preference cues from cross-history user-consistency signals.
- We introduce an MMD-based distributional alignment objective and a lightweight cross-attention injection mechanism to bridge preference representations and diffusion conditioning.
- We evaluate PREFGEN with a tiered protocol spanning controlled synthetic-agent diagnostics, transfer to real preference data, and human-facing preference evaluation.

2 Method

We present PREFGEN, a framework that generates preference-conditioned images from user preference feedback, as shown in Fig. 1. Our framework integrates multimodal preference modeling with conditional diffusion generation through three core innovations: (1) hierarchical preference embedding extraction via systematic MLLM analysis, (2) distribution-aligned preference conditioning, and (3) joint preference generation training.

2.1 Problem Formulation

We formulate preference-conditioned image generation as synthesizing an image that satisfies a text prompt while reflecting user-specific visual preferences. For a user u with a sparse preference history $\mathcal{H}_u = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, where $\mathbf{x}_i$ is an image, $y_i \in \{0, 1\}$ denotes whether the image is disliked or liked, and n is the number of the preference set. The goal is to generate an image $\mathbf{x}^*$ for a target prompt p that is both prompt-faithful and likely to be preferred: $\mathbf{x}^* = \arg\max_{\mathbf{x}} P(y = 1 \mid \mathbf{x}, p, \mathcal{H}_u)$. The preference history may contain stylistic, semantic, compositional, and color-related cues that are difficult to verbalize explicitly. We therefore seek a compact user representation $\mathbf{e}_u = f_\theta(\mathcal{H}_u)$ that preserves the discriminative visual signals needed to guide generation.

2.2 Multimodal User Preference Modeling

Preference-oriented MLLM training. We first strengthen the MLLM's ability to reason over user histories and candidate images. We construct a preference-oriented VQA dataset $\mathcal{D}_{\text{pref}} = \{(\mathcal{H}_u, \mathbf{q}_j^{(u)}, \mathbf{a}_j^{(u)}, \mathbf{x}_j^{(u)}, y_j^{(u)})\}$, where $\mathbf{x}_j^{(u)}$ is a candidate image, $\mathbf{q}_j^{(u)}$ asks whether user u would prefer this image, $y_j^{(u)}$ is the liked/disliked label, and $\mathbf{a}_j^{(u)}$ denotes the underlying preference attributes, such as artistic style, color palette, or composition. The MLLM is optimized with a cross-entropy loss over preference labels.[1] We denote the resulting preference-tuned model as PREFDISC.

Offline probing for preference signals. A central question is which MLLM hidden states should be used for generation conditioning. We therefore perform an offline, one-time probing analysis over different layer ranges and pooling strategies. The preference-tuned MLLM is frozen, and lightweight linear probes are trained to evaluate two complementary properties: (i) within-user preference discrimination, which tests whether an embedding distinguishes liked from disliked candidates for the same user; and (ii) cross-history user-consistency verification, which tests whether two disjoint histories express consistent preferences from the same user.

For within-user preference discrimination, each candidate image $\mathbf{x}_j^{(u)}$ is encoded together with the user history and query: $\mathbf{e}_j^{(u)} = f_\theta(\mathcal{H}_u, \mathbf{q}_j^{(u)}, \mathbf{x}_j^{(u)})$. We train a lightweight preference probe with binary cross-entropy, where $\hat{y}_j^{(u)} = \sigma(h_\phi(\mathbf{e}_j^{(u)}))$ denotes the predicted probability that user u likes candidate j:

$$\mathcal{L}_{\text{pref}} = -\frac{1}{|\mathcal{D}_{\text{intra}}|} \sum_{(\mathbf{e}_j^{(u)}, y_j^{(u)}) \in \mathcal{D}_{\text{intra}}} \left[y_j^{(u)} \log \hat{y}_j^{(u)} + (1 - y_j^{(u)}) \log(1 - \hat{y}_j^{(u)}) \right]. \tag{1}$$

For cross-history user-consistency verification, positive pairs are formed by splitting one user's history into two non-overlapping subsets, while negative pairs are sampled from different users. Given two histories $\mathcal{H}_a$ and $\mathcal{H}_b$, we obtain history-level embeddings $\mathbf{e}_a = f_\theta(\mathcal{H}_a)$ and $\mathbf{e}_b = f_\theta(\mathcal{H}_b)$, and

[1] We initially attempted to jointly predict preference labels and attribute descriptions, but this produced less stable outputs. We therefore use attribute descriptions as alignment targets rather than auxiliary decoding targets in this stage.

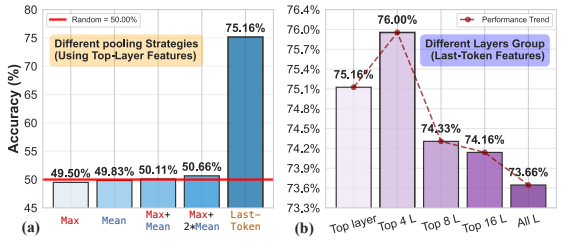


Figure 2: Study comparing different pooling strategies and layer selections for the preference discrimination task. The results show that using $\mathbf{e}_{\text{intra}}$ from the top four layers with the last-token strategy achieves the best performance in like–dislike discrimination.

Table 1: Pairwise same-user verification on 300 held-out users. We encode two disjoint preference histories and train a linear probe, with negative pairs drawn from different users.

Method	Acc (%)	F1 (%)	AUC (%)
CLIP avg	77.83	77.50	85.88
Top layer	66.00	50.49	88.72
Top 4 layers	74.33	77.55	86.42
Top 8 layers	77.17	73.19	88.35
Layers 8-20	**81.50**	**81.89**	**90.73**
All layers	67.33	53.11	88.67

construct pairwise interaction features $\mathbf{z}_{ab} = [\mathbf{e}_a; \mathbf{e}_b; |\mathbf{e}_a - \mathbf{e}_b|; \mathbf{e}_a \odot \mathbf{e}_b]$. A linear probe predicts whether the two histories come from the same user:

$$\mathcal{L}_{\text{same}} = -\frac{1}{|\mathcal{D}_{\text{inter}}|} \sum_{(\mathbf{z}_{ab}, s_{ab}) \in \mathcal{D}_{\text{inter}}} \left[s_{ab} \log \hat{s}_{ab} + (1 - s_{ab}) \log(1 - \hat{s}_{ab}) \right], \tag{2}$$

where $\hat{s}_{ab} = \sigma(g_\psi(\mathbf{z}_{ab}))$ and $s_{ab} \in \{0, 1\}$ indicates same-user versus different-user histories.

Layer probing insights. The probing results reveal a structured separation of preference-related signals. (1) Last-token pooling consistently outperforms mean, max, and hybrid pooling strategies, suggesting that the final token acts as a compact multimodal summary of the user-history reasoning process. (2) Embeddings from the top four layers perform best for within-user liked/disliked discrimination. We denote this signal as $\mathbf{e}_{\text{intra}}$, which captures context-dependent preference distinctions such as favored palette, rendering style, lighting, and composition. (3) Mid-upper layers, especially layers 8–20, perform best for cross-history same-user verification. We denote this signal as $\mathbf{e}_{\text{cons}}$, which captures more stable user-level consistency across disjoint histories. These findings determine the fixed layer ranges and pooling strategy used in all subsequent generation experiments.

Summary. Top layers yield the most discriminative embeddings for within-user preference prediction ($\mathbf{e}_{\text{intra}}$), mid-upper layers best support cross-session user consistency verification ($\mathbf{e}_{\text{cons}}$). Pooling with the `Last-Token` is consistently the most effective.

2.3 Distribution Alignment via Maximum Mean Discrepancy Loss

The MLLM-derived embeddings are not naturally compatible with pretrained diffusion backbones, which are typically conditioned on CLIP text embeddings. We therefore align the most generation-relevant preference signal, $\mathbf{e}_{\text{intra}}$, with the diffusion conditioning space. We focus on $\mathbf{e}_{\text{intra}}$ because it is the most discriminative signal for liked/disliked preference prediction and captures high-level semantic and stylistic attributes that overlap with language-describable concepts. In contrast, $\mathbf{e}_{\text{cons}}$ represents cross-history consistency and is less directly suited to text-space alignment.

Specifically, a 6-layer MLP maps $\mathbf{e}_{\text{intra}}$ to $\hat{\mathbf{e}}_{\text{intra}}$. We train this mapping with a Maximum Mean Discrepancy (MMD) loss [Gretton et al., 2012], using CLIP text embeddings of preference attributes as the target distribution. Let $\mathbf{e}_{\text{text}}$ denote the last-token CLIP text embedding of the attribute description $\mathbf{a}$[2]. Given batches of mapped preference embeddings $\{\hat{\mathbf{e}}_{\text{intra}}\}$ and text embeddings $\{\mathbf{e}_{\text{text}}\}$, the MMD loss is defined as:

$$\mathcal{L}_{\text{MMD}} = \mathbb{E}_{\hat{\mathbf{e}}_{\text{intra}}, \hat{\mathbf{e}}'_{\text{intra}}}[k(\hat{\mathbf{e}}_{\text{intra}}, \hat{\mathbf{e}}'_{\text{intra}})] + \mathbb{E}_{\mathbf{e}_{text}, \mathbf{e}'_{text}}[k(\mathbf{e}_{text}, \mathbf{e}'_{text})] - 2\,\mathbb{E}_{\hat{\mathbf{e}}_{\text{intra}}, \mathbf{e}_{text}}[k(\hat{\mathbf{e}}_{\text{intra}}, \mathbf{e}_{text})], \tag{3}$$

where $(\cdot)'$ denotes an independently sampled embedding from the same distribution, and $k(\cdot, \cdot)$ is a universal kernel. We adopt the Gaussian Kernel [Schölkopf and Smola, 2002].

Why distribution-level alignment? Point-wise objectives such as MSE or cosine similarity force each preference embedding to match a single textual target, which can over-constrain the representation and remove user-specific residual variation. A preference history usually reflects multiple visual factors, while a single attribute description only partially summarizes them. MMD instead regularizes the mapped preference embeddings toward the text-conditioning manifold at the distribution level, improving compatibility with diffusion conditioning while allowing soft correspondences.

[2]The CLIP text embeddings $\mathbf{e}_{\text{text}}$ are derived from the attribute descriptions $\mathbf{a}_j^{(u)}$ introduced in Sec. 2.2.

2.4 Conditional Generation with User Representation

After aligning the intra-user preference embedding, we further strengthen conditioning with three signals that serve different roles. $\mathbf{e}_{\text{img}}$ provides fine-grained visual anchoring from a CLIP-encoded *liked* reference image. $\hat{\mathbf{e}}_{\text{intra}}$ carries context-dependent like–dislike semantics after distributional mapping. $\mathbf{e}_{\text{cons}}$ supplies a stabilizing cross-history consistency signal that is auxiliary relative to single-image quality drivers. Empirically, $\mathbf{e}_{\text{img}}$ most strongly affects per-image coherence and swapping analyses (Sec. 3.4), and $\mathbf{e}_{\text{cons}}$ improves robustness across histories while leaving headline generation metrics primarily driven by the other branches. Training tuples, selection of $\mathbf{e}_{\text{img}}$, and guarding against leakage between conditioning and diffusion targets are specified in App. C.1.

Fusion strategies. The final user representation is constructed by concatenating the three signals: $\mathbf{e}_u = [\hat{\mathbf{e}}_{\text{intra}}; \mathbf{e}_{\text{cons}}; \mathbf{e}_{\text{img}}]$, preserving the ordering used in our backbone. `Concat` fuses $\mathbf{e}_{\text{img}}$ (visual/style anchor), $\hat{\mathbf{e}}_{\text{intra}}$ (preference semantics), and $\mathbf{e}_{\text{cons}}$ (user-level stabilizer) rather than treating them as interchangeable or equally decisive for each metric. We compared `Attn` (cross-attention), `Res-Attn` (attention with residual connection), and `Concat` (direct concatenation); direct concatenation proves more stable at our training scale (App. C.5, Fig. 9).

Conditioning mechanism. The concatenated representation $\mathbf{e}_u$ is injected into the generative backbone using the IP-Adapter framework [Ye et al., 2023]. Specifically, in each text cross-attention layer, we add a parallel user cross-attention branch: the query $\mathbf{Q}$ attends to both text-derived $(\mathbf{K}, \mathbf{V})$ and user-derived $(\mathbf{K}', \mathbf{V}')$, and the outputs are combined additively: $\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) + \text{Attention}(\mathbf{Q}, \mathbf{K}', \mathbf{V}')$.

Training objective. The overall training objective extends the standard score-matching loss:

$$\mathcal{L}_{total} = \mathbb{E}_{t,\mathbf{z}_0,\boldsymbol{\epsilon}} \left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t, p, \mathbf{e}_u)\|^2 \right], \tag{4}$$

where p denotes the text prompt, $\mathbf{z}_t$ is the noisy latent at timestep t, and $\boldsymbol{\epsilon}$ is the noise prediction target. In all main experiments, we instantiate our framework with SDXL [Podell et al., 2024] and an IP-Adapter-style [Ye et al., 2023] conditioning branch. Because the proposed user representations are introduced as conditioning tokens rather than by modifying the diffusion backbone itself, the same design can in principle be applied to other text-conditioned diffusion models [Esser et al., 2024, Black Forest Labs, 2024].

3 Experiments

We evaluate PREFGEN through three evidence tiers. **Tier 1: Synthetic in-distribution diagnostic (PREFBENCH).** This tier tests whether the model internalizes the structured preference protocol used for training. **Tier 2: Controlled real-user transfer (Pick-a-Pic).** On cleaned, held-out real-user annotations, we test whether representations learned from synthetic histories transfer to disjoint human preference signals. **Tier 3: Human-facing individualized evaluation.** In a user-in-the-loop study, the same participant supplies a preference history and performs blind pairwise judgments, which most closely matches our target problem. Expert and crowd studies provide complementary pooled evidence when histories are reconstructed by third-party annotators rather than curated by the judge.

3.1 Experimental Setup

Data and benchmarks. We train on a large-scale synthetic preference corpus containing $\sim$1M images from over 50K Claude-simulated users, with user-disjoint train/validation/test splits of 80/10/10. We use this corpus as dense and internally consistent supervision for learning preference-sensitive visual representations. PREFBENCH is constructed from held-out synthetic test users: we sample 136 users from the synthetic test split, and for each user build evaluation queries using a ten-image preference history consisting of five liked and five disliked images. It therefore serves as an in-distribution diagnostic for measuring whether models can follow the structured synthetic preference protocol. To evaluate transfer to real user preferences, we further construct a filtered, leakage-free Pick-a-Pic benchmark [Kirstain et al., 2023] with 200 preference sets. Each set contains five liked–disliked conditioning pairs and a disjoint evaluation image from the same user. Full details on dataset construction, filtering, and splits are provided in Appendix B.



Figure 3: Qualitative comparison across methods. Each row shows a user's preference history and outputs from different approaches. PREFGEN better preserves stylistic and semantic cues implied by the history while remaining prompt-faithful, whereas baselines more often trade off preference alignment against prompt fidelity.

Metrics. We group metrics by evaluation role. For image quality, we report FID [Heusel et al., 2017] and CMMD [Jayasumana et al., 2024], where lower is better. For reference and style alignment, we report CLIP image similarity [Radford et al., 2021] and CSD [Somepalli et al., 2024], where higher is better. For PREFBENCH, we additionally report PREFDISC, a discriminator trained to separate preferred from rejected images under the synthetic construction. For human-facing evaluation, we report blind pairwise win rates.

Baselines. We compare against reference-based personalization and style-conditioning methods, including IP-Adapter [Ye et al., 2023], InstantStyle [Wang et al., 2024], StyleAligned [Hertz et al., 2023b], ViPer [Salehi et al., 2024], EasyRef [Zong et al., 2024], and Bagel [Deng et al., 2025a]. We also include recent strong generative backbones, Qwen-Image [Wu et al., 2025] and Flux Kontext [Labs et al., 2025], where applicable. Per-baseline reference access, and backbone are summarized in App. C.7.

3.2 Main Results

3.2.1 Synthetic in-distribution diagnostic: PREFBENCH

Table 2 summarizes in-distribution performance on PREFBENCH. PREFGEN achieves the strongest overall trade-off: it obtains the lowest FID, ties for the best CMMD, and achieves the highest CLIP image, CSD, and PREFDISC scores. These results indicate that the proposed representation and conditioning mechanism better capture the structured preference signals defined by the synthetic protocol than reference-adapter and multi-image personalization baselines.

Fig. 3 provides qualitative comparisons under matched prompts and histories. PREFGEN more consistently transfers user-specific palettes, textures, and composition patterns from liked/disliked histories to new prompts. It transfers the recurring teal–orange palette and illustrative rendering to the dog scene in (a), vivid geometric color blocks to the train scene in (b), and the blue–purple ambience with warm focal accents to the fireplace scene in (c).

Table 2: Quantitative comparison on PREFBENCH. **Bold** indicates the best result.

Method	FID ↓	CMMD ↓	CLIP Img ↑	CSD ↑	PrefDisc ↑
IP-Adapter	172.06	**0.25**	67.88	50.22	81.65
InstantStyle	162.20	0.37	73.40	54.69	79.09
StyleAligned	167.54	0.31	67.89	46.88	79.71
ViPer	210.89	0.75	68.04	39.94	54.57
Bagel	148.92	0.33	72.39	50.84	66.82
EasyRef	157.48	0.26	74.23	54.62	69.19
Qwen-Image	166.03	0.37	63.23	44.32	81.15
Flux Kontext	151.66	0.36	67.30	51.69	78.16
PREFGEN (Ours)	**143.79**	**0.25**	**76.03**	**59.22**	**81.86**

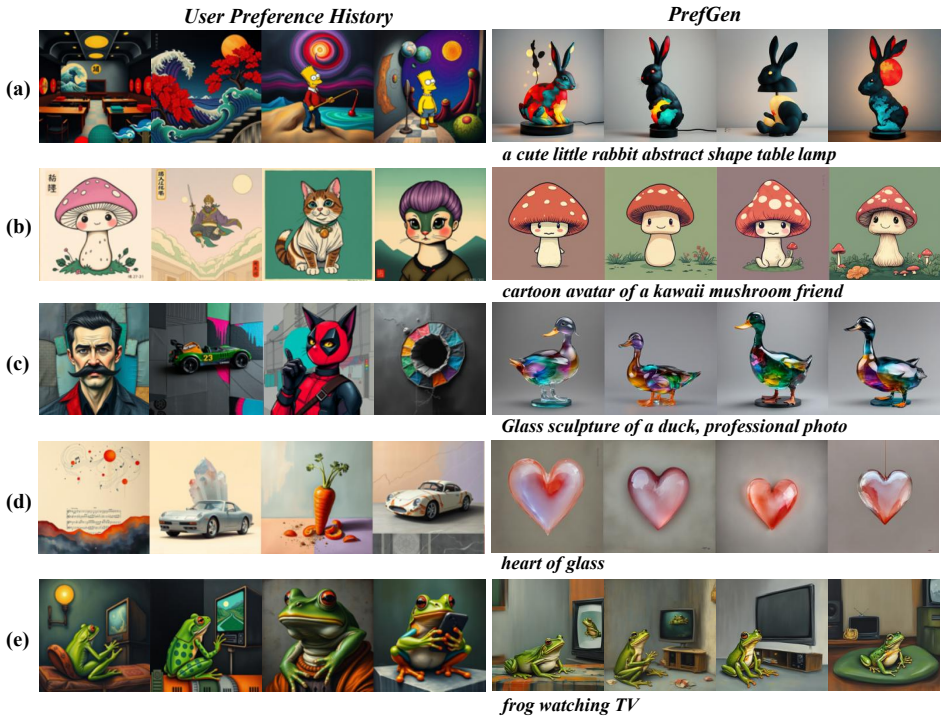


Figure 4: Application examples of PREFGEN. (a) Product design: PREFGEN integrates the color composition extracted from a user's preference history into the design of a rabbit-shaped lamp, aligning the generated output with user-specific aesthetic demands. (b) Character design: Given visual attribute specifications, PREFGEN generates characters with distinct background colors while preserving the desired attributes.

3.2.2 Controlled real-user transfer: Pick-a-Pic

Table 3 evaluates controlled transfer to held-out real-user annotations. PREFGEN achieves the best FID, CLIP image score, and CSD, while tying the best CMMD. This suggests that the learned preference representations capture visual cues that transfer beyond the synthetic-agent distribution. Importantly, PREFGEN is trained only on synthetic-agent preferences and never sees Pick-a-Pic annotations during generative training.

We also study whether directly mixing Pick-a-Pic annotations into training improves the preference discriminator. Pick-a-Pic-only and mixed training underperform synthetic-agent-only training under our objective, suggesting a structural mismatch between sparse pairwise crowd labels and the dense user-level histories required by our representation-learning formulation. We provide this analysis in App. D.7, together with filtering ablations in App. D.1.

Table 3: Quantitative results on a filtered Pick-a-Pic dataset with 200 preference sets.

Method	FID ↓	CMMD ↓	CLIP Img ↑	CSD ↑
IP-Adapter	188.34	**0.25**	0.767	54.15
InstantStyle	182.18	0.27	0.792	55.76
StyleAligned	190.95	**0.25**	0.730	45.54
ViPer	220.06	0.56	0.724	46.57
Bagel	179.95	**0.25**	0.757	53.25
EasyRef	183.25	0.26	0.797	58.22
PREFGEN (Ours)	**177.89**	**0.25**	**0.813**	**59.26**

Table 4: User-in-the-loop study.

Ours vs.	Win (%)
IP-Adapter	85.33
Bagel	71.67
EasyRef	87.00
ViPer	90.33

Table 5: Ablation on embeddings and alignment objectives.

$\mathbf{e}_{\text{img}}$	$\mathbf{e}_{\text{intra}}$	$\mathbf{e}_{\text{cons}}$	Align.	FID	CMMD	CLIP Img	CSD	Pref Disc
✓			-	151.65	0.64	0.73	45.03	59.77
✓	✓	✓	-	152.84	0.32	0.75	54.44	74.68
	✓		$\mathcal{L}_{\text{MMD}}$	163.35	0.41	0.64	31.16	55.98
	✓	✓	$\mathcal{L}_{\text{MMD}}$	159.72	0.41	0.69	37.47	56.83
✓	✓	✓	MSE+Cos	150.34	0.50	0.71	43.36	57.97
✓	✓	✓	$\mathcal{L}_{\text{MMD}}$	**143.79**	**0.25**	**0.76**	**59.22**	**81.86**

3.3 Human-Facing Evaluation

3.3.1 Individualized user-in-the-loop study

To more directly evaluate personalized generation, we conduct a user-in-the-loop study in which each participant both provides the preference history and judges the generated results. It provides a complementary test of whether the model can use sparse liked/disliked examples to adapt generation to an unseen user's own visual preferences.

Each participant first curates a personal history from a fixed candidate pool. We generate 50 stylistically diverse candidate images using FLUX and an author-written prompt bank fixed before participant recruitment. The prompts are designed to cover diverse visual attributes, including color palette, composition, rendering style, texture, subject category, and background complexity. Participants select five liked and five disliked images, forming a ten-image history $\mathcal{H}$ used only for preference elicitation and disjoint from all evaluation outputs. For evaluation, we use a separate set of ten author-written prompts $\mathcal{Q}$, fixed before the study and shared across all participants. For each participant and prompt $q \in \mathcal{Q}$, we generate one image from PREFGEN and each baseline under the same $(\mathcal{H}, q)$ conditioning. The outputs are presented in blind pairwise comparisons with randomized left/right placement and hidden method identities. Participants choose the image that better matches their own aesthetic preference given their selected history. Across 30 participants, 10 prompts, and 4 baselines, this protocol yields up to 1,200 pairwise judgments. Implementation details and interface screenshots are reported in App. D.17 and App. D.18.

Table 4 reports pooled participant preferences. PREFGEN is preferred over all compared baselines by margins substantially above chance. Because the participants who curate the histories are also the judges, this study provides the strongest human-facing evidence that the model can use sparse liked/disliked histories to personalize generation for unseen users.

3.3.2 Complementary expert and crowd studies

We additionally conduct pooled human studies with expert and crowd annotators. These panels use reconstructed histories and test whether third-party judges perceive preference consistency from a given history. Experts (N=15) and Prolific workers (N=50, 900 pairwise comparisons) judge which generation better matches the inferred preferences under the same prompt and history. The results in Tab. 7 show that PREFGEN is consistently preferred over baselines across both panels.

Table 6: Swapping-based analysis of the embeddings.

Method	CLIP Text (↑)	CSD (↑)
PREFGEN	25.83	59.22
Swap $\mathbf{e}_{\text{intra}}$	25.38	58.05
Swap $\mathbf{e}_{\text{img}}$	25.72	53.27
Swap $\mathbf{e}_{\text{cons}}$	25.52	58.10

Table 7: Complementary pooled human studies.

Ours vs. Baseline	Crowd (Prolific)	Experts
Bagel	78.67	68.38
EasyRef	88.00	63.60
InstantStyle	90.67	63.60
IP-Adapter	88.67	87.13
StyleAligned	88.67	63.24
ViPer	97.33	95.60

Table 8: Matched controls on PREFBENCH. (a) DirectHist conditions the same SDXL/IP-Adapter generator directly on pooled CLIP history features under the same training schedule and conditioning budget. (b) Variants retrained from scratch with a reduced 20k-step budget and identical architecture and token count; omitted signals are zeroed before projection.

(a) Direct history conditioning

Method	CSD↑	CLIP-I↑	PrefDisc↑	CMMD↓	FID↓
Like-only	56.9	0.75	79.56	0.254	150.0
Like+dislike	57.8	0.75	78.23	0.258	154.1
PREFGEN	**59.2**	**0.76**	**81.86**	**0.250**	**143.8**

(b) Component ablation (20k steps)

Variant	CSD↑	CLIP-I↑	PrefDisc↑	CMMD↓	FID↓
$\mathbf{e}_{\text{img}}$ only	41.8	0.73	56.9	0.652	158.2
$+\hat{\mathbf{e}}_{\text{intra}}$	52.8	0.73	57.0	0.301	155.4
$+\mathbf{e}_{\text{cons}}$	52.3	**0.74**	62.1	0.288	154.7
Full	**53.0**	**0.74**	**68.8**	**0.279**	**153.5**

3.4 Ablation Study

Direct history conditioning. To test whether an explicit preference representation is necessary, we add DirectHist, a fine-tuned baseline that conditions the same SDXL/IP-Adapter generator directly on mean-pooled CLIP features of the history, without the preference MLLM, layer probing, or MMD alignment (App. D.2). Its Like-only variant pools liked images, and its Like+dislike variant additionally subtracts pooled disliked features. Both variants are competitive, but PREFGEN performs best on all metrics (Table 8a); adding pooled negative-history features does not close the gap, supporting the value of explicitly learning the factors that distinguish liked from disliked examples.

Controlled component ablation. To isolate the contributions of $\hat{\mathbf{e}}_{\text{intra}}$ and $\mathbf{e}_{\text{cons}}$ on top of the image anchor, we retrain four variants from scratch under the same 20k-step budget, architecture, parameter count, and conditioning-token count, applying MMD whenever $\hat{\mathbf{e}}_{\text{intra}}$ is present. In Table 8b, the image anchor alone does not match the full representation: $\hat{\mathbf{e}}_{\text{intra}}$ mainly improves CSD and CMMD, whereas $\mathbf{e}_{\text{cons}}$ contributes more strongly to PREFDISC. The full representation is best or tied-best on every metric, indicating complementary signals.

Preference representations. Table 5 studies the contribution of the image anchor $\mathbf{e}_{\text{img}}$, the intra-user preference embedding $\mathbf{e}_{\text{intra}}$, and the user-consistency embedding $\mathbf{e}_{\text{cons}}$. Using only the image anchor provides a reasonable reference-based baseline, but adding MLLM-derived preference representations substantially improves preference-sensitive metrics. The full three-branch model achieves the best joint performance across FID, CMMD, CLIP image similarity, CSD, and PREFDISC, indicating that the learned preference embeddings complement rather than replace visual reference anchoring.

Distribution alignment. Without distribution alignment, the full embedding stack improves over image-only anchoring but remains weaker than the final model. In Table 5, replacing $\mathcal{L}_{\text{MMD}}$ with a pointwise MSE+Cos alignment degrades all preference-sensitive metrics. This supports our use of distribution-level alignment: rather than forcing a rigid one-to-one mapping between MLLM preference embeddings and text-encoder targets, MMD better matches the conditioning distribution expected by the diffusion backbone.

Consistency embedding and latent swapping. Swapping $\mathbf{e}_{\text{img}}$ causes the largest CSD drop, reflecting its role as the strongest visual anchor. Swapping $\mathbf{e}_{\text{intra}}$ mainly affects preference realization under the target prompt. In contrast, swapping $\mathbf{e}_{\text{cons}}$ leads to a smaller but consistent degradation, suggesting that it contributes complementary user-level information rather than directly determining local style or texture. This agrees with the full-branch ablation in Tab. 5, where the best performance is obtained when $\mathbf{e}_{\text{cons}}$ is combined with both $\mathbf{e}_{\text{img}}$ and $\mathbf{e}_{\text{intra}}$. The representation-level analysis in App. D.13 further supports that $\mathbf{e}_{\text{cons}}$ captures cross-history user consistency. At generation time,

Table 9: PREFBENCH metrics versus the number k of liked–disliked history pairs at inference.

k	1	3	5	6	7	8	9	10
FID ↓	146.73	146.23	143.79	143.69	145.29	143.79	144.69	144.89
CSD ↑	57.91	58.01	59.22	59.30	59.24	59.19	59.19	59.23
PrefDisc ↑	80.75	81.05	81.86	81.77	81.65	81.71	81.60	81.75

zeroing $\mathbf{e}_{\text{cons}}$ in the converged checkpoint lowers cross-history output consistency on PREFBENCH ($70.6 \rightarrow 69.7$, paired t=3.04, p<0.01) and held-out-liked alignment on Pick-a-Pic ($57.7 \rightarrow 57.3$, t=2.28, p<0.05; App. D.3), so we characterize $\mathbf{e}_{\text{cons}}$ as an auxiliary cross-history stabilizer.

Effect of history length. We further evaluate the sensitivity of PREFGEN to the number of reference examples available at inference time. Specifically, using the same converged checkpoint, we vary the number of liked–disliked reference pairs provided as user history from 1 to 10. As shown in Table 9, performance improves through approximately five to six pairs and then remains within a narrow range, suggesting that five to six pairs are sufficient. At the same time, PREFGEN remains effective even in the 1-shot setting. The degradation from 5-shot to 1-shot is moderate, and CLIP Img remains relatively stable, indicating that the model can extract useful preference cues from very sparse histories.

Portability and training-data scale. The learned representation is not tied to SDXL/IP-Adapter: with a frozen Bagel-7B, training only a lightweight projection (5k steps) that maps $[\hat{\mathbf{e}}_{\text{intra}}; \mathbf{e}_{\text{cons}}]$ into two native preference tokens improves CSD from 48.8 to 53.1 with text-only input and from 50.8 to 54.2 when Bagel also receives a reference image (App. D.4). Under a fixed 6k-step budget, increasing the synthetic training data from 10% to 100% steadily improves Pick-a-Pic CSD ($48.9 \rightarrow 52.5$) and FID ($188.2 \rightarrow 185.4$), with no saturation in the tested range (App. D.5).

Additional analyses. Fig. 4 sketches application motifs (design, characters, cloning). Semantic-bias analyses and temporal preferences are in App. D.15 and D.14. We report fusion-strategy comparisons, OOD discrimination on natural images and sketches, inference latency, GPU memory, and additional qualitative results in App. C.5, App. D.8, App. D.11, and App. D.9.

Limitations. Although PREFGEN demonstrates strong preference alignment across synthetic, real-user transfer, and user-in-the-loop evaluations, its training relies on structured synthetic-agent preferences that may not fully capture the noisiness, inconsistency, and temporal dynamics of real human tastes.

4 Conclusion

In this work, we introduced PREFGEN, a framework for preference-conditioned image generation that leverages multimodal large language models to extract user-specific representations and incorporates a distribution alignment mechanism to bridge the gap between multimodal embeddings and diffusion backbones. We learn preference-sensitive representations from large-scale structured synthetic-agent supervision and treat simulated agents as controllable training distributions; external validity is assessed through leakage-free Pick-a-Pic transfer, user-in-the-loop evaluation, and complementary human studies. Extensive experiments demonstrate that PREFGEN consistently surpasses strong personalization baselines in image quality and preference alignment under these protocols. Matched direct-history and component controls, together with transfer to a frozen native multimodal generator, further support the utility and portability of the learned preference representation.

Acknowledgments and Disclosure of Funding

This research was partially supported by grants to Dimitris N. Metaxas from the National Science Foundation (NSF 2310966, 2235405, 2212301, 2003874, and 1951890) and the National Institutes of Health (NIH 2R01HL127661).

Competing interests. The authors declare no other competing interests.

References

Ruichuan An, Sihan Yang, Renrui Zhang, Zijun Shen, Ming Lu, Gaole Dai, Hao Liang, Ziyu Guo, Shilin Yan, Yulin Luo, Bocheng Zou, Chaoqun Yang, and Wentao Zhang. Unictokens: Boosting personalized understanding and generation via unified concept tokens. *CoRR*, abs/2505.14671, 2025. doi: 10.48550/ARXIV.2505.14671. URL https://doi.org/10.48550/arXiv.2505.14671.

Anthropic. Claude, 2024. URL https://claude.ai. Large language model.

Black Forest Labs. FLUX.1-dev: High-performance image generation model, 2024. URL https://huggingface.co/black-forest-labs/FLUX.1-dev.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025.

Meihua Dang, Anikait Singh, Linqi Zhou, Stefano Ermon, and Jiaming Song. Personalized preference fine-tuning of diffusion models. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8020–8030, 2025. URL https://api.semanticscholar.org/CorpusID:275470693.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025a.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. IEEE Computer Society, 2009.

Yufan Deng, Xun Guo, Yizhi Wang, Jacob Zhiyuan Fang, Angtian Wang, Shenghai Yuan, Yiding Yang, Bo Liu, Haibin Huang, and Chongyang Ma. Cinema: Coherent multi-subject video generation via mllm-based guidance. *ArXiv*, abs/2503.10391, 2025b. URL https://api.semanticscholar.org/CorpusID:276961000.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. *arXiv preprint arXiv:2403.03206*, 2024.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In Evangelos Simoudis, Jiawei Han, and Usama M. Fayyad, editors, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, USA*, pages 226–231. AAAI Press, 1996. URL http://www.aaai.org/Library/KDD/1996/kdd96-037.php.

Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=NAQvF08TcyG.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander J. Smola. A kernel two-sample test. *J. Mach. Learn. Res.*, 13:723–773, 2012. doi: 10.5555/2503308.2188410. URL https://dl.acm.org/doi/10.5555/2503308.2188410.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross-attention control. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023a. URL https://openreview.net/forum?id=_CDixzkzeyb.

Amir Hertz, Andrey Voynov, Shlomi Fruchter, and Daniel Cohen-Or. Style aligned image generation via shared attention. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4775–4785, 2023b. URL https://api.semanticscholar.org/CorpusID:265608730.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6626–6637, 2017.

Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking FID: towards a better evaluation metric for image generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 9307–9315. IEEE, 2024. doi: 10.1109/CVPR52733.2024.00889. URL https://doi.org/10.1109/CVPR52733.2024.00889.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. FLUX.1 kontext: Flow matching for in-context image generation and editing in latent space. *CoRR*, abs/2506.15742, 2025. doi: 10.48550/ARXIV.2506.15742. URL https://doi.org/10.48550/arXiv.2506.15742.

Hugo Laurençon, Léo Tronchon, Matthieu Cord, and Victor Sanh. What matters when building vision-language models? *arXiv preprint arXiv:2405.02246*, 2024.

Zongxian Lee. Mmd_loss.pytorch: A pytorch implementation of maximum mean discrepancy (mmd) loss. https://github.com/ZongxianLee/MMD_Loss.Pytorch, 2020.

Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *CoRR*, abs/2407.07895, 2024. doi: 10.48550/ARXIV.2407.07895. URL https://doi.org/10.48550/arXiv.2407.07895.

Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22511–22521, 2023. URL https://api.semanticscholar.org/CorpusID:255942528.

Yujia Li, Kevin Swersky, and Richard S. Zemel. Generative moment matching networks. In Francis R. Bach and David M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1718–1727. JMLR.org, 2015. URL http://proceedings.mlr.press/v37/li15.html.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 740–755, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10602-1.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge. January 2024. URL https://llava-vl.github.io/blog/2024-01-30-llava-next/.

Jiancang Ma, Junhao Liang, Chen Chen, and H. Lu. Subject-diffusion: Open domain personalized text-to-image generation without test-time fine-tuning. *ACM SIGGRAPH 2024 Conference Papers*, 2023. URL https://api.semanticscholar.org/CorpusID:260091569.

Yiyang Ma, Xingchao Liu, Xiaokang Chen, Wen Liu, Chengyue Wu, Zhiyu Wu, Zizheng Pan, Zhenda Xie, Haowei Zhang, Xingkai yu, Liang Zhao, Yisong Wang, Jiaying Liu, and Chong Ruan. Janusflow: Harmonizing autoregression and rectified flow for unified multimodal understanding and generation, 2024.

J. MacQueen. Some methods for classification and analysis of multivariate observations. 1967. URL https://api.semanticscholar.org/CorpusID:6278891.

Wenyi Mo, Tianyu Zhang, Yalong Bai, Bing Su, Ji-Rong Wen, and Qing Yang. Dynamic prompt optimizing for text-to-image generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 26617–26626. IEEE, 2024. doi: 10.1109/CVPR52733.2024.02514. URL https://doi.org/10.1109/CVPR52733.2024.02514.

Wenyi Mo, Ying Ba, Tianyu Zhang, Yalong Bai, and Biye Li. Learning user preferences for image generation model, 2025. URL https://arxiv.org/abs/2508.08220.

Xuan Vinh Nguyen, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *J. Mach. Learn. Res.*, 11:2837–2854, 2010. URL https://api.semanticscholar.org/CorpusID:13962547.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael G. Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *CoRR*, abs/2304.07193, 2023. doi: 10.48550/ARXIV.2304.07193. URL https://doi.org/10.48550/arXiv.2304.07193.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=di52zR8xgf.

Prolific Team. Prolific. https://www.prolific.com, 2014. Accessed: 2025-11-24.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021. URL http://proceedings.mlr.press/v139/radford21a.html.

Andrew Rosenberg and Julia Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *Conference on Empirical Methods in Natural Language Processing*, 2007. URL https://api.semanticscholar.org/CorpusID:14153811.

Peter J. Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987. URL https://api.semanticscholar.org/CorpusID:189900.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 22500–22510. IEEE, 2023. doi: 10.1109/CVPR52729.2023.02155. URL https://doi.org/10.1109/CVPR52729.2023.02155.

Sogand Salehi, Mahdi Shafiei, Teresa Yeo, Roman Bachmann, and Amir Zamir. Viper: Visual personalization of generative models via individual preference learning. *CoRR*, abs/2407.17365, 2024. doi: 10.48550/ARXIV.2407.17365. URL https://doi.org/10.48550/arXiv.2407.17365.

Bernhard Schölkopf and Alexander Johannes Smola. *Learning with Kernels: support vector machines, regularization, optimization, and beyond*. Adaptive computation and machine learning series. MIT Press, 2002. ISBN 9780262194754. URL https://www.worldcat.org/oclc/48970254.

Gowthami Somepalli, Anubhav Gupta, Kamal Gupta, Shramay Palta, Micah Goldblum, Jonas Geiping, Abhinav Shrivastava, and Tom Goldstein. Measuring style similarity in diffusion models. *CoRR*, abs/2404.01292, 2024. doi: 10.48550/ARXIV.2404.01292. URL https://doi.org/10.48550/arXiv.2404.01292.

Kunpeng Song, Yizhe Zhu, Bingchen Liu, Qing Yan, Ahmed Elgammal, and Xiao Yang. Moma: Multimodal LLM adapter for fast personalized image generation. In Ales Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors, *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part XL*, volume 15098 of *Lecture Notes in Computer Science*, pages 117–132. Springer, 2024. doi: 10.1007/978-3-031-73661-2_7. URL https://doi.org/10.1007/978-3-031-73661-2_7.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Haofan Wang, Matteo Spinelli, Qixun Wang, Xu Bai, Zekui Qin, and Anthony Chen. Instantstyle: Free lunch towards style-preserving in text-to-image generation. *CoRR*, abs/2404.02733, 2024. doi: 10.48550/ARXIV.2404.02733. URL https://doi.org/10.48550/arXiv.2404.02733.

Haohan Wang, Songwei Ge, Zachary C. Lipton, and Eric P. Xing. Learning robust global representations by penalizing local predictive power. In *NeurIPS*, pages 10506–10518, 2019.

Shuyu Wang, Weiqi Li, Qian Wang, Shijie Zhao, and Jian Zhang. Mind-edit: Mllm insight-driven editing via language-vision projection. *ArXiv*, abs/2505.19149, 2025. URL https://api.semanticscholar.org/CorpusID:278905204.

Zijie J. Wang, Evan Montoya, David Munechika, Haoyang Yang, Benjamin Hoover, and Duen Horng Chau. DiffusionDB: A large-scale prompt gallery dataset for text-to-image generative models. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 893–911, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.51. URL https://aclanthology.org/2023.acl-long.51.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Shengming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, Hang Zhang, Hao Meng, Hu Wei, Jingyuan Ni, Kai Chen, Kuan Cao, Liang Peng, Lin Qu, Minggang Wu, Peng Wang, Shuting Yu, Tingkun Wen, Wensen Feng, Xiaoxiao Xu, Yi Wang, Yichang Zhang, Yongqiang Zhu, Yujia Wu, Yuxuan Cai, and Zenan Liu. Qwen-image technical report. *CoRR*, abs/2508.02324, 2025. doi: 10.48550/ARXIV.2508.02324. URL https://doi.org/10.48550/arXiv.2508.02324.

Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. *arXiv preprint arXiv:2410.13848*, 2024.

Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *CoRR*, abs/2308.06721, 2023. doi: 10.48550/ARXIV.2308.06721. URL https://doi.org/10.48550/arXiv.2308.06721.

Zhengyang Yu, Zhaoyuan Yang, and Jing Zhang. Dreamsteerer: Enhancing source image conditioned editability using personalized diffusion models. *ArXiv*, abs/2410.11208, 2024. URL https://api.semanticscholar.org/CorpusID:273351116.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3813–3824, 2023. URL https://api.semanticscholar.org/CorpusID:256827727.

Zikai Zhou, Shitong Shao, Lichen Bai, Shufei Zhang, Zhiqiang Xu, Bo Han, and Zeke Xie. Golden noise for diffusion models: A learning framework. In *International Conference on Computer Vision*, 2025.

Zhuofan Zong, Dongzhi Jiang, Bingqi Ma, Guanglu Song, Hao Shao, Dazhong Shen, Yu Liu, and Hongsheng Li. Easyref: Omni-generalized group image reference for diffusion models via multimodal LLM. *CoRR*, abs/2412.09618, 2024. doi: 10.48550/ARXIV.2412.09618. URL https://doi.org/10.48550/arXiv.2412.09618.

In this supplementary material, we provide additional details and analyses that complement the main paper. We first review relevant literature, including prior work on unified vision-language models and concept-conditioned generation, as well as studies on personalized preference extraction, multimodal conditioning, and personalization strategies for diffusion models such as fine-tuning, token-based methods, and adapters. We then describe the datasets used in our experiments, including the processing steps applied to Pick-a-Pic and the construction of the agent dataset. Next, we outline further experimental details, with a particular focus on the impact of the number of reference images. We also present additional experiments, covering a matched direct-history baseline, the generation-time effect of $\mathbf{e}_{\text{cons}}$, portability to a native multimodal generator, training-data scaling, evaluations of preference discrimination, the influence of training data quality, computational efficiency, and resource usage.

A Related Work

Research on Preference-Conditioned and multimodal generation spans several directions. One line of work develops unified vision-language models that close the gap between understanding and generation. Another thread focuses on extracting and injecting user-specific signals into pretrained generative backbones. A third direction explores lightweight or training-free personalization techniques that trade computational cost for deployment flexibility. We summarize these areas below and position our contributions accordingly.

Unified VLMs and concept-conditioned generation. Recent efforts have sought to build unified vision-language models [Wu et al., 2024, Ma et al., 2024, Chen et al., 2025] and semantic encoders that can support both recognition and generation. These models emphasize learning high-quality semantic features that can be reused across tasks, rather than training separate image priors or autoencoders for each domain. UniCTokens [An et al., 2025] extends this approach toward personalization by learning unified concept tokens that bridge personalized understanding and generation, improving both concept recognition and synthesis. Our work shares the motivation of reusing semantic signals from multimodal encoders, but differs by explicitly separating user-consistency cues from context-dependent intra-user preference representations and aligning the latter to the text embedding space of generative backbones.

Personalized preference extraction and multimodal conditioning. Another growing research direction focuses on extracting user preferences and transforming them into effective conditioning signals for generative models. ViPer [Salehi et al., 2024] presents a structured pipeline for preference-centric personalization by collecting user comments on a small set of images and converting them into attribute-level constraints for generation, underscoring the importance of structured preference capture. [Mo et al., 2025] introduces a contrastive learning framework based on MLLMs to effectively capture fine-grained user preferences. By modeling relative preference relationships among samples, this approach distinguishes between user "likes" and "dislikes." UniCTokens [An et al., 2025] and EasyRef [Zong et al., 2024] also contribute to this line of research by proposing mechanisms to integrate multi-image references and user-provided concepts into unified VLMs and diffusion models. EasyRef leverages multimodal large language models to aggregate multi-image references into robust and generalizable embeddings for diffusion conditioning, demonstrating how consistent signals can be distilled across multiple reference images for plug-and-play personalization. More recent approaches leverage multimodal large language models (MLLMs) [Li et al., 2024, Liu et al., 2024] to aggregate multiple image references into robust embeddings for diffusion-based personalization [Song et al., 2024, Dang et al., 2025]. Within this paradigm, IP-Adapter [Ye et al., 2023] offers a parameter-efficient mechanism for preference injection by exploiting CLIP's strong image–text alignment [Radford et al., 2021], outperforming alternative modules such as GLIGEN [Li et al., 2023] and ControlNet [Zhang et al., 2023] in both efficiency and fidelity. Despite these advances, aligning MLLM-derived embeddings with diffusion text encoders remains challenging. Recent studies attempt to bridge this gap with point-wise alignment objectives: MoMA [Song et al., 2024] uses mean squared error (MSE) loss to align MLLM embeddings with CLIP image embeddings, MIND-Edit [Wang et al., 2025] adopts cosine similarity, and CINEMA [Deng et al., 2025b] combines both to balance magnitude and directional alignment. While these objectives partially reduce the discrepancy, they often impose rigid constraints, limiting stability and hindering optimal transfer of preference information. Our approach similarly extracts multi-image preference cues using MLLMs, but we extract

two representations: a user-consistency embedding that captures stable cross-session traits, and an intra-user preference embedding that reflects context-dependent decisions. We further align intra-user preference embeddingwith the generator's text space to reduce representational mismatch.

Fine-tuning, token-based, and adapter personalization for diffusion models. Personalization in text-to-image diffusion models has also been studied through parameter-efficient fine-tuning, token-level injection, and adapter-based conditioning. Textual Inversion [Gal et al., 2023] and DreamBooth [Ruiz et al., 2023] represent early and influential methods that associate a subject or concept with a newly learned token or fine-tuned model weights, thereby enabling faithful synthesis of the subject in diverse contexts. While effective, these methods require per-concept optimization and careful strategies to avoid overfitting. IP-Adapter [Ye et al., 2023] introduces a modular adapter design that injects image prompts into pretrained diffusion models without retraining the backbone, enabling flexible image-conditioned personalization. More recent work such as Subject-Diffusion [Ma et al., 2023] and DreamSteerer [Yu et al., 2024] extends these ideas to improve subject-driven generation and editability, supporting identity preservation and flexible editing with minimal additional training. Compared with these approaches, our method combines the stability benefits of image-anchored conditioning with an explicit semantic alignment stage that maps MLLM-derived intra-user preference embeddings into the text-conditioning manifold used by generative backbones, thereby enabling more robust and user-aligned generation. Unlike subject-driven personalization methods, which focus on binding a generator to a specific instance, our work tackles user-preference personalization, where the goal is to capture stable and context-dependent aesthetic preferences from a few reference images and use them to guide generation.

B Dataset

Our training and evaluation rely on two complementary sources of data. The first is a curated subset of the Pick-a-Pic dataset [Kirstain et al., 2023], which provides real user annotations. The second is a large-scale synthetic corpus in which agents supply structured, high-consistency preference labels. We use this corpus as a scalable supervision signal and evaluate transfer on the Pick-a-Pic annotations above.

Pick-a-Pic Dataset Processing. We leverage the Pick-a-Pic dataset, which contains image–text pairs annotated by real users. To ensure data quality and minimize noise introduced by irrelevant prompts, we apply a multi-step filtering pipeline. First, we compute CLIP text embeddings and cluster them using DBSCAN [Ester et al., 1996], which naturally identifies semantic groups without requiring the number of clusters to be specified. Prompts identified as outliers are removed. Next, we prune samples that fail to meet general quality standards based on both PickScore and CLIPScore, using empirically chosen thresholds. Finally, approximately 10% of remaining samples that still exhibit unclear or inconsistent preferences are manually discarded. This procedure yields a cleaned subset that better reflects user preference signals while reducing annotation noise. Some examples are shown in Figure 5.

Pick-a-Pic evaluation protocol. The raw Pick-a-Pic test split contains many annotators with very sparse labels: a non-trivial fraction ($\sim$12%) provide only a single preference pair, whereas our reference-based assembly needs five like–dislike conditioning pairs plus one disjoint user-annotated evaluation image. After filtering (above), we assemble 200 preference sets for reporting in the main paper. Each set is built from annotations that provide five like–dislike pairs used as conditioning history and *one additional user-annotated image reserved solely for evaluation*, disjoint from the conditioning pairs so that leakage from history to metric is prevented. Metrics in Tab. 3 aggregate over these 200 sets under equal weighting. Auxiliary evaluations that require richer histories (for example, PREFDISC and proxy discrimination in App. D.10) use held-out judgments drawn from the same filtered Pick-a-Pic dataset under the same history–evaluation separation principle. The main generative table stays tied to the five-pair plus one-image construction.

Agent Dataset Construction. The synthetic preference dataset is constructed using preference-simulating agents to provide the volume necessary for robust training. Following the design principle of multi-dimensional visual attributes [Salehi et al., 2024], we employ Claude-3.5-Sonnet [Anthropic, 2024] as the backbone agent. Each agent is assigned a personalized aesthetic profile defined



Figure 5: Some examples of the processed Pick-a-Pic dataset.

across multiple visual dimensions (approx. 50 attributes on average) to ensure diversity and internal consistency. During generation, we sample random attributes and append them to human-written prompts [Wang et al., 2023, Lin et al., 2014], generating images via FLUX.1-dev [Black Forest Labs, 2024]. To encourage distinct simulated profiles while preserving within-profile consistency, any two agents' liked/disliked attribute sets have Jaccard distance at least 85%. Agents then annotate the results based on their profiles, yielding at least ten like–dislike pairs per agent. This pipeline results in roughly one million images (990,998) from over 50,000 simulated users. We partition the dataset by user identity (80/10/10 split). The synthetic preference data, including generated images, per-user liked/disliked histories, prompts, and attribute profiles, are publicly available at `https://huggingface.co/datasets/wenyii/PrefBench`.

Following [Salehi et al., 2024] and [Mo et al., 2025], we construct artificial users with distinct and consistent aesthetic preferences. To this end, we adopt the Claude-3.5-Sonnet agent [Anthropic, 2024] and employ the Flux.1-dev [Black Forest Labs, 2024] generator to synthesize images. We begin by defining a comprehensive taxonomy of aesthetic attributes that spans several key dimensions shown in Table 10, inspired by [Salehi et al., 2024], including artistic style, color palette, compositional strategy, skill level, level of detail, dominant hues, and medium. Each agent is assigned a personalized profile consisting of preferred and disliked attributes sampled from this taxonomy. This configuration yields controllable, fine-grained simulated preferences that act as *high-consistency supervision* for learning preference-sensitive representations. The input prompt to Claude-3.5-Sonnet is as follows:

```
You are a user with a distinct aesthetic preference profile.
Your profile is defined across multiple visual dimensions such as
artistic style, color palette, composition strategy, level of detail,
dominant hues, and medium.
From this taxonomy, a subset of attributes has been assigned to you.
These attributes are divided into two categories:
- Liked attributes: <attr>, <attr>, ..., <attr>
- Disliked attributes: <attr>, <attr>, ..., <attr>
Treat these preferences as stable and consistent, reflecting your unique
aesthetic identity.

You will be given a text prompt for image generation and a set of
generated images.
Your task is to evaluate each image according to the following procedure:

1. Examine the image carefully and compare it against your assigned
liked and disliked attributes.
2. Decide whether the image aligns with your liked attributes or
conflicts with your disliked attributes.
3. Assign a binary label to the image:
   - If the image matches your liked attributes, label it as "liked."
```

```
   - If the image matches your disliked attributes or deviates from your
taste, label it as "disliked."
4. Ensure consistency across all evaluations by relying strictly on the
given profile, without adding any personal judgment beyond the assigned
attributes.
5. For each image, output only "liked" or "disliked," without additional
commentary.

Here is the base text prompt for image generation: <prompt>
Here is the generated image: <image>
Here is the next image: <image>
```

Examples from the dataset are shown in Figure 6. In addition, $\mathcal{D}_{\text{intra}}$ and $\mathcal{D}_{\text{inter}}$ used in Sec. 2.2 contain 300 users, each associated with 11 labeled images.

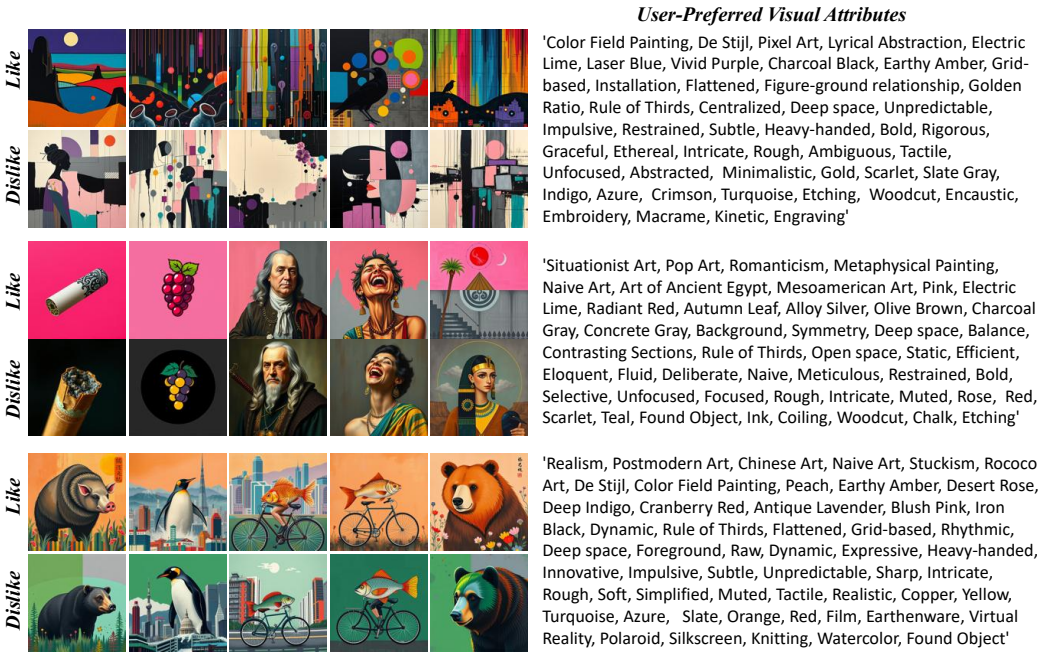


Figure 6: Some examples of the agent dataset.

The Processed ImageNet and ImageNet-sketch Dataset. To evaluate whether PrefGen generalizes to user preferences outside the training distribution, we construct an OOD preference discrimination task using real photographs and sketches from ImageNet [Deng et al., 2009] and ImageNet-Sketch [Wang et al., 2019], which are never seen during training. We use PickScore to annotate preference pairs and test whether our learned preference discriminator remains consistent with a high-quality external preference oracle on real data.

C Experimental Details

Our experiments are designed to accommodate varying numbers of preference history images, following the observations in [Salehi et al., 2024] regarding the relationship between reference count and preference discrimination. For each user, we randomly sample 6–14 annotated images as conditions, with all images resized to 512×512. For PREFDISC, we follow [Salehi et al., 2024] and use IDEFICS2-8B [Laurençon et al., 2024] as the multimodal backbone. Training runs for 3,000 steps on the training split with eight A100 (80GB) GPUs, an effective batch size of 16, a learning rate of 1×10^{-5}, and weight decay 1×10^{-2}. The distribution alignment module is a lightweight MLP (see Appendix C.3), trained for 400k steps on one A100 GPU with AdamW, learning rate 1×10^{-4}, and batch size 64. For PREFGEN, we pre-extract embeddings from the MLLM to reduce computational overhead during training. The weights of the MLLM are kept frozen, and only the

Table 10: A taxonomy of aesthetic attributes spanning style, medium, palette, composition, skill, detail, and hue.

Dimension	**Representative Attributes**
Artistic Medium	Painting (Oil, Fresco, Acrylic, Gouache), Sculpture (Stone, Metal, Ceramic, Glass), Textile Arts (Weaving, Quilting, Embroidery), Digital Forms (3D Modeling, AI Art, Virtual Reality), Printmaking (Etching, Woodcut, Silkscreen), Photography, Ceramics, Drawing
Color Palettes	Vibrant (Radiant Red, Vivid Purple), Pastel Hues (Lilac, Blush Pink, Powder Blue), Earthy Colors (Olive Green, Warm Brown, Clay Gray), Dark Colors (Charcoal Black, Deep Indigo, Burgundy), Oceanic Tones (Aquamarine, Navy, Turquoise), Neon Shades (Electric Lime, Atomic Green), Autumnal (Pumpkin Spice, Cranberry Red)
Skill Expression	Dynamic, Intuitive, Meticulous, Gestural, Bold, Spontaneous, Polished/Raw, Inventive, Lyrical, Controlled, Experimental, Graceful/Powerful, Free-flowing/Structured
Compositional Strategy	Rhythmic patterns, Centralized forms vs. decentralized layouts, Foreground and background contrast, Shallow depth vs. extended depth, Open areas vs. enclosed zones, Figure–ground interplay, Balanced or unbalanced arrangements, Installation format vs. pictorial surface, Fragmented structures vs. compressed ones, Static versus dynamic qualities, Positive and negative spatial tension, Contrasting divisions, Grid-structured organization, Exaggerated vs. flattened perception
Detail Treatment	Intricate, Simplified, Vivid, Subtle, Fine, Elaborate, Rough, Ethereal, Ambiguous, Textured, Realistic, Expressive, Minimalistic, Muted
Hues	Red, Scarlet, Magenta, Teal, Slate, Indigo, Gold, Copper, Yellow, Emerald, Azure, Turquoise, Burgundy, Cerulean, Crimson
Art Styles	Baroque Art, Graffiti, Impressionism, Ukiyo-e, Surrealism, Minimalism, Romanticism, Byzantine Art, Digital Art, Cubism, Mannerism, Conceptual Art, Renaissance Art, Futurism, Street Art, Symbolism, Art Nouveau, Queer Art, Contemporary Abstraction

adapter layers in the diffusion model, including the newly introduced decoupled cross-attention branch and its corresponding projectors, are trained. Stable Diffusion XL [Podell et al., 2024] is used as the base generator, initialized with IP-Adapter weights. Training uses AdamW with batch size 16, learning rate 1×10^{-4}, weight decay 1×10^{-2}, and 300k steps. Supervision is provided by user-liked images. During inference, images are generated using 30 denoising steps with a guidance scale of 0.6, following the hyperparameter settings in [Ye et al., 2023]. Results are averaged over five seeds $(1, 2, 3, 4, 5)$. For the IP-Adapter branch, we use one liked image per forward pass as the (e_{img}) anchor, randomly sampled from the user's liked images. During inference, we use the first available liked image as the anchor for consistency.

C.1 PREFGEN diffusion training tuples and IP-Adapter conditioning

Each training step samples a user u, a text prompt p, and the user's preference history $\mathcal{H}_u = \{(\mathbf{x}_i, y_i)\}$ with binary like/dislike labels. The MLLM consumes a subset of 6–14 annotated images drawn uniformly at random from $\mathcal{H}_u$ (same range as Sec. C.6), producing $\mathbf{e}_{\text{cons}}$ and $\mathbf{e}_{\text{intra}}$ after projection/alignment. For the IP-Adapter image branch, we sample *one* liked image per forward pass uniformly from u's liked set and encode it with CLIP to obtain $\mathbf{e}_{\text{img}}$. The denoising target is a user-liked image (standard score matching on latents of preferred outputs). It is sampled independently of the CLIP anchor each step, so the two are decorrelated whenever a user contributes multiple likes. At inference we take the first available liked image in the history for $\mathbf{e}_{\text{img}}$ for reproducibility.

For training the proposed PREFDISC model, we adopt a quantized low-rank adaptation (QLoRA) strategy. Specifically, the LoRA configuration is set with rank $r = 8$, scaling factor $\alpha = 8$, and dropout rate 0.1. We apply LoRA modules to all projection layers, including down-projection, gate-projection, up-projection, and the standard attention projections (k, q, v, o), across the text encoder, modality projection, and perceiver resampler. The LoRA weights are initialized using a Gaussian distribution to ensure stable training. Our framework is implemented on top of PyTorch, and the Maximum Mean Discrepancy (MMD) loss is incorporated following the open-source implementation provided by Lee [2020].

For comparison, we follow the original settings of baseline methods. ViPer relies on explicit preference signals from users, formulated as feedback statements such as "`I like this image.`" Bagel extracts user preferences from the provided reference images and generates outputs conditioned on these preferences, combined with the given prompt in the form: "`Extract the user's preferences based on the reference images and generate an image that conforms to the user's preferences. The content is: <prompt>.`" EasyRef is evaluated with its default prompting strategy without additional modification.

C.2 Pairwise Same-User Verification Probe

To evaluate the discriminative power of MLLM embeddings for user-consistency signals, we replace the previous 300-way user classification with a pairwise verification task. Given two disjoint preference histories $\mathcal{H}_a$ and $\mathcal{H}_b$, the goal is to predict whether they come from the same user. Positive pairs are created by splitting one user's history into two non-overlapping subsets, while negative pairs are formed by pairing histories from different users.

Each history is encoded into an embedding $\mathbf{e} = f_{\boldsymbol{\theta}}(\mathcal{H})$. For each pair, we construct a comparison feature

$$\mathbf{z}_{ab} = [\mathbf{e}_a; \mathbf{e}_b; |\mathbf{e}_a - \mathbf{e}_b|; \mathbf{e}_a \odot \mathbf{e}_b],$$

and train a single-layer linear probe with BCE-with-logits loss to predict same-user vs. different-user. For all runs, we fix the random seed to 42 and use a decision threshold of 0.5 on sigmoid scores at test time. The evaluation uses 300 users and reports three metrics: Accuracy (fraction of correct binary predictions), F1 (harmonic mean of precision and recall for the positive class), and AUC (area under the ROC curve, threshold-independent ranking quality). In addition to MLLM layer slices, we include a CLIP average-pooling baseline.

The results (Tab. 1 in the main paper) show that mid-upper layers (8–20) consistently outperform all alternatives, achieving the best Acc/F1/AUC. This indicates that these layers encode the most separable and stable user-level preference-consistency traits. By contrast, all-layer and top-layer settings can retain ranking information (relatively high AUC) but yield weak binary separation (lower F1), suggesting that user-consistency signals are diluted by less relevant semantics when layer selection is too broad or too shallow. These findings directly motivate our use of mid-upper-layer embeddings as $\mathbf{e}_{\text{cons}}$.

C.3 Implementation Details of Projection and Adapter Modules

Projection Model for $\mathbf{e}_{\text{cons}}$. The projection model is a lightweight projection network that maps MLLM embeddings into additional context token for the generator's cross-attention layers. The module consists of a single linear projection followed by reshaping and LayerNorm. Specifically,

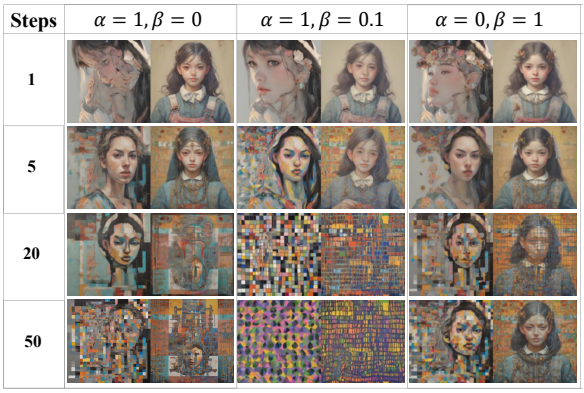
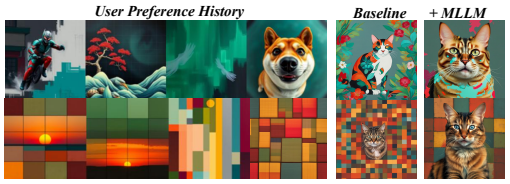
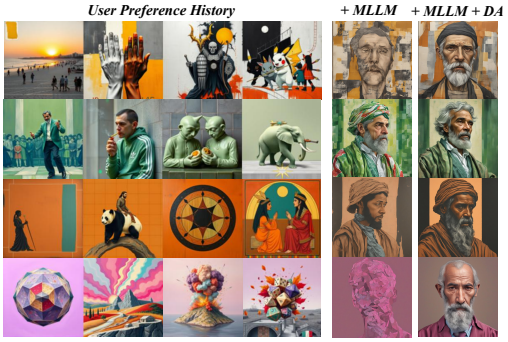


Figure 7: Alignment loss ablation under limited data. Aggressive point-wise shaping (β=0) collapses personalization signals, whereas $\mathcal{L}_{\text{MMD}}$ maintains stable generation quality consistent with Tab. 5.



(a) Effect of MLLM embeddings ("A cat").



(b) Effect of distribution alignment ("A man").

Figure 8: Visual ablations for embedding and alignment design. Left: removing MLLM embeddings loses fine-grained preference cues. Right: removing distribution alignment degrades stylistic coherence.

MLLM embeddings of dimension 4096 are projected into $k \times d_{\text{attn}}$ with $k = 1$ and $d_{\text{attn}} = 2048$. The resulting tensor is normalized before being used as extra context tokens.

MLP Adapter for $\mathbf{e}_{\text{intra}}$. The MLP Adapter is a six-layer feedforward network designed to compress high-dimensional embeddings into a compact latent representation. The input dimension of 4096 is first projected through a linear layer of size 4096, followed by a LayerNorm, a GELU activation, and a dropout layer with a rate of 0.1. After that, the network contains four hidden layers, each of size 2048, with the same LayerNorm–GELU–dropout structure. Finally, the output is mapped to a 2048-dimensional representation and reshaped to $[B, 1, 2048]$ for integration into the generator.

C.4 MMD estimator and training practice

We use the biased mini-batch MMD estimator with a Gaussian kernel, computed over batches of 64 mapped intra-user preference embeddings and their paired CLIP text embeddings. Following common practice, the kernel bandwidth is selected with the median heuristic within each batch and clamped to avoid numerical instability when a batch contains near-duplicate embeddings. We found the estimator stable in training. Fig. 7 and Tab. 5 compare MMD with MSE/cosine objectives under the same SDXL/IP-Adapter training setup.

C.5 Fusion Strategy Ablation

We compare three strategies for injecting the concatenated preference representation into the diffusion UNet. `Attn` uses cross-attention with $\mathbf{e}_{\text{img}}$ as the query and $\mathbf{e}_{\text{cons}}$ and $\hat{\mathbf{e}}_{\text{intra}}$ as keys and values. `Res-Attn` adds a residual path with a zero-initialized linear projection and layer normalization. `Concat` injects a single concatenated vector into the UNet. As shown in Fig. 9, direct concatenation outperforms learned attention variants.



Figure 9: Fusion strategy ablation. Direct concatenation (`Concat`) outperforms attention-based variants (`Attn`, `Res-Attn`).

C.6 Impact of the Number of Reference Images

Prior works [Salehi et al., 2024, Mo et al., 2025] have shown that incorporating as few as three liked and disliked images into the context already yields around 60% accuracy in preference prediction, and that performance continues to improve as the number of reference images increases. Building on this observation, we adopt a more flexible strategy during training. Instead of fixing the number of reference images, we randomly vary the input size to encourage robustness across different user scenarios. Specifically, for each user, we sample between 6 and 14 annotated images as conditions. This design choice allows the model to generalize better across varying history lengths and avoids overfitting to a fixed reference size.

In Figure 10, we compare ViPer and our PREFDISC across different sequence lengths. Our method remains stable, while ViPer proxy metric degrades when the sequence is either too short (insufficient preference information) or overly long (noisy or conflicting signals).

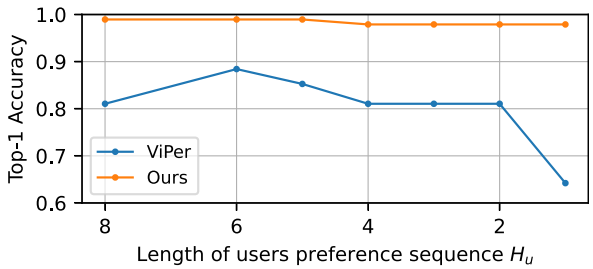


Figure 10: Effect of the length of the user preference sequence H_u on PREFBENCH. The horizontal axis shows the number of like-dislike pairs in the history. We compare ViPer and our method across different sequence lengths. Our method remains stable while ViPer degrades when the sequence becomes overly long.

C.7 Comparison to Other Methods.

Table 11 summarizes the number of reference images required and whether each method uses only positive examples or both positive and negative user preferences. To ensure a fair comparison, we use the same number of reference images for all methods that support multiple image inputs.

C.8 Overview of inference stage.

As illustrated in Figure 11, at inference time, a user's preference history consisting of liked and disliked images is processed by a frozen MLLM. The intermediate representations are then mapped by two separate adaptation heads: a linear projection layer yields the user-consistency embedding $\mathbf{e}_{\text{cons}}$, and a 6-layer MLP maps the intra-user preference embedding $\mathbf{e}_{\text{intra}}$ to its CLIP-aligned form $\hat{\mathbf{e}}_{\text{intra}}$. Implementation details of these projection adapters are provided in Appendix C.3. In parallel, a CLIP image encoder extracts a visual embedding $\mathbf{e}_{\text{img}}$ from a representative liked image. These embeddings are concatenated to form a unified user representation and injected into the frozen diffusion backbone via a decoupled cross-attention branch, together with the text embedding of the input prompt.

Table 11: Baseline input parity and evaluation protocol. We use each method in its closest supported setting and keep the text prompt fixed across methods. "SDXL@ours" indicates the same SDXL/IP-Adapter-style generation settings, and native large generators are evaluated with their released pipelines.

Method	Single Liked Ref.	Multiple Liked Refs.	Liked/Disliked Pairs	Generator
IP-Adapter	✓	✗	✗	SDXL@ours
InstantStyle	✓	✗	✗	SDXL@ours
StyleAligned	✓	✗	✗	SDXL@ours
Bagel	✓	✗	✗	native Bagel
Qwen-Image	✓	✗	✗	native Qwen-Image
Flux Kontext	✓	✗	✗	native Flux Kontext
EasyRef	✗	✓	✗	native EasyRef
ViPer	✗	✓	✓	released ViPer pipeline
PREFGEN	✓	✓	✓	SDXL@ours

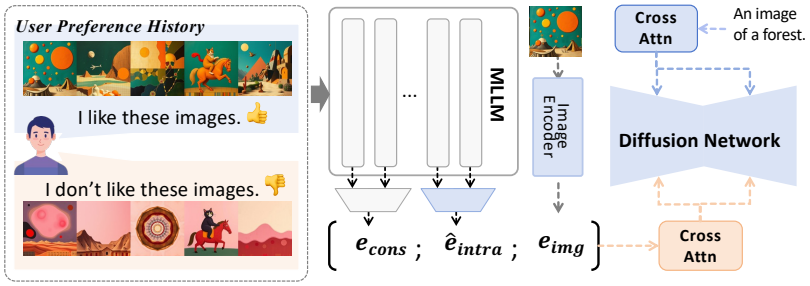


Figure 11: Overview of the inference-stage pipeline of PREFGEN.

D More Experiments

D.1 Pick-a-Pic filtering ablation

Because the main benchmark uses multiple filtering stages, we test sensitivity to annotation noise with an ordered dataset gradient while keeping the protocol fixed: five like–dislike conditioning pairs, one disjoint user-annotated evaluation image per set, and aggregation over 200 preference sets. Metrics are averaged with equal weight over 200 preference sets.

Table 12 follows the pipeline in Sec. B: protocol-only users (no DBSCAN/score/manual), then +DBSCAN, then +PickScore/CLIPScore thresholds, and finally +manual removal. We evaluate the same fixed PREFGEN checkpoint on every dataset.

As noise increases, performance degrades smoothly. Conversely, tighter filtering yields monotonic improvement, with gains already present before manual removal. Hence, PREFGEN remains competitive on noisier datasets.

D.2 Direct History Conditioning Baseline

DirectHist follows the setting "given these images the user likes, generate…": a fine-tuned generator receives the preference history directly as visual context, without learning an explicit preference representation. It uses the same SDXL backbone, IP-Adapter conditioning interface, generative training data, training schedule, diffusion objective, and conditioning budget as PREFGEN, but no preference-oriented MLLM, layer probing, or MMD alignment. The underlying conditioning interface is itself a strong choice: in Table 2, IP-Adapter outperforms Qwen-Image and Flux Kontext on CMMD, CLIP Img, and PREFDISC. The *Like-only* variant encodes every liked image with CLIP, mean-pools the features, and projects the result to the conditioning interface. The *Like+dislike* variant pools liked and disliked features separately and subtracts the disliked conditioning with scale s=1.0. As shown in Table 8a, both variants recover useful appearance cues, but PREFGEN performs best on all metrics. Directly aggregating history-image features is therefore a competitive control, but it does not recover the full gain of an explicitly learned preference representation.

Table 12: PREFGEN versus Pick-a-Pic filtering intensity.

Filtering applied to dataset	# pref. sets	FID (↓)	CLIP Img (↑)
Protocol only (no DBSCAN/score/manual)	200	185.40	0.799
+ DBSCAN prompt cleaning	200	182.06	0.805
+ PickScore/CLIPScore thresholds	200	179.74	0.809
Full pipeline (+ manual removal)	200	177.89	0.813

Table 13: Zeroing $\mathbf{e}_{\text{cons}}$ at inference in the converged checkpoint. "Improved" counts users (PREFBENCH) or preference sets (Pick-a-Pic) whose score decreases when $\mathbf{e}_{\text{cons}}$ is removed.

Test	With	Without	Δ	Paired t, p	Improved
PREFBENCH: output consistency	70.6	69.7	$+0.9$	$3.04, < 0.01$	80/136
Pick-a-Pic: held-out-liked alignment	57.7	57.3	$+0.4$	$2.28, < 0.05$	122/200

D.3 Generation-Time Effect of $\mathbf{e}_{\text{cons}}$

A finite history mixes stable user preferences with cues specific to the sampled images. $\mathbf{e}_{\text{intra}}$ captures liked-versus-disliked distinctions within the current history, whereas $\mathbf{e}_{\text{cons}}$ captures tendencies that remain stable across different subsets of the same user's history, consistent with its stronger user-level clustering in Table 21 (ARI 0.497 vs. 0.364; NMI 0.702 vs. 0.650). To test its effect on generation, we use the converged PREFGEN checkpoint and zero $\mathbf{e}_{\text{cons}}$ at inference without retraining. For each user or preference set, we sample $K{=}4$ history subsets, each containing two liked–disliked pairs, and fix the target prompt and noise across subsets and across the two conditions. On PREFBENCH, the score is the mean pairwise CSD among the four outputs, measuring cross-history output consistency. On Pick-a-Pic, the score is the mean CSD between the four outputs and the held-out liked image, which is disjoint from all conditioning images. Paired t-tests use one mean score per synthetic user ($n{=}136$) or Pick-a-Pic preference set ($n{=}200$).

The gain is consistent on both benchmarks, including on unseen real-user traces (Table 13). We therefore characterize $\mathbf{e}_{\text{cons}}$ as an auxiliary cross-history stabilizer that makes the generated preference less dependent on which specific history examples happen to be provided. The retrained controls in Table 8b provide complementary evidence on its utility.

D.4 Portability to a Native Multimodal Generator

To test whether the learned representation is tied to the SDXL/IP-Adapter interface, we transfer it to a frozen Bagel-7B [Deng et al., 2025a], a native multimodal generator that processes image and text tokens in a unified sequence. Bagel's transformer, ViT, and VAE are frozen, and only a lightweight projection is trained for 5k steps to map $[\hat{\mathbf{e}}_{\text{intra}}; \mathbf{e}_{\text{cons}}]$ into two preference tokens inserted into Bagel's native input sequence. This setting uses no IP-Adapter-style cross-attention branch. The reference-image conditions additionally provide a liked image through Bagel's native visual input.

Without a reference image, the preference tokens are the only preference-specific input and improve CSD from 48.8 to 53.1. When Bagel also receives a native visual reference, they further improve CSD from 50.8 to 54.2 (Table 14). Thus, the history representation can be transferred as native conditioning tokens and can either provide the preference signal alone or complement existing visual conditioning.

D.5 Training-Data Scaling

To study how much synthetic data is needed, we train matched 6k-step checkpoints on 10%, 30%, 50%, and 100% of the synthetic training corpus and evaluate them on the same leakage-free Pick-a-Pic benchmark.

All three metrics improve consistently with more training data (Table 15), so data scaling has not saturated within the tested range.

Table 14: Conditioning a frozen Bagel-7B with PREFGEN preference tokens. Gains are relative to the corresponding control without preference tokens.

Conditioning	Reference image	PREFGEN tokens	CSD ↑	Gain
Text only	✗	✗	48.8	–
Text + PREFGEN	✗	✓	53.1	+4.3
Text + reference image	✓	✗	50.8	–
Text + reference image + PREFGEN	✓	✓	**54.2**	+3.4

Table 15: Synthetic training-data scaling at a fixed 6k-step budget, evaluated on Pick-a-Pic. CLIP Img is scaled by 100.

Training data	CSD ↑	CLIP Img ↑	FID ↓
10%	48.9	72.8	188.2
30%	50.8	73.4	186.6
50%	51.8	73.9	186.1
100%	**52.5**	**74.3**	**185.4**

D.6 Explicit Attribute-Matching Baseline

To test whether our gains mainly come from *explicit textual preference descriptions* rather than learned preference embeddings, we introduce an explicit attribute-matching baseline that uses the same user history as other methods but removes all preference-embedding learning. Given a user history with liked/disliked images, we first convert the history into a natural-language preference profile, e.g., *"The user tends to prefer warm color palettes, painterly textures, centered composition, soft lighting, and dislikes dark cyberpunk, high contrast, cluttered backgrounds."* We then concatenate this profile with the original generation prompt and directly sample images with SDXL (without PREFDISC/PREFGEN encoders or adapter training). We evaluate a variant in this prompt-only setting. **LLM-Attr Prompt** uses Claude to summarize preference attributes from like/dislike history and appends that summary to the prompt for SDXL generation, reflecting a realistic no-training pipeline.

D.7 Impact of Training Data Quality

To investigate how training mixture affects the preference discriminator, we compare discrimination accuracy under different data configurations. ViPer constructs its proxy metric by combining agent-annotated data with the Pick-a-Pic dataset, which is not publicly available. In contrast, our agent corpus supplies curated, high-consistency pairwise labels at scale for controlled ablations.

Table 17 shows that PREFDISC trained solely on our agent data achieves the highest accuracy on PREFBENCH and the best listed result on the filtered Pick-a-Pic discrimination split built from the same dataset and filtering protocol described above. The underperformance of Pick-a-Pic-only and mixed training reflects a mismatch between our training objective, which targets coherent user-level preference structure from dense histories, and Pick-a-Pic's naturally collected pairwise annotations, which are sparse, prompt-dependent, and only sometimes assembled into long-term profiles. All PREFDISC and PREFGEN models reported elsewhere in the paper use **our agent data alone** as the training mixture, which empirically performs best overall.

D.8 OOD Preference Signals on Natural Images and Sketches.

To verify whether our learned preference representations generalize beyond the generated image domain, we conducted an evaluation on natural images and sketches. Specifically, we constructed a dataset of 1,920 images, randomly sampling 8 image pairs as user conditions to build 120 positive/negative test pairs across ImageNet and ImageNet-Sketch. Additional details and examples are provided in App. B. We report discrimination accuracy, AUC, and statistical significance metrics to assess performance. In Table 18, the results show significant alignment between PrefDisc's preference direction and PickScore's preference judgments, even when the input domain shifts from model-generated images to natural photographs and sketches. This result suggests that the learned representations retain useful preference signals outside the generated-image domain.

Table 16: Quantitative comparison on PREFBENCH. **Bold** indicates the best result.

Method	FID ↓	CMMD ↓	CLIP Img ↑	CSD ↑	PrefDisc ↑
LLM-Attr Prompt	160.42	0.27	72.41	55.08	78.84
PREFGEN (Ours)	**143.79**	**0.25**	**76.03**	**59.22**	**81.86**

Table 17: Comparison of different training data on preference discrimination accuracy (%).

Training Data	Model	Pick-a-Pic	PREFBENCH
Pick-a-Pic + ViPer's agent data	ViPer Proxy Metric	53.40%	81.62%
Pick-a-Pic	PrefDisc	51.89%	53.68%
Our agent data + Pick-a-Pic	PrefDisc	52.11%	85.29%
Our agent data	PrefDisc	**61.17%**	**98.67%**

D.9 More Qualitative Analysis Results

In Figure 12, we present the results of five users for five prompts that explicitly define the art style, medium, or color palette. Even though the results are still personalized and there is a clear preference pattern in each row, they consistently adhere to the input prompt's meaning.

D.10 Preference Discrimination Evaluation

To further assess the capability of PREFDISC in preference estimation, we evaluate it on two benchmark datasets and report top-1 accuracy as the primary metric. The evaluation protocol involves both automatic agents and human annotators to ensure a fair comparison.

We begin with human evaluation, where ten experts are asked to infer user preferences. For every test case, annotators are presented with several reference images explicitly divided into liked and disliked examples, followed by two candidate images. Their task is to identify which candidate aligns more closely with the inferred preferences.

We evaluate automatic agents under the same conditions. For each test instance, the Claude agent is provided with the same reference images and candidate pair. The agent receives a structured prompt that specifies which images are liked and which are disliked, and is instructed to infer visual preference patterns before selecting the preferred candidate. The input prompt is:

Table 18: Quantitative results of OOD preference prediction.

Metric	Value
Accuracy	0.7583 (91 / 120)
AUC	0.7552
Wilcoxon p-value	<1e-6 (stat = 6038.00)
Paired t-test p-value	<1e-6 (t = 7.53)
Mean diff (pos–neg)	0.3048 ± 0.4418
Median diff	0.2810

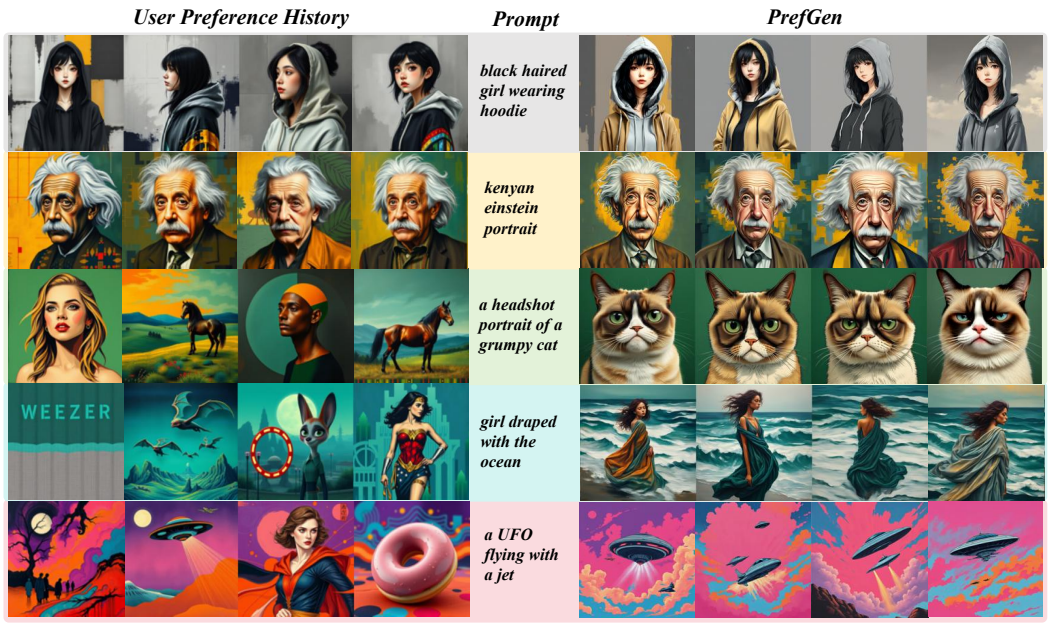


Figure 12: The images generated by PREFGEN. Each example shows user history preference on the left, the text prompt in the middle, and results from PREFGEN on the right. Our approach adapts to user-specific aesthetic signals, generating outputs that more faithfully reflect the preference history.

```
You are given a set of reference images that represent a user's
preferences.

Images in [<image>,...,<image>] are liked.
Images in [<image>,...,<image>] are disliked.

Your task is to:

1. Analyze the reference images to infer the user's visual preference
patterns.
2. Compare the two candidate images: <image> and <image>.
3. Select the single candidate (index 0 or 1) that best matches the
inferred preferences.

Output format:
Return only the index (0 or 1) of the chosen candidate, followed by a
one-sentence explanation of the decision.
Do not output anything else.
```

Unlike ViPer Proxy Model, which is limited by a fixed number of reference examples, PREFDISC is designed to flexibly handle varying numbers of reference images. This adaptability makes it more practical for real-world personalization, where the amount of available preference data often varies across users. Furthermore, PREFDISC is trained on a broader and higher-quality dataset generated

Table 19: Comparison of different methods on preference discrimination accuracy (%).

Model	Pick-a-Pic	PREFBENCH
ViPer Proxy Metric	53.40%	81.62%
Claude-3.5-Sonnet	48.54%	90.44%
Human	62.14%	98.67%
PrefDisc	61.17%	98.67%

Table 20: GPU memory usage and inference latency (per image).

Method	**Base VRAM (allocated)**	**Latency (s)**
SDXL (base)	6.57 GB	1.85
IP-Adapter	10.75 GB	1.89
EasyRef	11.68 GB	3.50
PREFGEN	26.50 GB	2.30
Bagel	27.30 GB	18.71
ViPer (w/ Refiner)	42.24 GB	5.66

by Flux.1-dev [Black Forest Labs, 2024], providing stronger coverage of visual preference patterns compared to the smaller, agent-limited datasets used by ViPer.

Table 19 presents the comparative results. On the processed Pick-a-Pic dataset, PREFDISC achieves the highest accuracy among automated preference proxies and is the closest to human experts, outperforming ViPer's proxy metric and Claude-3.5-Sonnet, which shows meaningful transfer from synthetic-agent training to real pairwise preferences. On PREFBENCH, PREFDISC reaches 98.67%, matching the reported human score.

D.11 Computational Efficiency and Resource Usage

We compare the computational requirements of our framework with existing methods along three axes: dataset scale, GPU usage, and training time. EasyRef is trained on over 130M high-quality images together with 4M text–image pairs, which requires nearly five days of computation on 64 H800 GPUs. Bagel operates on an even larger corpus of 2,665M examples, although the paper does not disclose the exact GPU configuration or training duration. Both methods therefore, rely on massive datasets and substantial GPU resources. ViPer is trained on the Pick-a-Pic dataset and adopts a setup by constructing preference data from 5,000 agents, each annotated with 50 positive and negative attributes and corresponding generated images. Its proxy metric is trained in less than five hours on a single A100-80GB GPU, but the paper does not report the GPU usage or training time for its generative model.

Our method offers a balanced trade-off between efficiency and performance. PREFDISC is trained on approximately 1M preference-labeled examples and converges within one day using 8 A100-80GB GPUs, while PREFGEN requires about five days on the same hardware. This resource footprint is far smaller than that of EasyRef and Bagel, while the resulting models achieve stronger preference-conditioned generation results. Compared to ViPer, our approach also delivers stronger performance.

Per-image inference latency and GPU memory. We report wall-clock inference cost and peak allocated VRAM under our evaluation setup in Tab. 20. PREFGEN achieves competitive latency (2.30s) with a single MLLM forward pass, comparable to lightweight adapters (e.g., IP-Adapter 1.89s) and much faster than inversion-heavy or autoregressive-heavy baselines such as ViPer (5.66s) and StyleAligned (27.45s in our setting). Memory use (26.50 GB) exceeds the SDXL backbone due to the MLLM, but remains comparable to Bagel (27.30 GB) and well below ViPer with refiner (42.24 GB).

D.12 Comparing $\mathbf{e}_{\text{intra}}$ and $\mathbf{e}_{\text{cons}}$ Representations

To better understand the functional roles of different embedding layers, we compare $\mathbf{e}_{\text{cons}}$, extracted from middle-to-upper layers, with $\mathbf{e}_{\text{intra}}$, derived from the topmost layers. The former is expected to retain enduring user-specific traits, while the latter refines these traits into more explicit semantic



Figure 13: Controlled embedding swap between two users. Histories from users (a) and (b) yield $\mathbf{e}_{\text{intra}}$, $\mathbf{e}_{\text{img}}$, and $\mathbf{e}_{\text{cons}}$. Panel (c) shows generations when one component from (a) is replaced by the corresponding component from (b), holding the others fixed.

Table 21: Clustering performance comparison between $\mathbf{e}_{\text{intra}}$ and $\mathbf{e}_{\text{cons}}$. Higher values indicate better clustering quality.

Metric	$\mathbf{e}_{\text{intra}}$	$\mathbf{e}_{\text{cons}}$
ARI (↑)	0.3641	0.497
NMI (↑)	0.6497	0.7024
AMI (↑)	0.5332	0.6017
Homogeneity (↑)	0.6538	0.7082
Completeness (↑)	0.6456	0.6968
V-measure (↑)	0.6497	0.7024

judgments. To assess their clustering properties, we apply K-means [MacQueen, 1967] over embeddings from 17 categories (users) and evaluate the resulting clusters against ground-truth labels using several standard metrics.

As shown in Table 21, $\mathbf{e}_{\text{cons}}$ consistently outperforms $\mathbf{e}_{\text{intra}}$ across NMI, AMI [Nguyen et al., 2010], V-measure, Homogeneity, and Completeness [Rosenberg and Hirschberg, 2007]. These results indicate that $\mathbf{e}_{\text{cons}}$ embeddings better capture global category structures, yielding clusters that are both more homogeneous within each class and more separated across different classes. This analysis suggests that $\mathbf{e}_{\text{cons}}$ serves as a more faithful representation of underlying categorical information, making it a stronger candidate for downstream applications that require robust preference modeling.

t-SNE visualization of MLLM feature representations. To examine whether the learned embeddings $\mathbf{e}_{\text{cons}}$ and $\mathbf{e}_{\text{intra}}$ capture user-specific preference structures, we project them into a two-dimensional space using t-SNE. We extract embeddings from 170 preference histories in the test set, covering 17 users. Each point in Fig. 14 and Fig. 15 corresponds to a user's preference history embedding derived from multiple labeled images, and points are color-coded by user identity. The visualization shows that embeddings of preference histories from the same user form coherent clusters, while embeddings from different users remain well separated. This demonstrates that the model encodes consistent signals of individual preference rather than collapsing to generic representations. We additionally report inter- versus intra-user distance ratios and permutation tests in Sec. D.13.

D.13 Inter- versus intra-user separability of embeddings

We complement Figs. 14 and 15 with a quantitative analysis on the *same* set of user-labeled preference histories underlying Table 21. For each embedding type, we treat each user as a class, compute cluster-level statistics, and report (i) the ratio $R = d_{\text{inter}}/d_{\text{intra}}$, where d_{intra} is the mean within-user pairwise ℓ_2 distance and d_{inter} is the mean between-user distance (averaged over user pairs), (ii) the mean silhouette coefficient [Rousseeuw, 1987], and (iii) a permutation test with $4{,}999$ random shuffles of user labels: we recompute R under each shuffle and summarize the standardized score z_{perm} of the observed statistic relative to the null distribution together with the two-sided empirical p-value p_{perm}.

Table 22 supports three conclusions. **(i)** Compared with a strong frozen visual encoder (DINOv2, $R{=}0.81{<}1$), both $\mathbf{e}_{\text{intra}}$ and $\mathbf{e}_{\text{cons}}$ achieve $R{>}1$, indicating that average between-user dispersion

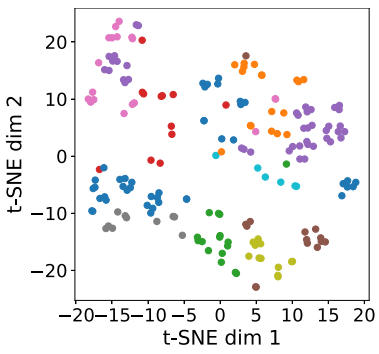


Figure 14: t-SNE of $\mathbf{e}_{\text{cons}}$.

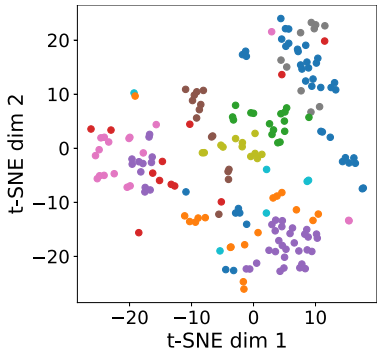


Figure 15: t-SNE of $\mathbf{e}_{\text{intra}}$.

Table 22: Inter-/intra-user distance ratio R, mean silhouette, and permutation statistics (4,999 label shuffles). Higher R, silhouette, and z_{perm} indicate stronger user-specific structure, whereas lower p_{perm} indicates departure from label-randomized chance.

Embedding	R (↑)	Silhouette (↑)	z_{perm} (↑)	p_{perm} (↓)
DINOv2 [Oquab et al., 2023]	0.81	0.066	27.6	< 0.001
CSD [Somepalli et al., 2024]	1.12	0.107	38.0	< 0.001
$\mathbf{e}_{\text{intra}}$	1.17	0.131	36.1	< 0.001
$\mathbf{e}_{\text{cons}}$	**1.37**	**0.193**	**43.6**	< 0.001

exceeds average within-user dispersion under Euclidean geometry in embedding space. **(ii)** The effect is statistically extreme under label randomization: every row attains $p_{\text{perm}}<0.001$ with large z_{perm}, so the observed structure is unlikely to arise from arbitrary reassignment of histories to users. **(iii)** $\mathbf{e}_{\text{cons}}$ consistently yields stronger separation than $\mathbf{e}_{\text{intra}}$ (1.37 vs. 1.17 on R, and higher silhouette), consistent with the role of mid-upper layers as a user-consistency readout in Sec. 2.2.

We also include CSD [Somepalli et al., 2024], a CLIP-based representation trained for artistic similarity. CSD improves over DINOv2 (R=1.12) but remains below our MLLM-derived embeddings, suggesting that generic style similarity captures overlapping cues yet leaves headroom for user-specific preference geometry.

D.14 Time-varying User Preferences.

Figure 16 illustrates a qualitative stress test for time-varying user preferences. In practice, long-term preference histories can be viewed as a sequence of short-term, relatively stable windows, within which users tend to exhibit consistent visual and aesthetic tendencies. In Figure 16(a), we visualize the temporal evolution of a user's visual style preferences, where the dominant color palettes and overall tonal characteristics gradually change over time. Figure 16(b) presents the images generated by PREFGEN when conditioning on different temporal windows of the same user's preference history, while keeping the base textual prompt fixed as "a beautiful landscape". The results suggest that the model can respond to temporal preference shifts, producing images whose color tones, atmosphere, and overall styles align with the selected preference window.

D.15 Semantic Preferences

To evaluate whether our method captures semantic preferences rather than exhibiting stylistic bias, we design an experiment using both real and synthetic preference histories in Figure 17. Specifically, we construct user preference histories from (a–b) ImageNet images [Deng et al., 2009], where preference pairs are formed across different fine-grained dog categories, and (c) synthetic preference images generated by FLUX [Black Forest Labs, 2024]. During generation, we fix the text prompt to the same input for all cases, "a photo of a dog." For the synthetic histories in (c), we additionally normalize background and color distributions.

Across both real ImageNet-based and FLUX-generated preference histories, PrefGen tends to generate dogs aligned with users' preferred fine-grained semantic categories. These results suggest that

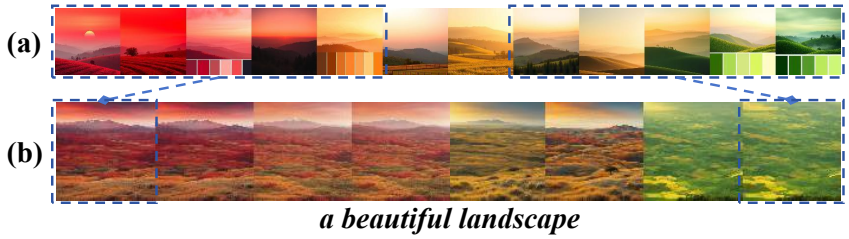


Figure 16: Time-varying user preferences and corresponding generated results. (a) Illustrates the evolution of a user's preferred visual styles over time, shown by changes in color palettes and tone. (b) Images generated by PREFGEN conditioned on different temporal preference windows. The base prompt is "a beautiful landscape."

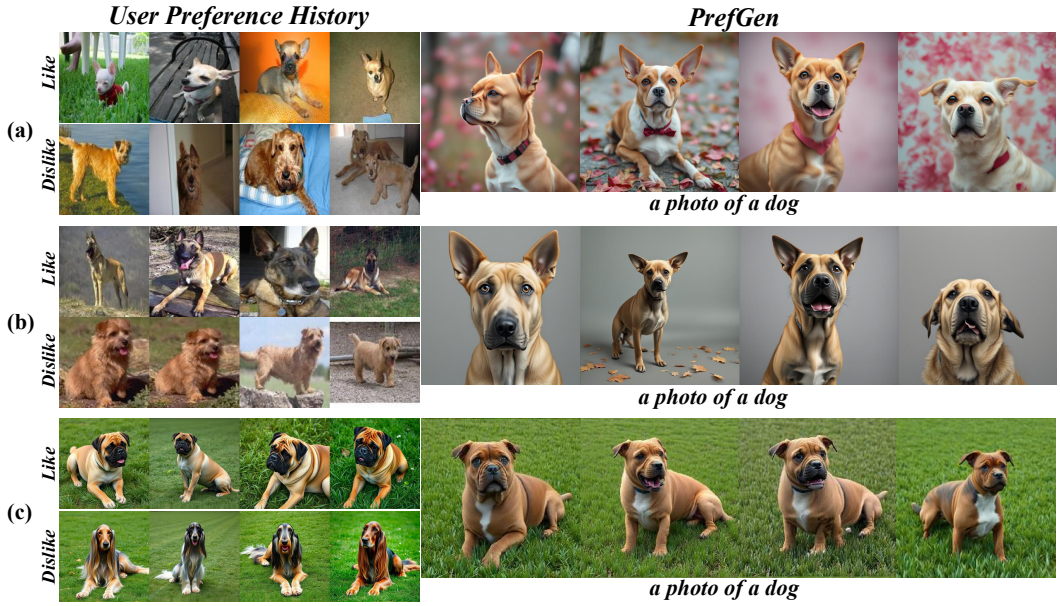


Figure 17: Semantic preference modeling with real and synthetic user preference histories. (a–b) User preference histories constructed from ILSVRC2012 ImageNet [Deng et al., 2009], where positive and negative examples are sampled from different fine-grained categories. (c) Synthetic preference histories generated by Flux.1-dev [Black Forest Labs, 2024]. PREFGEN successfully generates dog images that match users' fine-grained semantic preferences (e.g., breed type and facial morphology), demonstrating its ability to model semantic preferences beyond superficial style or color patterns.

PrefGen can capture semantic-level user preferences, rather than relying only on superficial color or stylistic cues.

Diversity vs. alignment. Personalization can reduce cross-seed spread when the generator commits to a tighter aesthetic region. Following the protocol in our evaluation code, for each prompt we generate $K{=}5$ seeds, extract CLIP ViT-L/14 image features before ℓ_2 normalization, and report the mean pairwise Euclidean distance averaged over prompts. Under this measure, IP-Adapter attains the highest spread (10.47), PREFGEN is lower (8.98), and Flux Kontext is similar (8.48), indicating that our drop relative to IP-Adapter co-occurs with stronger user alignment (Tab. 2) rather than a collapse absent in other strong baselines.

D.16 User Study

We conduct two complementary human evaluations. First, we recruit 15 expert evaluators with prior experience in visual design or aesthetic assessment. In each trial, experts are shown a pair of images generated by two different methods using the same text prompt and the same user history. They are asked to select the image that better reflects the preferences implied by the reference set. Evaluators are instructed to consider both prompt fidelity and preference alignment, including whether the

generated image matches the semantic content of the prompt and whether its color palette, artistic style, and overall visual presentation are consistent with the aesthetic signals in the liked examples.

To complement the expert evaluation with a broader set of non-expert judgments, we also conduct a larger pairwise study on Prolific [Prolific Team, 2014], involving 50 participants and 900 pairwise comparisons. This study is intended to assess whether the observed preference-alignment trends remain visible under more general crowd evaluation. Compared with expert annotators, crowd participants may rely more on immediately perceptible visual cues, such as color, style, and overall appearance. The Prolific study is implemented as a self-contained web interface. Participation is voluntary, and participants provide informed consent before beginning the task. Crowd participants are compensated at a fair hourly rate consistent with Prolific's payment guidance. Method identities remain hidden throughout the evaluation. The expert panel is conducted separately from the Prolific study.

In both studies, method identities are hidden from evaluators, and the left/right ordering of the two images is randomized independently for each trial. We report expert and crowd results separately, since the expert panel emphasizes more specialized design judgment, while the Prolific study provides broader subjective feedback from general viewers.

D.17 User-in-the-loop Personalized Pairwise Study

We further conduct a user-in-the-loop personalized pairwise study to evaluate whether PREFGEN can adapt to preferences provided directly by the same participant. The study involves 30 participants, 10 evaluation prompts, and four baseline comparisons, yielding up to 1,200 pairwise judgments in total.

Each participant completes a two-stage workflow through a web interface. In Stage A, participants construct their own preference history. They are shown 50 seed images sampled to cover a diverse range of visual styles, in randomized order and without replacement. Each participant selects exactly five images they like and exactly five images they dislike, with liked and disliked sets required to be disjoint.

In Stage B, participants evaluate generated images conditioned on the preference history they created in Stage A. For each evaluation prompt and baseline comparison, the participant is shown one image generated by PREFGEN and one image generated by a baseline method, both conditioned on the same text prompt and the same liked/disliked history. The participant is asked to choose the image that feels more aligned with the preferences expressed in their own selected history.

All comparisons are blinded: method names are not shown to participants, and the left/right placement of PREFGEN and the baseline output is randomized independently for each trial.

D.18 Study Interface

At the start of the session, participants read a brief overview of the task and the two-stage workflow before providing consent (Fig. 18, top). They are then shown 50 diverse seed images and asked to select exactly five liked images and exactly five disliked images (Fig. 18, bottom). The interface enforces disjoint liked and disliked selections, and participants cannot proceed until both quotas are satisfied.

Clicking an image tile cycles its label through *neutral* $\rightarrow$ *Like* $\rightarrow$ *Dislike* $\rightarrow$ *neutral*. To reduce accidental over-selection, once the quota of five liked images is reached, additional neutral tiles are assigned to *Dislike* on first click. Participants may revise their selections before proceeding to the evaluation stage.

After curating their preference history, participants completed 40 pairwise evaluation trials. In each trial, they viewed their curated liked/disliked reference history, the text prompt, and two anonymized candidate generations labeled Left and Right (Fig. 19). They were asked: *"Which image better reflects the aesthetic preferences shown in your history, while also matching the text prompt? Left or Right?"* Method identities were hidden throughout, and left/right assignment was randomized independently per trial.

D.19 Licenses of Core Assets

Core components are credited through their cited publications and upstream releases. SDXL-base-1.0 [Podell et al., 2024] is distributed under the CreativeML Open RAIL++-M license. IDEFICS2-8B [Laurençon et al., 2024] and Bagel-7B-MoT [Deng et al., 2025a] are distributed under Apache 2.0, and the IP-Adapter [Ye et al., 2023] code also uses Apache 2.0. The CLIP [Radford et al., 2021] code uses the MIT license. FLUX.1-dev [Black Forest Labs, 2024] weights are distributed under the FLUX.1 [dev] Non-Commercial License. Claude-3.5-Sonnet [Anthropic, 2024] is accessed as a proprietary service under Anthropic's applicable terms. The prompts used to build the synthetic dataset come from DiffusionDB [Wang et al., 2023], released under CC0 1.0, and COCO captions [Lin et al., 2014], released under CC BY 4.0. Pick-a-Pic [Kirstain et al., 2023], ImageNet [Deng et al., 2009], and ImageNet-Sketch [Wang et al., 2019] are used for research evaluation under the terms of their original releases.

D.20 Broader impacts.

Preference-conditioned generation can make creative tools more adaptive to individual users, reducing repeated prompt engineering and helping users express visual preferences that are difficult to verbalize. However, models that infer and use user preference histories also raise potential risks. Preference histories may contain sensitive information about users' tastes, identities, or intentions, and personalized generation systems could be misused for manipulative content, biased personalization, or targeted synthetic media generation. Future deployments should therefore emphasize user consent, data minimization, transparent control over stored preference histories, and safeguards against harmful or deceptive generation.

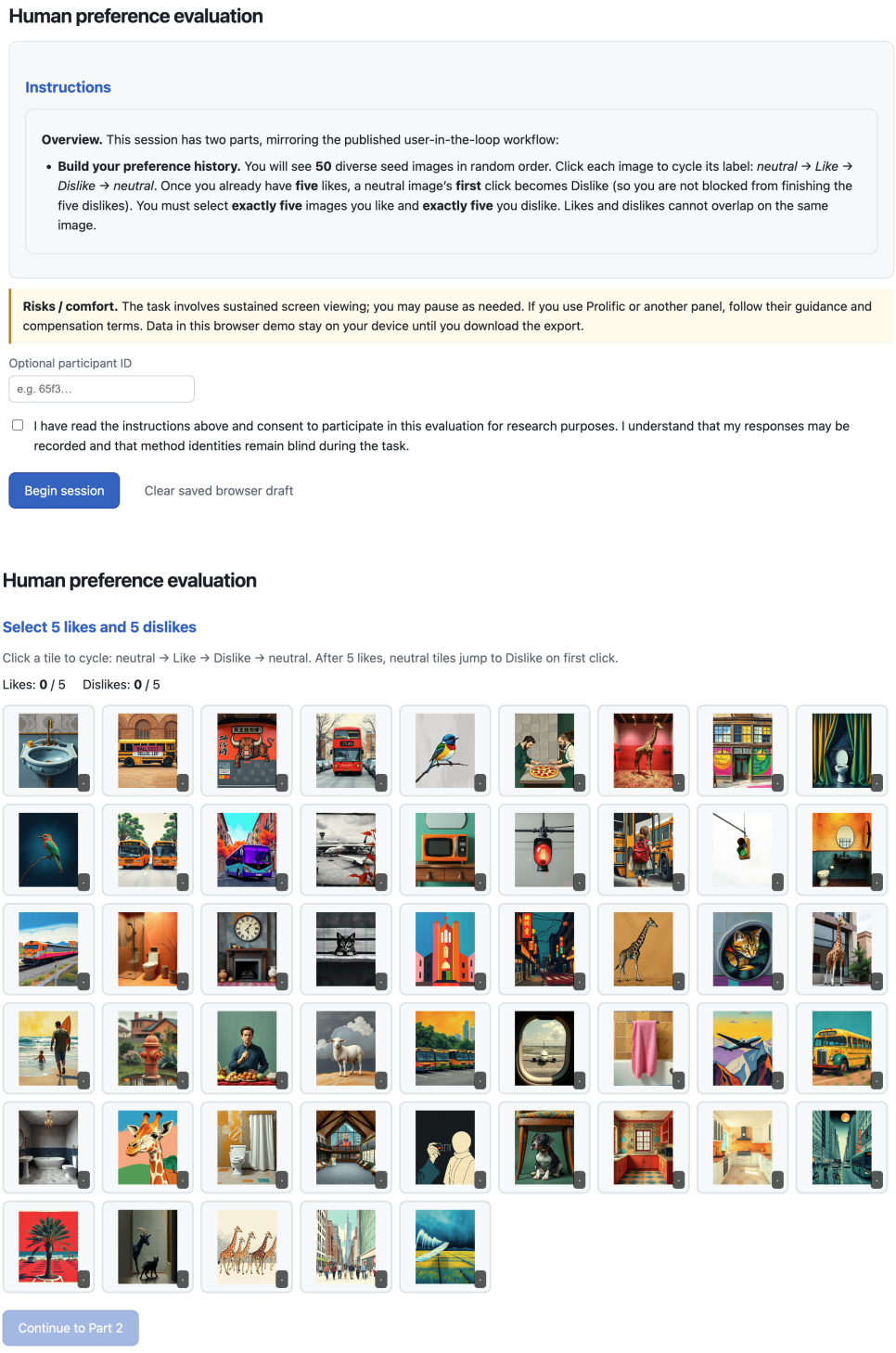


Figure 18: **Top:** Consent and instruction page shown at the start of the session, summarizing the workflow and the risks/comfort notice. Participants entered an optional Prolific ID and checked the consent box before beginning. **Bottom:** The preference-history curation screen. Participants viewed 50 seed images in random order and selected exactly five "Like" and five "Dislike" images (disjoint). Tile labels cycle *neutral* → *Like* → *Dislike* → *neutral*; the counter at the top tracks progress. The "Continue" button is disabled until both quotas are met.

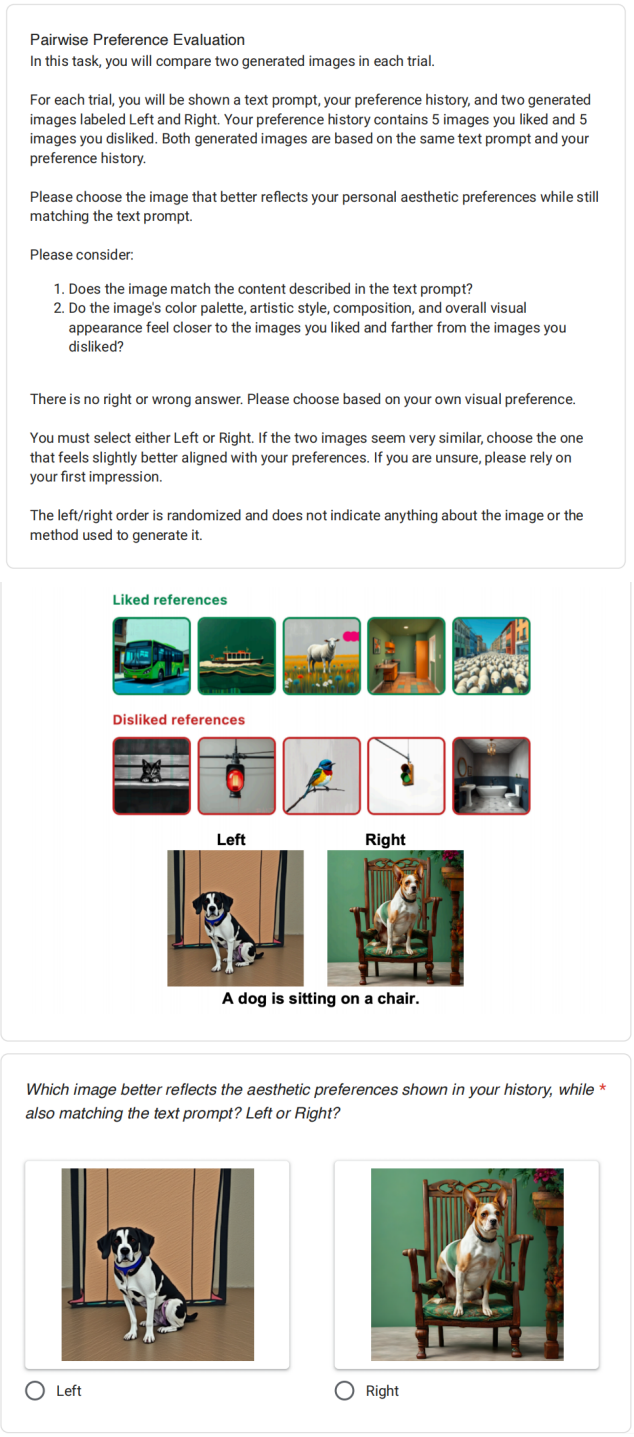


Figure 19: **Top:** Per-trial task instructions specifying the two evaluation criteria: (1) semantic match with the text prompt and (2) aesthetic alignment with the participant's liked references relative to disliked ones. **Bottom:** An example trial showing the reference history (Dislike row and Like row), the text prompt, and two anonymized candidate images (Left and Right) with radio-button selectors. Method identities are hidden and left/right assignment is randomized independently per trial.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state our contributions (MLLM preference modeling, probing, MMD alignment, SDXL conditioning, PREFBENCH and Pick-a-Pic evaluation, human studies), and Sections 2–3 together with the supplement match those claims; we also qualify transfer and synthetic-vs-real gaps where appropriate (e.g., Sec. 3 and App. B).

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper explicitly discusses the main limitations of the work.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper does not present numbered theorems or formal proofs; we use standard definitions (e.g., MMD) as methodological background and explicitly avoid overstating finite-batch MMD as a distributional certificate (Sec. 2).

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We describe datasets, splits, metrics, baselines, training hyperparameters, and implementation details in Sec. 3 and the supplement (e.g., App. B, training and resource sections); third-party base models and benchmarks are cited.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The paper provides detailed experimental and implementation information in the supplement to support reproducibility. The synthetic preference dataset (generated images, per-user liked/disliked histories, prompts, and attribute profiles) is publicly available at `https://huggingface.co/datasets/wenyii/PrefBench` and linked from the project page (`https://prefgen.github.io`); see App. B.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Sec. 3 and the supplement specify data splits, metrics, optimizer settings, LoRA configuration, MMD training, and SDXL/IP-Adapter training details needed to interpret the results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We report paired t-tests for the generation-time effect of $\mathbf{e}_{\text{cons}}$, using synthetic users and Pick-a-Pic preference sets as units (App. D.3), permutation tests for embedding separability (App. D.13), and significance tests for OOD preference prediction (App. D.8). Main generation tables and pooled human win rates are reported as point estimates averaged over seeds, preference sets, or judgments.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The supplement reports GPU types (e.g., A100 80GB), training durations/steps, and inference memory/latency comparisons (App. D.11, Tab. 20).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We reviewed the NeurIPS Code of Ethics; the paper cites data sources and models appropriately, and human evaluations follow anonymized protocols described in App. D.16.

   Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper includes a broader-impact discussion.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We do not release a scraped dataset or a new foundation model. The released synthetic preference dataset consists of FLUX.1-dev generations from public DiffusionDB and COCO prompts with simulated-agent annotations, and contains no real-user personal data; we build on publicly documented diffusion and MLLM components under their upstream licenses, and any future model release would follow those upstream policies.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [No]

    Justification: We cite datasets and model sources (e.g., Pick-a-Pic, SDXL, IP-Adapter, backbone MLLMs) and list the licenses of the core model and code components in App. D.19; license documentation for every evaluation asset is not exhaustively listed.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The synthetic preference dataset is released at `https://huggingface.co/datasets/wenyii/PrefBench` with a dataset card documenting the file layout, per-user record fields, and loading formats; its construction and the PREFBENCH protocol are described in Sec. 3 and App. B.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: We include the full text of instructions given to participants and screenshots.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes].

    Justification: We did not collect names, facial images, biometric data, medical records, demographic attributes, or other sensitive personal information. Platform worker identifiers were used only for compensation and quality control, and responses were analyzed in anonymized aggregate form.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: Core methods use an MLLM backbone (IDEFICS2-8B with QLoRA for PREFDISC; Sec. 2 and App. C) and Claude-based simulated users for large-scale preference labeling in dataset construction (Sec. 3); usage is described in the main text and supplement.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.