# DIVERSITY PREFERENCE RANKER: DIVERSE TEXT-TO-IMAGE GENERATION WITH UNIFIED SCORING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

We explore diverse image generation from accelerated text-to-image models, whose outputs can remain similar across different random seeds. Existing selection methods typically combine independently trained diversity and quality scorers that encode each candidate in isolation. We argue that diversity and preference scoring can benefit from a shared visual representation, allowing both supervision signals to shape the features used for scoring. We further emphasize that capturing a candidate's distinctiveness requires considering the other images in the pool. These considerations motivate Diversity Preference Ranker (DPR), a lightweight ranker that replaces separate diversity and preference scorers with a shared representation of the candidate pool. Self-attention encodes all candidates together into a shared field that provides pool-conditioned pairwise diversity distances and per-candidate preference scores. Spatial feature distillation and human preference comparisons supervise the shared representation. Supervising diversity and preference individually does not directly teach the ranker how to combine them when selecting candidates. We therefore introduce listwise selection distillation, which trains the combined scores to imitate a teacher's next-candidate choices given the selected subset. Extensive experiments across four accelerated generators show that the same frozen ranker achieves comparable diversity and preference to the evaluated selection baseline without generator-specific retraining. As a further application, we demonstrate that the same frozen ranker also supports noise optimization.

## 1 INTRODUCTION

Text-to-image (T2I) diffusion models produce high-quality images from natural language descriptions (Rombach et al., 2022; Esser et al., 2024), underpinning various applications such as personalization (Ruiz et al., 2023) and data augmentation (Azizi et al., 2023). Recent distillation approaches make these models dramatically faster, reducing sampling to a handful of steps (Yin et al., 2024; Sauer et al., 2024). Since generating many candidates per prompt is now cheap, the diversity among those instances directly determines their practical value: a user who requests several images for a single prompt expects distinct options, not near-duplicates. However, images generated from different seeds often retain similar content and layout (Corso et al., 2024; Sadat et al., 2024), an effect that is particularly significant in these accelerated models (Gandikota & Bau, 2026). This stems in part from distillation itself: students trained to imitate a teacher under strong guidance (e.g., with a high CFG weight (Ho & Salimans, 2022)) inherit its bias toward high-density, dominant outputs (Meng et al., 2023; Li et al., 2026).

Many approaches have emerged to improve diversity in diffusion and flow-based models (Corso et al., 2024; Um & Ye, 2024; Kim et al., 2025) (see Appendix A.1). The key challenge lies in improving the distinctiveness of outputs while maintaining their realistic quality. To achieve this, a recent paradigm incorporates two dedicated scoring models: one for quality (e.g., HPSv2 (Wu et al., 2023)) and one for diversity (e.g., LPIPS (Zhang et al., 2018)). It samples trajectories that maximize the combined score to balance the two aspects (Parmar et al., 2026; Harrington et al., 2026).

These considerations motivate two aspects of our scoring design. (i) We argue that diversity and preference scoring can benefit from a unified visual representation. Preference models are typically fine-tuned from CLIP (Wu et al., 2023; Kirstain et al., 2023), and diversity is measured by distances



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

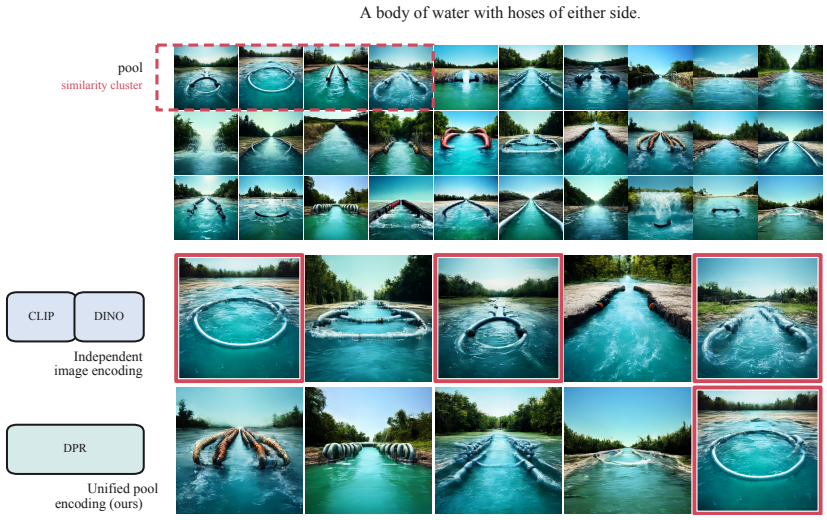



Figure 1: **Motivating example of our unified pool-conditioned scoring.** Both methods select images from the same illustrative candidate pool. The red dashed outline indicates a similarity cluster containing a tight group of candidates identified using SSCD similarity. Red outlines mark candidates from the highlighted similarity cluster. SGI (Parmar et al., 2026) retains several members of this cluster, potentially because it encodes each candidate independently. In contrast, our DPR encodes candidates together and reads both diversity and preference from their shared representations, yielding a more varied image subset.

in comparable embedding spaces (Zhang et al., 2018; Fu et al., 2023). These connections suggest an opportunity to share visual features, but separately trained scorers prevent either supervision from informing the other. We therefore learn one representation under both diversity and preference supervision, allowing both signals to shape the features used for scoring (see Table 2). (ii) We further emphasize that capturing a candidate's distinctiveness requires considering the other images in the pool. For example, a side view offers a new alternative among front views, but less so among other side views. However, an encoder that processes each image independently produces the same representation in both cases, leaving these relationships to the subsequent scoring rule. We therefore encode candidates together, allowing their representations to reflect these relationships (see Table 3).

Building on these motivations, we present Diversity Preference Ranker (DPR), a lightweight ranker that replaces separate diversity and preference scorers for subset selection. Our key idea is to learn diversity and preference within a shared representation of the candidate pool, eliminating the need for separate encoders. Specifically, we encode all candidate images together so that both scores reflect relationships within the pool. We train the representation through spatial feature distillation and human preference comparisons. However, supervision of individual criteria does not directly teach the ranker which candidate to select given the previously selected images. We therefore introduce listwise selection distillation, training the combined selection score to imitate a teacher's next-candidate choices that account for both diversity and preference.

We highlight that our unified scorer selects a more varied subset, as shown in Figure 1. DPR can serve as a shared diversity and preference scorer across reward-based generation methods. We train the scorer on HPD v2 and evaluate subset selection across four few-step generators without retraining. These results support DPR as a compact, reusable alternative to separate diversity and preference scorers, offering comparable selection performance and reduced runtime across the evaluated configurations. As a further application, we demonstrate that the same frozen scorer also supports noise optimization without retraining.

Our contributions are summarized as follows:

- We propose DPR, a unified scorer that encodes candidates together and derives diversity and preference scores from a shared visual representation. Encoding the candidate pool together allows both criteria to use information from the other candidates.

- We develop a training framework that combines spatial distillation and human preference supervision with listwise selection distillation. Teacher-guided selection decisions train the two criteria together, complementing their individual supervision.
- We evaluate the same trained scorer on subset selection across four few-step generators without retraining. Experiments examine diversity–preference trade-offs, while ablations assess the contributions of shared representations, pool context, and listwise selection distillation. We further demonstrate its application to noise optimization.

## 2 METHOD

Following the motivations in Sec. 1, our Diversity Preference Ranker (DPR) learns a shared field for diversity and preference scoring. We encode candidates together so that each encoded feature incorporates information from the other images in the pool. The learned scores guide subset selection by evaluating both candidate diversity and preference. We describe the scoring interface, the shared ranker, and its training and deployment in Secs. 2.1–2.3.

### 2.1 BACKGROUND AND PRELIMINARIES

Given a prompt $c$, a pretrained text-to-image generator $G$ (Chen et al., 2024; 2025; Sauer et al., 2024; Black Forest Labs et al., 2025) maps Gaussian noise $z$ to an image $\hat{x} = G(z, c)$. A diversification method renders a pool of $M$ candidates from independently sampled noise. We seek a subset $\mathcal{S} \subset [M]$ of size $B$ containing high-quality images that are diverse from one another. Given candidate representations $F_i$, we use per-candidate quality scores $q(F_i)$ and pairwise diversity scores $D(F_i, F_j)$ to evaluate candidate subsets.

A common selection objective combines quality and diversity through an additive set score (Parmar et al., 2026):

$$\mathcal{J}(\mathcal{S}) = \sum_{i \in \mathcal{S}} q(F_i) + \lambda \sum_{i<j \in \mathcal{S}} D(F_i, F_j), \tag{1}$$

where $\lambda$ controls the contribution of pairwise diversity to the selected batch.

Existing scoring-based diversification methods combine a visual embedding model with a separate quality scorer (Parmar et al., 2026; Harrington et al., 2026). The quality scorer evaluates text-image alignment or human preference. With independently trained encoders, supervision of one criterion cannot shape the representation used for the other. We argue that diversity and preference scoring can benefit from a shared visual representation and therefore train one field under both supervision signals. Appendix A.12 examines their interaction within the learned representation. The field's preference score provides the quality term, while distances within the field provide the diversity term.

### 2.2 SHARED REPRESENTATION FOR DIVERSITY AND PREFERENCE

Consider a ranker $R_\theta$ that evaluates both diversity and quality of $M$ image candidates $\{\hat{x}_i\}_{i=1}^M$ sharing a prompt. We represent each candidate with a spatial field that supplies both score readouts. Two images may depict the same object at different positions in the frame. A single spatially pooled feature vector does not explicitly preserve which image region each feature describes. We therefore retain a spatial grid and compare features at corresponding locations, allowing diversity scores to reflect differences in visual layout. For each image $\hat{x}_i$, the ranker produces a spatial feature field $F_i$:

$$(F_1, \ldots, F_M) = R_\theta(\hat{x}_1, \ldots, \hat{x}_M), \qquad F_i = \{f_{i,l}\}_{l=1}^C \in \mathbb{R}^{C \times d}, \tag{2}$$

where $f_{i,l}$ represents a spatial cell within $F_i$, $C$ is the number of spatial cells, and $d$ denotes the feature dimension.

Since the ranker processes all candidates in the pool at once, we adopt a lightweight feature extractor based on ImageNet-pretrained MobileNetV3-Large (Howard et al., 2019). Pooling and projecting its features onto a $4\times4$ grid gives $C{=}16$ spatial tokens $u_{i,l} \in \mathbb{R}^{d_{\text{tok}}}$, with $d_{\text{tok}}{=}256$. For each image, we combine pooled stem features with high-pass texture features and project them to one global token $g_i \in \mathbb{R}^{d_{\text{tok}}}$. Spatial tokens receive learned positional embeddings shared across candidates, while a separate shared embedding identifies the global token.

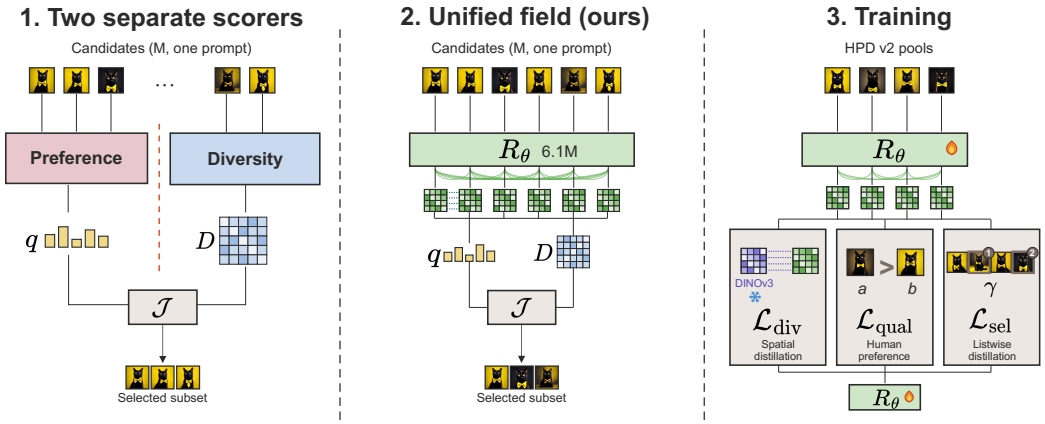



Figure 2: **Overview of DPR.** (1) Existing methods (Parmar et al., 2026) combine scores from separate quality and diversity scoring models. (2) A shared spatial field supplies both diversity $D$ and preference $q$, with self-attention incorporating information across candidates. (3) Training combines spatial feature distillation, human pairwise preference supervision, and listwise selection distillation.

The selection objective compares candidates through pairwise distances. We also incorporate candidate relationships into feature construction using self-attention across the pool. This allows each candidate's representation to use information from the alternatives available alongside it. We concatenate $\{u_{i,1}, \ldots, u_{i,C}, g_i\}_{i=1}^{M}$ into a sequence of length $M(C+1)$ and process it with two Transformer blocks. Because positional embeddings identify cells rather than candidates, reordering the images correspondingly reorders their output fields. The updated spatial tokens incorporate information from both spatial and global tokens across the pool. We project these spatial tokens to $d$=768 dimensions and normalize them to obtain the cell features $f_{i,l}$ forming $F_i$.

Both readouts operate directly on the resulting field, so that preference and diversity supervision update the same spatial features. We define the two scores as follows:

$$D(F_i, F_j) = 1 - \frac{1}{C}\sum_{l=1}^{C}\langle f_{i,l}, f_{j,l}\rangle, \tag{3}$$

$$q(F_i) = w^\top \bar{f}_i, \qquad \bar{f}_i = \frac{1}{C}\sum_{l=1}^{C} f_{i,l}, \tag{4}$$

where $w$ is a learned preference direction and $\bar{f}_i$ is the cell-averaged feature. Diversity compares corresponding spatial cells, while preference projects their average onto $w$ without taking the prompt as input. Appendix A.5 provides the detailed architecture and block diagram of the ranker.

### 2.3 Unified Training and Deployment

We train the ranker $R_\theta$ so that its shared fields $\{F_i\}_{i=1}^{M}$ support both visual comparison and human preference prediction. Human annotations provide preference orderings, while spatial targets from a frozen DINOv3 model supervise the field's geometry. We combine these signals by augmenting HPD v2 (Wu et al., 2023) with the spatial targets. Each training pool contains images sharing a prompt, their human preference comparisons, and the corresponding spatial targets. We supervise the field's geometry, its preference ordering, and the selections induced by combining the two criteria. The ranker is trained end-to-end with three complementary losses:

$$\mathcal{L} = \mathcal{L}_{\text{div}} + \mathcal{L}_{\text{qual}} + \mathcal{L}_{\text{sel}}.$$

**Diversity loss.** A frozen DINOv3 teacher (Siméoni et al., 2025) provides target fields $T_i = \{t_{i,l}\}_{l=1}^{C}$ that anchor the student's visual geometry. We average and normalize its patch features within each cell of a $4\times4$ grid matching the student's field. The student matches these targets at corresponding

locations through

$$\mathcal{L}_{\text{div}} = \mathbb{E}_i\Big[1 - \frac{1}{C}\sum_l \langle f_{i,l}, t_{i,l}\rangle\Big], \tag{5}$$

where $t_{i,l}$ denotes the teacher feature at cell $l$ of candidate $i$.

**Preference loss.** Spatial feature matching provides a visual comparison space, but does not specify which image a person would prefer. We fit the field and its preference direction $w$ using score differences, without requiring absolute preference targets across prompts. For each annotated pair $(i, j) \in \mathcal{H}$, let $I_{ij} = +1$ if image $i$ is preferred over $j$, and $I_{ij} = -1$ otherwise. We minimize

$$\mathcal{L}_{\text{qual}} = \mathbb{E}_{(i,j)\sim\mathcal{H}}\Big[\text{softplus}\big(-I_{ij}(q(F_i) - q(F_j))/\tau_{\text{q}}\big)\Big], \tag{6}$$

where $\mathcal{H}$ contains the annotated preference pairs and $\tau_{\text{q}} > 0$ rescales the signed score difference. Smaller $\tau_{\text{q}}$ penalizes incorrectly ordered pairs more strongly.

**Listwise selection distillation.** Our spatial and preference losses shape a common field, but do not directly supervise the choices made by combining their readouts. Following sequential diversified ranking (Zhu et al., 2014; Xia et al., 2015), listwise selection distillation supervises these combined decisions given the previously selected images. We construct a selection teacher that combines distances between DINOv3 target fields $T_i$ with preference estimates from smoothed human win rates.

The teacher and student use the same selection rule, but obtain their scores from different sources. The student uses distances $D(F_i, F_j)$ and preference scores $q(F_i)$ from its learned fields. Differences in score scales can affect the balance between diversity and preference, independent of the chosen trade-off weight. For both teacher and student, we divide $D$ by its mean off-diagonal entry and standardize $q$ within the pool to obtain $\hat{D}$ and $\hat{q}$, respectively. We express this shared rule using the student's normalized scores:

$$s_j(\mathcal{S};\gamma) = d_j(\mathcal{S}) + \gamma\hat{q}(F_j), \qquad d_j(\mathcal{S}) = \begin{cases} \frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}} \hat{D}(F_j, F_s), & \mathcal{S} \neq \varnothing, \\ \frac{1}{M}\sum_{s=1}^{M} \hat{D}(F_j, F_s), & \mathcal{S} = \varnothing. \end{cases} \tag{7}$$

Here $\gamma$ weights preference, while $d_j(\mathcal{S})$ measures distance to the selected candidates. Averaging distances removes the explicit scaling with the number of selected images that a sum would introduce. For the initial choice, the selected set is empty, so the full pool serves as the reference for the diversity term. The training rollout uses mean distances, whereas the deployment objective in Eq. (1) sums pairwise contributions. The sampled $\gamma$ specifies the training trade-off and is not a learned parameter.

We sample $\gamma \sim p(\gamma)$ to include both diversity-only choices and choices that also account for preference. At each step $k \in \{1, \ldots, B\}$, the teacher selects the highest-scoring available candidate $j_k^\star$. Starting from $\mathcal{S}_1 = \varnothing$, we update the selected set as $\mathcal{S}_{k+1} = \mathcal{S}_k \cup \{j_k^\star\}$. The ordered candidate indices $(j_1^\star, \ldots, j_B^\star)$ form the teacher's selection sequence. Listwise selection distillation trains the student to maximize the log-likelihood of this sequence. The loss averages the negative log-probability of each choice conditioned on the previously selected images:

$$\mathcal{L}_{\text{sel}} = \mathbb{E}_{\gamma\sim p(\gamma),\,\text{pool}}\Big[-\frac{1}{B}\sum_{k=1}^{B}\log\frac{\exp\big(s_{j_k^\star}(\mathcal{S}_k;\gamma)/\tau\big)}{\sum_{j\notin\mathcal{S}_k}\exp\big(s_j(\mathcal{S}_k;\gamma)/\tau\big)}\Big], \tag{8}$$

Here $s_j$ uses the student's scores, and $\mathcal{S}_k$ contains the teacher's $k-1$ previous picks. The temperature $\tau$ controls the student's selection probabilities. At each step, the softmax compares the teacher's chosen candidate against all remaining candidates, using the teacher's selection history. At inference, $\lambda$ therefore combines scores trained to support these pool-dependent selection decisions, rather than independently learned criteria. Appendix A.6 specifies $p(\gamma)$, the temperatures, and the remaining training settings.

Listwise selection distillation does not guarantee identical teacher choices at deployment, where the scoring rule differs from training. Appendix A.2 interprets the listwise loss through combined selection margins and derives sufficient conditions for greedy selection stability under this mismatch.

After training, the same frozen ranker can be incorporated into diversity-improving techniques that employ reward models. We focus on subset selection, replacing the separate scoring models in SGI with DPR without retraining for any evaluated generator. Appendix A.7 provides training and selection pseudocode, and Appendix A.9 specifies the deployed objective and settings.

Table 1: **Quantitative results.** Selection comparisons, with DPR denoted as Ours, on MS-COCO prompts with $B$=5 outputs and selection pools of $M$=30. Bold marks the better mean within each scorer pair. Vanilla sampling, CADS, and CNO provide additional context. Small gray intervals below DPR means report pointwise 95% paired bootstrap CIs for the difference from the original scorer (DPR minus original). Bold indicates mean comparisons, not statistical significance. See Appendix A.8 for the bootstrap protocol.

| | | Diversity | | | Preference | | Alignment |
|---|---|---|---|---|---|---|---|
| Model | Method | Vendi↑ | MSS↓ | LPIPS↑ | HPS↑ | Pick↑ | CLIP↑ |
| PixArt-$\alpha$-DMD | Vanilla sampler | 3.6150 | 0.4471 | 0.4741 | 28.2760 | 22.1829 | 31.0910 |
| | CADS | 3.7484 | 0.4173 | 0.4900 | 28.0726 | 22.1180 | 31.0644 |
| | CNO | 3.7997 | 0.4055 | 0.5224 | 28.3332 | 22.1727 | 31.0781 |
| | SGI | 3.8485 | 0.3942 | **0.5315** | 28.1948 | 22.1445 | **31.4717** |
| | Ours | **3.8594** | **0.3913** | 0.5308 | **28.2081** | **22.1464** | 31.0644 |
| | Δ CI | [+0.0045, +0.0174] | [-0.0044, -0.0015] | [-0.0014, +0.0002] | [-0.0123, +0.0388] | [-0.0064, +0.0101] | [-0.4312, -0.3834] |
| SANA-Sprint | Vanilla sampler | 4.1535 | 0.3253 | 0.5681 | 30.2045 | 22.8876 | 31.5099 |
| | CADS | 4.1603 | 0.3235 | 0.5703 | 30.1699 | 22.8796 | 31.4935 |
| | CNO | 4.2666 | 0.2958 | 0.5964 | 30.1265 | 22.8626 | 31.5063 |
| | SGI | 4.3057 | 0.2854 | 0.6113 | 29.7408 | **22.7058** | **31.7085** |
| | Ours | **4.3137** | **0.2831** | **0.6117** | **29.7739** | 22.6943 | 31.3905 |
| | Δ CI | [+0.0015, +0.0145] | [-0.0040, -0.0006] | [-0.0004, +0.0013] | [+0.0023, +0.0635] | [-0.0227, +0.0001] | [-0.3494, -0.2873] |
| SDXL-Turbo | Vanilla sampler | 4.1920 | 0.3125 | 0.5855 | 28.9842 | 22.6444 | 31.7795 |
| | CADS | 4.1982 | 0.3109 | 0.5866 | 28.9667 | 22.6400 | 31.7668 |
| | CNO | 4.2850 | 0.2883 | 0.6103 | 28.9457 | 22.6161 | 31.7033 |
| | SGI | 4.3785 | 0.2634 | 0.6284 | 28.8035 | 22.5785 | **31.8848** |
| | Ours | **4.3844** | **0.2622** | **0.6307** | **28.8316** | **22.5817** | 31.6416 |
| | Δ CI | [-0.0012, +0.0128] | [-0.0031, +0.0007] | [+0.0014, +0.0032] | [-0.0093, +0.0654] | [-0.0087, +0.0152] | [-0.2759, -0.2106] |
| FLUX.1-schnell | Vanilla sampler | 4.5219 | 0.2205 | 0.6459 | 29.4847 | 22.6454 | 31.5328 |
| | CADS | 4.5116 | 0.2235 | 0.6414 | 29.4397 | 22.6379 | 31.5602 |
| | CNO | 4.5698 | 0.2063 | 0.6632 | 29.4338 | 22.6194 | 31.5346 |
| | SGI | 4.6706 | 0.1739 | 0.6949 | 28.9665 | **22.5201** | **31.7272** |
| | Ours | **4.6716** | **0.1732** | **0.6994** | **29.0039** | 22.5159 | 31.4155 |
| | Δ CI | [-0.0038, +0.0058] | [-0.0023, +0.0008] | [+0.0034, +0.0055] | [+0.0042, +0.0714] | [-0.0146, +0.0062] | [-0.3419, -0.2815] |

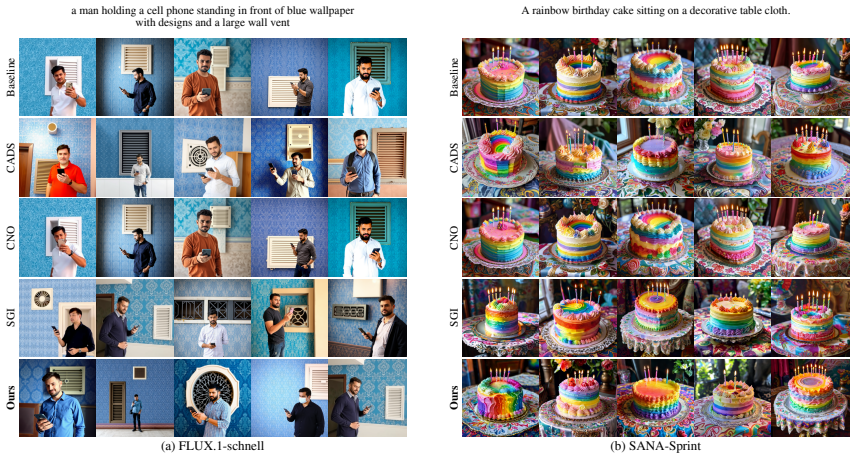
# 3 EXPERIMENTS

We evaluate DPR as a ranker for selecting diverse, preferred images from candidate pools. We compare it with the original selection scorer under matched sampling conditions and examine the diversity–preference trade-off. We then ablate the representation and training objectives introduced in Sec. 2.3, before evaluating noise optimization as a further application.

## 3.1 SETUP

**Backbones.** We evaluate on four few-step distilled backbones, PixArt-$\alpha$-DMD (Chen et al., 2024), SANA-Sprint (Chen et al., 2025), SDXL-Turbo (Sauer et al., 2024), and FLUX.1-schnell (Black Forest Labs et al., 2025), all frozen with their released weights and default schedules. We train the ranker on external preference data and exclude candidate pools from these downstream evaluations.

**Selection setup.** SGI (Parmar et al., 2026) formulates subset selection as a quadratic integer program combining CLIP similarity (Hessel et al., 2021) and DINOv2 cosine distance (Oquab et al., 2023). We use its greedy solver to select $B$ candidates from $M$=30. To isolate the effect of replacing the separate scorers with DPR, we retain SGI's candidate pool and selection rule. The training-form rule with $\gamma = 0.06$ yields comparable diversity and preference on PixArt-$\alpha$-DMD, as shown in our rule-mismatch comparison (App. A.2, Table 5). For Table 1, SGI uses $\lambda$=1.5, while DPR uses a weight tuned on a held-out split (App. A.9). CADS (Sadat et al., 2024) and CNO (Kim et al., 2025) carry no external scorer and are reported as context (see Appendix A.1 for related approaches). All backbones run at their released resolution.




(a) FLUX.1-schnell (b) SANA-Sprint

Figure 3: **Qualitative comparisons on FLUX.1-schnell and SANA-Sprint.** Rows show Baseline (vanilla sampling), CADS, CNO, SGI, and Ours, with all $B$=5 outputs shown for each method. SGI and Ours select from the same candidate pool using the configurations in Table 1.

**Protocol.** We use MS-COCO 2014 (Lin et al., 2014) validation captions (Sadat et al., 2024; Corso et al., 2024), with $B$=5 images per prompt and metrics averaged over prompts. We exclude prompts matching MS-COCO captions from the HPD v2 training set to prevent overlap with evaluation data. For a controlled selection comparison, both scorer arms share the prompts, seeds, candidate pools, and subset solver. Appendix A.9 specifies the selection objective and deployed settings. Table 1 reports means and pointwise paired bootstrap confidence intervals for scorer differences, conditional on fixed checkpoints and sampling runs (App. A.8). All runs use one NVIDIA RTX PRO 6000 Blackwell Max-Q Workstation Edition GPU with 96 GB of memory.

**Metrics.** We measure diversity using Vendi (Friedman & Dieng, 2023) and mean pairwise similarity (MSS) on SSCD embeddings (Pizzi et al., 2022), together with pairwise LPIPS (Zhang et al., 2018). We evaluate human preference using HPSv2 (Wu et al., 2023) and PickScore (Kirstain et al., 2023), and text alignment using CLIPScore (Hessel et al., 2021).

### 3.2 Main Results

**Quantitative results.** Table 1 compares DPR with SGI. Observe that DPR achieves higher mean Vendi and HPS and lower MSS across all evaluated backbones. Overall, DPR provides comparable diversity and preference with a compact unified scorer. LPIPS and PickScore differences vary across backbones. The lower CLIPScore relative to SGI is expected given that SGI directly uses CLIP similarity for selection, whereas DPR uses a learned human preference score. We highlight that the same frozen ranker serves all evaluated generators, supporting a compact alternative to separate scoring models without generator-specific retraining. Table 10 reports the scorer parameter and memory reductions. DPR also reduces end-to-end runtime across the evaluated selection and optimization configurations (Table 11 in the appendix).

**Qualitative comparison.** Figure 3 compares the selected images on FLUX.1-schnell and SANA-Sprint. A notable difference in the FLUX example is that Ours selects candidates spanning a wider range of subject scales and framing. The SANA-Sprint example shows variation in cake decoration and frosting patterns while retaining the rainbow colors and decorative tablecloth. These examples illustrate how changing the scorer can select different visual alternatives from the same candidate pool, without modifying the generator. Appendix A.14 provides additional selection examples.

**Trade-off frontiers.** Table 1 compares deployed operating points, while Figure 4 sweeps $\lambda$ for both SGI and Ours. Ours reaches higher Vendi and HPS than SGI within the evaluated ranges. The larger gains at larger pool sizes (Appendix A.3) suggest that DPR's relative advantage can increase with more candidates.

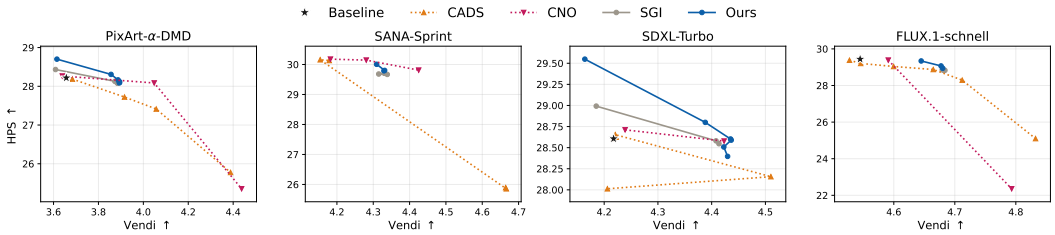



Figure 4: **Selection trade-off curves.** Each panel shows one backbone. SGI and Ours sweep the diversity weight $\lambda$ using the same subset-selection procedure. CADS and CNO sweep their noise scale and learning rate, respectively, while Baseline denotes vanilla sampling. Selection settings follow Sec. 3.1.

Table 2: **From two scorers to one field.** Representation-sharing ablations on PixArt-$\alpha$-DMD ($B$=5). Variants share the DINOv3 teacher, HPD v2 training pools, pool attention, and parameter budget. SGI uses the same candidate pools ($M$=30), raw readouts, and $\lambda$=7.10. DivGen rescales weights by each variant's within-pool score spread. DivGen is the further application described in Sec. 3.4. Best per column in bold, second underlined.

| | SGI (selection) | | | | | DivGen (optimization) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Scoring architecture | Vendi↑ | MSS↓ | LPIPS↑ | HPS↑ | Pick↑ | Vendi↑ | MSS↓ | LPIPS↑ | HPS↑ | Pick↑ |
| (i) Two separate nets (2×3.0M) | 3.8710 | 0.3897 | **0.5325** | **28.2053** | **22.0827** | 4.0430 | 0.3440 | <u>0.5741</u> | <u>28.0231</u> | **22.0439** |
| (ii) Shared encoder, decoupled heads | <u>3.8716</u> | <u>0.3896</u> | 0.5298 | <u>28.1684</u> | <u>22.0681</u> | <u>4.0525</u> | <u>0.3428</u> | 0.5694 | 27.8961 | 22.0039 |
| (iii) DPR (unified field) | **3.8934** | **0.3852** | <u>0.5312</u> | 28.1227 | 22.0560 | **4.0762** | **0.3375** | **0.5768** | **28.0809** | <u>22.0257</u> |

### 3.3 Analyses

**Representation sharing.** Table 2 compares separate networks, a shared encoder with decoupled heads, and the unified field for selection and further optimization. The unified field achieves the highest Vendi and lowest MSS in both hosts. Separate networks retain the highest LPIPS, HPS, and PickScore in selection, whereas the unified field achieves the highest LPIPS and HPS in optimization. Separate networks retain the highest PickScore in both hosts. These comparisons show that representation sharing changes the diversity–preference trade-off, with preference effects depending on the deployment procedure.

**Component ablation.** Table 3 evaluates pool attention, listwise selection distillation, spatial supervision, and model capacity. Removing pool attention lowers diversity and HPS, but increases PickScore. Removing listwise selection distillation yields the largest diversity drop among single-component removals. This supports directly supervising selection decisions in addition to learning the diversity and preference criteria individually. The plain student removes both components, and the full model achieves higher diversity and HPS than this baseline. The model-width ablation suggests that the 6.1M-parameter ranker provides sufficient capacity for diversity-aware selection. Taken together, these analyses support learning diversity and preference within a unified field, with pool context and listwise supervision further improving diversity-aware selection.

### 3.4 Further Application

We further apply the frozen DPR to the noise optimization method of Harrington et al. (2026), which we refer to as DivGen following its official repository name. DPR provides differentiable scores without retraining, retaining DivGen's noise update procedure and regularizer. Gradients pass through the ranker and generator to update the initial noises, while both networks remain frozen.

**Setup.** The DivGen baseline uses HPSv2 preference rewards and mean pairwise DINO distance. DPR uses mean preference and mean pairwise distance from its learned representation. Both configurations share prompts, seeds, initial noises, and optimization schedules, using the metrics and hardware in Sec. 3.1. Appendix A.10 provides optimization pseudocode and specifies the objectives and

Table 3: **Component ablation of the field.** Variants remove individual components, remove pool attention and listwise selection distillation together, or vary model width. Each ranker is retrained and deployed unchanged in the SGI host on PixArt-$\alpha$-DMD ($M$=30, $B$=5). All variants share the DINOv3 teacher, HPD v2 pools, and host configuration. The plain student scores candidates independently without $\mathcal{L}_{\text{sel}}$, retaining spatial distillation and human preference supervision at a comparable parameter count. Its indented row denotes an additional removal from the preceding variant. Best per column in bold.

| Variant | Vendi↑ | MSS↓ | LPIPS↑ | HPS↑ | Pick↑ |
|---|---|---|---|---|---|
| DPR (6.1M) | **3.8934** | **0.3852** | **0.5312** | **28.1227** | 22.0560 |
| w/o attention across candidates | 3.8676 | 0.3907 | 0.5284 | 28.0988 | **22.0616** |
| w/o cell alignment (pooled-vector distillation) | 3.8681 | 0.3904 | 0.5294 | 28.0702 | 22.0505 |
| w/o listwise selection distillation ($\mathcal{L}_{\text{sel}}$) | 3.8547 | 0.3934 | 0.5273 | 28.1082 | 22.0557 |
| + w/o attention (plain student) (6.2M) | 3.8487 | 0.3943 | 0.5222 | 28.0952 | 22.0547 |
| 0.5× width (4.3M) | 3.8775 | 0.3884 | 0.5290 | 28.1185 | 22.0566 |
| 4× width (34.9M) | 3.8580 | 0.3930 | 0.5278 | 28.1223 | 22.0531 |

Table 4: **Quantitative results for noise optimization.** DivGen with HPSv2–DINO scoring is compared with DPR, denoted as Ours. Both configurations use mean pairwise distance and generate $B$=5 outputs per prompt. Bold marks the better mean within each pair. Small gray intervals report pointwise 95% paired bootstrap CIs for Ours minus the baseline. Bold indicates mean comparisons, not statistical significance. See Appendix A.8 for the bootstrap protocol.

| | | Diversity | | | Preference | | Alignment |
|---|---|---|---|---|---|---|---|
| Model | Method | Vendi↑ | MSS↓ | LPIPS↑ | HPS↑ | Pick↑ | CLIP↑ |
| PixArt-$\alpha$-DMD | DivGen (HPS+DINO) | 4.0090 | 0.3537 | 0.5671 | **28.3716** | **22.1090** | **31.0292** |
| | Ours | **4.0315** | **0.3476** | **0.5701** | 28.0651 | 22.0663 | 30.9808 |
| | Δ CI | [+0.0081, +0.0371] | [-0.0097, -0.0026] | [+0.0007, +0.0052] | [-0.3604, -0.2537] | [-0.0600, -0.0255] | [-0.1004, +0.0028] |
| SANA-Sprint | DivGen (HPS+DINO) | 4.2699 | 0.2947 | 0.5993 | **30.3382** | **22.8000** | **31.8457** |
| | Ours | **4.3068** | **0.2832** | **0.6089** | 30.0910 | 22.7475 | 31.7830 |
| | Δ CI | [+0.0270, +0.0470] | [-0.0140, -0.0090] | [+0.0080, +0.0110] | [-0.2870, -0.2070] | [-0.0670, -0.0380] | [-0.1050, -0.0210] |
| SDXL-Turbo | DivGen (HPS+DINO) | 4.3179 | 0.2780 | 0.6207 | **28.1611** | **22.5693** | **31.2102** |
| | Ours | **4.3195** | **0.2763** | **0.6231** | 27.9848 | 22.5413 | 31.1875 |
| | Δ CI | [-0.0113, +0.0146] | [-0.0051, +0.0017] | [+0.0003, +0.0045] | [-0.2285, -0.1239] | [-0.0449, -0.0109] | [-0.0763, +0.0334] |

deployed settings. At the released resolution, DivGen does not fit in memory on FLUX.1-schnell, so we omit that pair.

**Quantitative results.** Table 4 shows that DPR achieves higher mean diversity across all evaluated backbones under the same mean pairwise objective. The baseline achieves higher HPS and PickScore across backbones. These results demonstrate that the same frozen ranker can guide optimization toward more diverse outputs, suggesting potential applications to other reward-based diversification methods. Qualitative comparisons also illustrate varied viewpoints and spatial arrangements in the optimized outputs (Appendix A.15).

## 4 Conclusion and Limitations

We presented DPR, a lightweight ranker that learns diversity and preference within a shared, pool-conditioned representation. Spatial distillation and human preference comparisons supervise the shared field, while listwise distillation trains the combined selection decisions. Across four accelerated generators, the same frozen ranker provides comparable selection performance with reduced runtime. Its application to noise optimization further demonstrates reuse beyond subset selection. The current preference readout uses only visual features, leaving explicit text conditioning for future work. This limitation is discussed in Appendix A.13.

## AI USE STATEMENT

We used AI tools to assist with drafting and polishing the manuscript, retrieving and discovering relevant literature, and research ideation and execution, including experimental design, code development, and analysis of results. The authors take full responsibility for the final manuscript, experimental results, and conclusions.

## REFERENCES

Qingyao Ai, Keping Bi, Jiafeng Guo, and W. Bruce Croft. Learning a deep listwise context model for ranking refinement. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, pp. 135–144, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356572. doi: 10.1145/3209978.3209985. URL https://doi.org/10.1145/3209978.3209985.

Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J Fleet. Synthetic data from diffusion models improves imagenet classification. *arXiv preprint arXiv:2304.08466*, 2023.

Irwan Bello, Sayali Kulkarni, Sagar Jain, Craig Boutilier, Ed Chi, Elad Eban, Xiyang Luo, Alan Mackey, and Ofer Meshi. Seq2slate: Re-ranking and slate optimization with rnns. *arXiv preprint arXiv:1810.02019*, 2018.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. FLUX.1 Kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pp. 129–136, New York, NY, USA, 2007. Association for Computing Machinery. ISBN 9781595937933. doi: 10.1145/1273496.1273513. URL https://doi.org/10.1145/1273496.1273513.

Jaime Carbonell and Jade Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pp. 335–336, New York, NY, USA, 1998. Association for Computing Machinery. ISBN 1581130155. doi: 10.1145/290941.291025. URL https://doi.org/10.1145/290941.291025.

Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-$\alpha$: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *International conference on learning representations*, volume 2024, pp. 57611–57640, 2024.

Junsong Chen, Shuchen Xue, Yuyang Zhao, Jincheng Yu, Sayak Paul, Junyu Chen, Han Cai, Song Han, and Enze Xie. Sana-sprint: One-step diffusion with continuous-time consistency distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16185–16195, 2025.

Gabriele Corso, Yilun Xu, Valentin De Bortoli, Regina Barzilay, and Tommi Jaakkola. Particle guidance: non-iid diverse sampling with diffusion models. In *International Conference on Learning Representations*, volume 2024, pp. 22480–22507, 2024.

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. In *Forty-first international conference on machine learning*, 2024.

Dan Friedman and Adji Bousso Dieng. The Vendi Score: A diversity evaluation metric for machine learning. *Transactions on Machine Learning Research*, 2023.

Stephanie Fu, Netanel Yakir Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. Dreamsim: Learning new dimensions of human visual similarity using synthetic data. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=DEiNSfh1k7.

Rohit Gandikota and David Bau. Distilling diversity and control in diffusion models. In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1304–1313, 2026. doi: 10.1109/WACV61042.2026.00133.

Anne Harrington, A Koepke, Shyamgopal Karthik, Trevor Darrell, and Alexei A Efros. It's never too late: Noise optimization for collapse recovery in trained diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 43124–43134, 2026.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 7514–7528, 2021.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1314–1324, 2019.

Byungjun Kim, Soobin Um, and Jong Chul Ye. Diverse text-to-image generation via contrastive noise optimization. *arXiv preprint arXiv:2510.03813*, 2025.

Michael Kirchhof, James Thornton, Louis Béthune, Pierre Ablin, Eugene Ndiaye, and Marco Cuturi. Shielded diffusion: Generating novel and diverse images using sparse repellency. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 30911–30942, 2025.

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. *Advances in neural information processing systems*, 36:36652–36663, 2023.

Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. *Advances in Neural Information Processing Systems*, 37:122458–122483, 2024.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International conference on machine learning*, pp. 3744–3753. PMLR, 2019.

Haoyu Li, Tingyan Wen, Lin Qi, Zhe Wu, Yihuang Chen, Xing Zhou, Lifei Zhu, Xueqian Wang, and Kai Zhang. 1.x-Distill: Breaking the diversity, quality, and efficiency barrier in distribution matching distillation. *arXiv preprint arXiv:2604.04018*, 2026.

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 *European conference on computer vision*, pp. 740–755. Springer, 2014.

Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14297–14306. IEEE, 2023.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Liang Pang, Jun Xu, Qingyao Ai, Yanyan Lan, Xueqi Cheng, and Jirong Wen. Setrank: Learning a permutation-invariant ranking model for information retrieval. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, pp. 499–508, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450380164. doi: 10.1145/3397271.3401104. URL https://doi.org/10.1145/3397271.3401104.

Gaurav Parmar, Or Patashnik, Daniil Ostashev, Kuan-Chieh Wang, Kfir Aberman, Srinivasa Narasimhan, and Jun-Yan Zhu. Scaling group inference for diverse and high-quality generation. In *International Conference on Learning Representations*, 2026.

Ed Pizzi, Sreya Dutta Roy, Sugosh Nagavara Ravindra, Priya Goyal, and Matthijs Douze. A self-supervised descriptor for image copy detection. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14512–14522. IEEE, 2022.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

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 *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22500–22510. IEEE, 2023.

Seyedmorteza Sadat, Jakob Buhmann, Derek Bradley, Otmar Hilliges, and Romann Weber. Cads: Unleashing the diversity of diffusion models through condition-annealed sampling. In *International Conference on Learning Representations*, volume 2024, pp. 23723–23755, 2024.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pp. 87–103. Springer, 2024.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Soobin Um and Jong Chul Ye. Minority-focused text-to-image generation via prompt optimization. *arXiv preprint arXiv:2410.07838*, 2024.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023.

Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. Listwise approach to learning to rank: theory and algorithm. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pp. 1192–1199, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605582054. doi: 10.1145/1390156.1390306. URL https://doi.org/10.1145/1390156.1390306.

Long Xia, Jun Xu, Yanyan Lan, Jiafeng Guo, and Xueqi Cheng. Learning maximal marginal relevance model via directly optimizing diversity evaluation measures. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, pp. 113–122, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450336215. doi: 10.1145/2766462.2767710. URL https://doi.org/10.1145/2766462.2767710.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6613–6623. IEEE, 2024.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

Yadong Zhu, Yanyan Lan, Jiafeng Guo, Xueqi Cheng, and Shuzi Niu. Learning for search result diversification. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '14, pp. 293–302, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450322577. doi: 10.1145/2600428.2609634. URL https://doi.org/10.1145/2600428.2609634.

# A Appendix

## A.1 Related Work

**Improving diversity in diffusion models.** Beyond the latest scoring-based approaches (Parmar et al., 2026; Harrington et al., 2026), a long-standing line of work has focused on improving diversity in diffusion models. These methods mostly intervene at inference time, incorporating guidance signals to steer sampling trajectories toward more diverse outcomes (Corso et al., 2024; Um & Ye, 2024; Kirchhof et al., 2025; Kim et al., 2025). Other instances include perturbing the conditioning signal (Sadat et al., 2024) or using a reduced CFG guidance scale early in sampling (Kynkäänniemi et al., 2024). We instead learn a shared diversity and preference scorer supervised through both individual criteria and their combined selection decisions.

**Learning to rank and diversification.** Diversified ranking balances relevance and novelty, as in maximal marginal relevance (MMR) (Carbonell & Goldstein, 1998). Listwise objectives supervise rankings over candidate lists, as developed in ListNet and ListMLE (Cao et al., 2007; Xia et al., 2008). R-LTR and PAMM learn ranking functions that combine relevance and diversity (Zhu et al., 2014; Xia et al., 2015), while Seq2Slate models successive choices conditioned on preceding selections (Bello et al., 2018). Set Transformer models interactions among set elements through attention (Lee et al., 2019), and context-aware rankers use candidate interactions to inform ranking scores (Ai et al., 2018; Pang et al., 2020). DPR brings these architectural and supervision intuitions to generated image pools. Cross-candidate attention produces a shared visual representation for diversity and preference, while listwise selection distillation supervises the teacher's next choice among remaining candidates given the selected subset.

## A.2 Conditional selection stability

We analyze when listwise selection distillation preserves teacher choices under a different greedy scoring rule. The following sufficient condition concerns a fixed pool and fixed ranker readouts. We provide a complete proof and apply the condition to the mean–sum discrepancy. The analysis concerns a greedy selector, without assuming that every deployment solver follows this procedure.

**Setup.** Fix a pool of $M$ candidates, its ranker readouts, a weight $\gamma$, and a teacher sequence $(j_1^\star, \dots, j_B^\star)$ of distinct candidate indices, with $1 \le B \le M$. Teacher ties are resolved by a fixed rule. Let $\mathcal{S}_k = \{j_1^\star, \dots, j_{k-1}^\star\}$, $\mathcal{A}_k = [M] \setminus \mathcal{S}_k$, and $\ell_k$ be the unaveraged stepwise cross-entropy in Eq. (8). For an alternative greedy score $r_\theta(j \mid \mathcal{S})$, define

$$\begin{aligned} h_k(j) &= r_\theta(j \mid \mathcal{S}_k) - s_j(\mathcal{S}_k; \gamma), \\ \kappa_k &= \max_{j \in \mathcal{A}_k} h_k(j) - \min_{j \in \mathcal{A}_k} h_k(j). \end{aligned} \tag{9}$$

The range $\kappa_k$ measures distortion of relative scores at the teacher prefix.

**Notation.** For the proof, write

$$a_k^T := j_k^\star, \qquad \mathcal{S}_k^T := \mathcal{S}_k, \qquad s_k(j) := s_j(\mathcal{S}_k; \gamma).$$

We also write $s_\theta(j \mid \mathcal{S}; \gamma) := s_j(\mathcal{S}; \gamma)$ when displaying the dependence on the selected subset. The teacher indices are pairwise distinct, so $a_k^T \in \mathcal{A}_k = [M] \setminus \mathcal{S}_k^T$ at every step. Distinct indices may still refer to identical image content. The unaveraged cross-entropy for the next teacher choice is

$$\ell_k = -\log \frac{\exp(s_k(a_k^T)/\tau)}{\sum_{j \in \mathcal{A}_k} \exp(s_k(j)/\tau)}, \qquad \tau > 0. \tag{10}$$

At each prefix $\mathcal{S}$, a candidate-independent positive affine transformation preserves the greedy choice. In particular, $r_\theta(j \mid \mathcal{S}) = \alpha(\mathcal{S}) r_\theta^{\mathrm{raw}}(j \mid \mathcal{S}) + \beta(\mathcal{S})$ is equivalent when $\alpha(\mathcal{S}) > 0$. We fix this score representation before computing $h_k$ and $\kappa_k$ in Eq. (9). The distortion range depends on the chosen scale, but not on a common additive offset.

**Proposition 1** (Selection preservation under score mismatch)**.** *Under the setup above, assume $\tau > 0$ and finite scores. If every teacher-prefix step satisfies*

$$\ell_k < \log(1 + \exp(-\kappa_k/\tau)), \qquad k = 1, \dots, B, \tag{11}$$

*then a selector starting from the empty set and greedily maximizing $r_\theta$ over remaining candidates reproduces the entire teacher sequence.*

*Proof of Proposition 1.* Fix a teacher prefix $\mathcal{S}_k^T$ and write $a = a_k^T$. If $|\mathcal{A}_k| = 1$, then $\mathcal{A}_k = \{a\}$ and the greedy choice necessarily agrees with the teacher. For the one-step margin argument below, assume $|\mathcal{A}_k| \geq 2$. Finite scores imply $\ell_k > 0$, so $\log(\exp(\ell_k) - 1)$ is well-defined. Expanding the cross-entropy in Eq. (10) gives

$$\begin{aligned}
\ell_k &= -\log \frac{\exp(s_k(a)/\tau)}{\sum_{j\in\mathcal{A}_k} \exp(s_k(j)/\tau)} \\
&= \log \frac{\sum_{j\in\mathcal{A}_k} \exp(s_k(j)/\tau)}{\exp(s_k(a)/\tau)} \\
&= \log \sum_{j\in\mathcal{A}_k} \exp\left(\frac{s_k(j) - s_k(a)}{\tau}\right) \\
&= \log \left[1 + \sum_{j\in\mathcal{A}_k\setminus\{a\}} \exp\left(\frac{s_k(j) - s_k(a)}{\tau}\right)\right].
\end{aligned} \tag{12}$$

The last equality separates the teacher candidate, whose contribution inside the logarithm is one.

This expansion also clarifies the role of listwise supervision. For each remaining candidate $j \neq a$, the combined selection margin is

$$m_{k,j} = s_k(a) - s_k(j) = d_a(\mathcal{S}_k) - d_j(\mathcal{S}_k) + \gamma\big[\hat{q}(F_a) - \hat{q}(F_j)\big].$$

Each competing term in Eq. (12) is therefore $\exp(-m_{k,j}/\tau)$. The loss directly supervises these combined margins, with diversity-only supervision when $\gamma = 0$. We now bound the margins to establish selection preservation.

For any remaining competitor $b \neq a$, exponentiating and retaining its term yields

$$\begin{aligned}
\exp(\ell_k) - 1 &= \sum_{j\in\mathcal{A}_k\setminus\{a\}} \exp\left(\frac{s_k(j) - s_k(a)}{\tau}\right) \\
&\geq \exp\left(\frac{s_k(b) - s_k(a)}{\tau}\right), \\
\log\big(\exp(\ell_k) - 1\big) &\geq \frac{s_k(b) - s_k(a)}{\tau}, \\
s_k(a) - s_k(b) &\geq -\tau \log\big(\exp(\ell_k) - 1\big).
\end{aligned} \tag{13}$$

Next, the assumed loss bound in Eq. (11) implies

$$\begin{aligned}
&\ell_k < \log(1 + \exp(-\kappa_k/\tau)) \\
\implies\quad &\exp(\ell_k) - 1 < \exp(-\kappa_k/\tau) \\
\implies\quad &\log\big(\exp(\ell_k) - 1\big) < -\kappa_k/\tau \\
\implies\quad &-\tau \log\big(\exp(\ell_k) - 1\big) > \kappa_k.
\end{aligned} \tag{14}$$

The last implication reverses the inequality because $-\tau < 0$. Combining Eqs. (13) and (14), the teacher candidate has a training-score margin greater than $\kappa_k$ against every competitor. Its margin under the alternative selection score therefore satisfies

$$\begin{aligned}
r_\theta(a \mid \mathcal{S}_k^T) - r_\theta(b \mid \mathcal{S}_k^T) &= s_k(a) + h_k(a) - s_k(b) - h_k(b) \\
&= \big(s_k(a) - s_k(b)\big) - \big(h_k(b) - h_k(a)\big) \\
&\geq \big(s_k(a) - s_k(b)\big) - \left(\max_{j\in\mathcal{A}_k} h_k(j) - \min_{j\in\mathcal{A}_k} h_k(j)\right) \\
&= s_k(a) - s_k(b) - \kappa_k \\
&\geq -\tau \log\big(\exp(\ell_k) - 1\big) - \kappa_k \\
&> 0.
\end{aligned} \tag{15}$$

The first inequality uses the definition of the distortion range. Thus, $a_k^T$ uniquely maximizes $r_\theta$ among the remaining candidates.

Finally, let $\mathcal{S}_k^\theta$ denote the prefix produced by the greedy selector. Both procedures start with $\mathcal{S}_1^\theta = \mathcal{S}_1^T = \varnothing$. Whenever their prefixes agree, the preceding argument gives

$$\begin{aligned} a_k^\theta &= \arg\max_{j \notin \mathcal{S}_k^\theta} r_\theta(j \mid \mathcal{S}_k^\theta) \\ &= \arg\max_{j \notin \mathcal{S}_k^T} r_\theta(j \mid \mathcal{S}_k^T) \\ &= a_k^T, \\ \mathcal{S}_{k+1}^\theta &= \mathcal{S}_k^\theta \cup \{a_k^\theta\} \\ &= \mathcal{S}_k^T \cup \{a_k^T\} = \mathcal{S}_{k+1}^T. \end{aligned} \tag{16}$$

Induction over $k = 1, \ldots, B$ proves equality of the full sequences. □

Notice that the argument requires a selection margin large enough to absorb changes in relative scores. The two score functions need not coincide. The sufficient condition remains stepwise and pool-specific, rather than a guarantee from the empirical average training loss. Table 3 evaluates the empirical effect of listwise supervision under the same deployment procedure at the reported operating point. This ablation assesses selection performance, while Proposition 1 concerns the stronger property of reproducing the teacher sequence.

**Application to mean-distance and sum-distance selection.** Consider greedy maximization of Eq. (1) using fixed raw readouts $D(F_i, F_j)$ and $q(F_j)$, with $\lambda > 0$. Let $\mu_D > 0$ be the mean off-diagonal distance, $\bar{q}$ the mean preference, and $\sigma_q > 0$ its standard deviation over the full pool. Define $\widehat{D}(F_i, F_j) = D(F_i, F_j)/\mu_D$, $\widehat{q}(F_j) = (q(F_j) - \bar{q})/\sigma_q$, and $\widetilde{\lambda} = \lambda\mu_D/\sigma_q$. For a nonempty prefix $\mathcal{S}$ of size $m$ and a candidate $j \notin \mathcal{S}$, the greedy marginal is

$$\begin{aligned} \Delta_j(\mathcal{S}) &= \mathcal{J}(\mathcal{S} \cup \{j\}) - \mathcal{J}(\mathcal{S}) \\ &= q(F_j) + \lambda \sum_{i \in \mathcal{S}} D(F_j, F_i) \\ &= \bar{q} + \sigma_q \widehat{q}(F_j) + \lambda\mu_D \sum_{i \in \mathcal{S}} \widehat{D}(F_j, F_i). \end{aligned} \tag{17}$$

The second equality cancels all terms already present in $\mathcal{S}$, using symmetry of $D$. The third substitutes the definitions of the normalized readouts. Subtracting the common offset $\bar{q}$ and dividing by $\lambda\mu_D m > 0$ preserves the maximizing candidate. The resulting equivalent score is

$$\begin{aligned} r_\theta(j \mid \mathcal{S}) &= \frac{\Delta_j(\mathcal{S}) - \bar{q}}{\lambda\mu_D m} \\ &= \frac{\sigma_q \widehat{q}(F_j) + \lambda\mu_D \sum_{i \in \mathcal{S}} \widehat{D}(F_j, F_i)}{\lambda\mu_D m} \\ &= \frac{1}{m} \sum_{i \in \mathcal{S}} \widehat{D}(F_j, F_i) + \frac{\sigma_q}{\lambda\mu_D m} \widehat{q}(F_j) \\ &= d_\theta(j \mid \mathcal{S}) + \frac{1}{\widetilde{\lambda} m} \widehat{q}(F_j). \end{aligned} \tag{18}$$

Here $d_\theta(j \mid \mathcal{S})$ is the mean-distance term in Eq. (7). At a nonempty teacher prefix, subtracting the training score gives

$$\begin{aligned} h_k(j) &= r_\theta(j \mid \mathcal{S}_k^T) - s_\theta(j \mid \mathcal{S}_k^T; \gamma) \\ &= \left[ d_\theta(j \mid \mathcal{S}_k^T) + \frac{\widehat{q}(F_j)}{\widetilde{\lambda}|\mathcal{S}_k^T|} \right] - \left[ d_\theta(j \mid \mathcal{S}_k^T) + \gamma \widehat{q}(F_j) \right] \\ &= \left( \frac{1}{\widetilde{\lambda}|\mathcal{S}_k^T|} - \gamma \right) \widehat{q}(F_j). \end{aligned} \tag{19}$$

Table 5: **Selection rules with the same frozen DPR.** PixArt-$\alpha$-DMD with $M$=30 and $B$=5 on the same 200 prompt pools. The SGI rule uses raw scores with $\lambda$=7.10. The training-form rule uses normalized scores and mean distances, including the full-pool mean distance for the first choice. Both rules use their own selected subsets during inference. Bold denotes the highest mean per column, or the lowest for MSS, rather than statistical significance.

| Selection rule | Vendi↑ | MSS↓ | LPIPS↑ | HPS↑ | Pick↑ | CLIP↑ |
|---|---|---|---|---|---|---|
| SGI | 3.8934 | 0.3852 | 0.5312 | 28.1227 | 22.0560 | **31.0005** |
| Training-form ($\gamma = 0$) | **3.8996** | **0.3829** | **0.5350** | 28.0086 | 22.0475 | 30.9785 |
| Training-form ($\gamma = 0.06$) | 3.8942 | 0.3851 | 0.5331 | **28.1723** | **22.0564** | 30.9725 |

Writing the coefficient as $c_k = 1/(\widetilde{\lambda}|\mathcal{S}_k^T|) - \gamma$, its distortion range becomes

$$\begin{aligned}\kappa_k &= \max_{j\in\mathcal{A}_k} h_k(j) - \min_{j\in\mathcal{A}_k} h_k(j) \\ &= \max_{j\in\mathcal{A}_k}(c_k\widehat{q}(F_j)) - \min_{j\in\mathcal{A}_k}(c_k\widehat{q}(F_j)) \\ &= |c_k|\left(\max_{j\in\mathcal{A}_k}\widehat{q}(F_j) - \min_{j\in\mathcal{A}_k}\widehat{q}(F_j)\right) \\ &= \left|\frac{1}{\widetilde{\lambda}|\mathcal{S}_k^T|} - \gamma\right|\left(\max_{j\in\mathcal{A}_k}\widehat{q}(F_j) - \min_{j\in\mathcal{A}_k}\widehat{q}(F_j)\right).\end{aligned} \tag{20}$$

The absolute value accounts for the reversal of the maximum and minimum when $c_k < 0$. This expression includes the mismatch without equating the training weight $\gamma$ and deployment weight $\lambda$.

For the first choice, the raw marginal is $\Delta_j(\varnothing) = q(F_j)$. We use the equivalent score $r_\theta(j \mid \varnothing) = (q(F_j) - \bar{q})/\sigma_q = \widehat{q}(F_j)$. Its difference from the initial training score is

$$\begin{aligned}h_1(j) &= r_\theta(j \mid \varnothing) - s_\theta(j \mid \varnothing; \gamma) \\ &= \widehat{q}(F_j) - \left[\frac{1}{M}\sum_{i=1}^{M}\widehat{D}(F_j, F_i) + \gamma\widehat{q}(F_j)\right] \\ &= (1-\gamma)\widehat{q}(F_j) - \frac{1}{M}\sum_{i=1}^{M}\widehat{D}(F_j, F_i).\end{aligned} \tag{21}$$

Taking $\kappa_1 = \max_j h_1(j) - \min_j h_1(j)$ includes this first-choice discrepancy in the same sufficient condition. All these identities assume the stated normalization and readouts fixed over the full pool.

**Scope.** The proposition gives a sufficient condition for preserving the teacher's greedy choices at each step of a fixed pool. Identical duplicate inputs with equal logits impose an additional restriction on the index-level condition. If $r \geq 2$ indistinguishable copies of the teacher choice remain, its cross-entropy is at least $\log r$. Such a step cannot satisfy Eq. (11), whose threshold is at most $\log 2$.

**Selection rule comparison.** We compare SGI's selection rule with the training-form rule using the same frozen DPR checkpoint and candidate pools on PixArt-$\alpha$-DMD. Table 5 shows that the training-form rule with $\gamma = 0.06$ yields higher mean LPIPS and HPS, with comparable Vendi, MSS, and PickScore. The diversity-only rule also increases mean LPIPS but reduces HPS. For a controlled comparison with SGI, Table 1 retains SGI's selection rule in both arms, isolating the effect of replacing its scorers.

### A.3 Pool size at deployment

We vary the candidate pool size in SGI on PixArt-$\alpha$-DMD, fixing the output batch size and each arm's deployed selection weight. Both arms receive the same candidate pool, and the default pool-size row reuses the corresponding weight-sweep configurations. Table 6 shows that DPR achieves higher Vendi and HPS than SGI on the larger pools, with greater gains at the largest evaluated pool size. Both methods return the same images when the pool equals the output batch. The larger-pool

results suggest that considering candidates together becomes more beneficial when selection has more alternatives.

Table 6: **Pool size at deployment.** SGI on PixArt-$\alpha$-DMD with $B$=5 and each arm's deployed $\lambda$ held fixed. $\Delta$ denotes Ours minus SGI, with bold indicating improvement in both criteria. The $M$=30 row reuses the corresponding weight-sweep configurations.

| | SGI (CLIP+DINO) | | Ours | | $\Delta$ | |
|---|---|---|---|---|---|---|
| $M$ | Vendi↑ | HPS↑ | Vendi↑ | HPS↑ | Vendi | HPS |
| 5 | 3.6531 | 28.241 | 3.6531 | 28.241 | +0.0000 | +0.000 |
| 10 | 3.7704 | 28.133 | 3.7682 | 28.181 | -0.0022 | +0.048 |
| 20 | 3.8471 | 28.105 | 3.8358 | 28.100 | -0.0113 | -0.005 |
| 30 | 3.8835 | 28.091 | 3.8934 | 28.123 | **+0.0099** | **+0.032** |
| 64 | 3.9288 | 28.047 | 3.9563 | 28.099 | **+0.0275** | **+0.053** |

### A.4 Controlling for the diversity backbone

We compare SGI scoring variants under matched sampling conditions, candidate pools, and output batch sizes. The PixArt-$\alpha$-DMD backbone-only control replaces DINOv2 with DINOv3 while retaining CLIP. Across generators, we also evaluate HPSv2+DINOv3 and HPSv2-only selection against DPR. Arms that directly select using HPSv2 are marked with ‡. At the evaluated operating points, Table 7 shows that DPR improves diversity over the backbone-only swap and achieves higher Vendi and lower MSS than HPSv2+DINOv3 across backbones. Selection using HPSv2 directly yields higher HPS than DPR, while PickScore generally favors the separate scorers. HPSv2-only selection yields the highest preference scores and lowest diversity.

Table 7: **Controlling for the diversity backbone.** SGI with $M$=30 and $B$=5 under matched sampling conditions. The backbone-only swap is evaluated on PixArt-$\alpha$-DMD. Best per column within each block is bold. ‡Indicates selection using HPSv2 predictions directly. DPR shares the HPD training source with HPSv2, as discussed in Appendix A.8.

| Model | Scorer | Vendi↑ | MSS↓ | HPS↑ | Pick↑ |
|---|---|---|---|---|---|
| PixArt-$\alpha$-DMD | SGI CLIP + DINOv2 | 3.8835 | 0.3876 | 28.0906 | 22.0527 |
| | backbone swap CLIP + DINOv3 | 3.8660 | 0.3908 | 28.1093 | 22.0501 |
| | teacher criteria HPSv2 + DINOv3‡ | 3.8750 | 0.3887 | 28.4892 | 22.1303 |
| | best-of-$N$ HPSv2 only‡ | 3.5791 | 0.4555 | **29.7863** | **22.4322** |
| | **Ours** | **3.8934** | **0.3852** | 28.1227 | 22.0560 |
| SANA-Sprint | SGI CLIP + DINOv2 | 4.3151 | 0.2837 | 29.6823 | 22.6033 |
| | teacher criteria HPSv2 + DINOv3‡ | 4.3262 | 0.2801 | 30.0984 | 22.6699 |
| | best-of-$N$ HPSv2 only‡ | 4.0979 | 0.3409 | **31.4388** | **22.9562** |
| | **Ours** | **4.3314** | **0.2784** | 29.8028 | 22.6081 |
| SDXL-Turbo | SGI CLIP + DINOv2 | 4.4136 | 0.2531 | 28.5465 | 22.4689 |
| | teacher criteria HPSv2 + DINOv3‡ | 4.4012 | 0.2575 | 28.8483 | 22.5549 |
| | best-of-$N$ HPSv2 only‡ | 4.1344 | 0.3280 | **30.2667** | **22.8538** |
| | **Ours** | **4.4364** | **0.2482** | 28.5893 | 22.4835 |
| FLUX.1-schnell | SGI CLIP + DINOv2 | 4.6875 | 0.1689 | 28.8519 | 22.5316 |
| | teacher criteria HPSv2 + DINOv3‡ | 4.6690 | 0.1747 | 29.3629 | 22.5171 |
| | best-of-$N$ HPSv2 only‡ | 4.4097 | 0.2541 | **31.2794** | **22.9426** |
| | **Ours** | **4.6965** | **0.1662** | 28.9338 | 22.5254 |

### A.5 Detailed ranker architecture

Figure 5 details the architecture of $R_\theta$ from image encoding to the two score readouts.

**Candidate token construction.** Each candidate is resized to $224 \times 224$ and normalized with ImageNet statistics before entering the shared feature extractor. The MobileNetV3-Large stem produces a $960 \times 7 \times 7$ feature map. Adaptive average pooling reduces this map to a $4 \times 4$ grid, and a linear

projection maps each cell to a 256-dimensional spatial token. The parallel branch processes the high-pass residual using three stride-two $3 \times 3$ convolutions with channel widths 32, 64, and 64. GELU activations follow the first two convolutions. Global mean pooling produces a 64-dimensional texture descriptor, which is concatenated with the backbone's global mean and maximum features. A separate linear projection maps this concatenation to one 256-dimensional global token.

**Unified pool encoding.** The resulting 17 tokens per candidate receive learned positional embeddings shared across images. These embeddings distinguish spatial cells and the global token without assigning candidate-specific indices. We concatenate all candidate tokens into a sequence of length $17M$ and apply two Transformer blocks. Each block uses pre-layer normalization, eight attention heads, and residual connections around attention and the feed-forward network. The feed-forward network expands the token width from 256 to 1024, applies GELU, and projects back to 256. Cross-candidate interaction occurs within these blocks, while the preceding feature extraction operates independently on each image.

**Shared field and readouts.** After pool encoding, we reshape the sequence by candidate and retain its spatial tokens. Layer normalization and a linear projection map each spatial token to 768 dimensions, followed by cellwise $\ell_2$ normalization. The resulting field supplies both readouts in Eqs. (3) and (4). Diversity averages cosine distances between corresponding cells, while preference applies a learned linear readout to the cell-averaged field. Global tokens participate in attention and can influence the spatial outputs, but do not enter either readout directly. Thus, the high-pass branch contributes through the global token and cross-candidate attention, without introducing a separate preference feature field.

### A.6 Training details

Each loss in Sec. 2.3 is averaged over its candidates, annotated pairs, or rollout steps. We sum these averaged losses with equal weight throughout training. We use $\tau = 0.1$ for selection and $\tau_{\mathrm{q}} = 0.05$ for pairwise preference. The rollout weight follows the uniform distribution over $\gamma \in \{0,\ 0.06\}$. We sample it once per optimizer iteration, share its value across the training pools, and keep it fixed throughout their rollouts. The selection temperature remains fixed throughout training. For the initial choice, the mean distance to the full pool includes the zero diagonal. DINOv3 ViT-B/16 receives $256\times256$ images, producing a $16\times16$ patch-token grid. We average the tokens within each cell of a $4\times4$ grid and $\ell_2$-normalize the pooled features. The student receives the same images at $224\times224$. Spatial and global tokens from all candidates participate in the same self-attention operation. The global token is omitted from the output field and its cellwise distillation target.

An HPD v2 pool enters training when it contains at least four images and one annotated comparison. All 103,514 pools qualify, covering 429,364 images and 644,046 pairwise comparisons, and 97.3% of them hold exactly four images. We sample up to $K{=}10$ candidates without replacement from each annotated prompt pool. For smaller pools, we retain all candidates and resample their images to obtain a fixed training pool size. This preserves the original prompt grouping and human preference annotations. Resampling varies candidate multiplicities without introducing additional distinct images. The teacher quality $q^{\mathrm{T}}$ is the Laplace-smoothed win rate $(\text{wins} + 0.5)/(\text{comparisons} + 1)$, which shrinks sparsely observed candidates toward 0.5. Wins and comparison counts use only observed annotations. The smoothed scores provide within-pool preference estimates from the observed comparisons.

We use AdamW with weight decay 0.01 and a learning rate cosine-annealed from $3\times10^{-4}$. The pretrained stem uses one tenth of this learning rate. Each step processes three pools of ten candidates. Training lasts 60,000 steps, corresponding to approximately 1.7 passes over the pools.

### A.7 Training and selection pseudocode

Algorithm 1 uses unit-normalized fields $F_i = \{f_{i,l}\}_{l=1}^{C}$ and cached teacher fields $T_i$. The selection score $s_j$ follows Eq. (7), including its normalization and empty-set rule. Teacher scores use $T_i$ and smoothed human win rates. Each annotated pair $(i, j) \in \mathcal{H}$ carries the preference label $I_{ij} \in \{+1, -1\}$ defined in Eq. (6). Training settings follow Appendix A.6, and selection settings follow Appendix A.9.

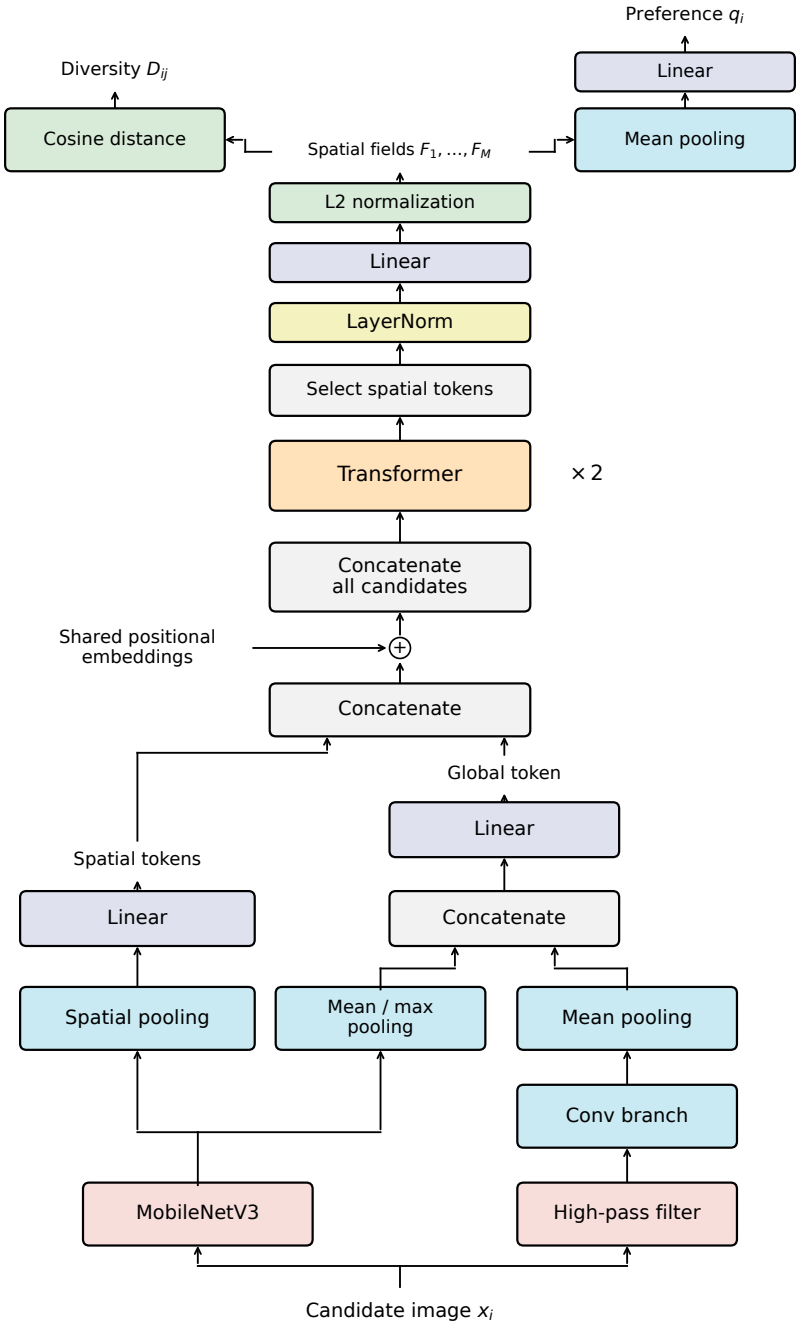



Figure 5: **Detailed block diagram of $R_\theta$.** Read from bottom to top. The shared image encoder produces spatial tokens, while pooled backbone and high-pass features form a global token. After adding shared positional embeddings, two Transformer blocks process all candidate tokens together. Only spatial outputs form the normalized fields used by both readouts. Diversity averages cosine distances over corresponding cells, while preference applies a linear readout to each field's mean.

**Algorithm 1:** DPR training and scorer replacement in SGI

**Input:** Ranker $R_\theta$, annotated image pools $X$, human preference pairs $\mathcal{H}$, spatial targets $T$

**Field and score readouts**

$\{F_i\}_{i=1}^{|X|} \leftarrow R_\theta(X), \quad D(F_i, F_j) \leftarrow 1 - \frac{1}{C}\sum_l f_{i,l}^\top f_{j,l}, \quad q(F_i) \leftarrow w^\top \frac{1}{C}\sum_l f_{i,l}$

**Training**

**for** *each minibatch of annotated pools* $(X, T, \mathcal{H})$ **do**

  $\{F_i\}_{i=1}^{K} \leftarrow R_\theta(X), \quad \gamma \sim \mathrm{Uniform}(\{0, 0.06\}), \quad \mathcal{S}_1 \leftarrow \varnothing$

  **for** $k = 1, \ldots, B$ **do**

    $j_k^\star \leftarrow \arg\max_{j \notin \mathcal{S}_k} s_j^{\mathrm{T}}(\mathcal{S}_k; \gamma), \quad \mathcal{S}_{k+1} \leftarrow \mathcal{S}_k \cup \{j_k^\star\}$

  $\mathcal{L}_{\mathrm{div}} \leftarrow \frac{1}{KC}\sum_{i,l}(1 - f_{i,l}^\top t_{i,l})$

  $\mathcal{L}_{\mathrm{qual}} \leftarrow \frac{1}{|\mathcal{H}|}\sum_{(i,j)\in\mathcal{H}} \mathrm{softplus}\big(-I_{ij}(q(F_i) - q(F_j))/\tau_{\mathrm{q}}\big)$

  $\mathcal{L}_{\mathrm{sel}} \leftarrow -\frac{1}{B}\sum_{k=1}^{B} \log \frac{\exp(s_{j_k^\star}(\mathcal{S}_k;\gamma)/\tau)}{\sum_{j\notin\mathcal{S}_k}\exp(s_j(\mathcal{S}_k;\gamma)/\tau)}$

  $\theta \leftarrow \mathrm{AdamW}\big(\theta, \nabla_\theta \mathbb{E}_{\mathrm{pool}}[\mathcal{L}_{\mathrm{div}} + \mathcal{L}_{\mathrm{qual}} + \mathcal{L}_{\mathrm{sel}}]\big)$

Freeze $\theta$ for selection

**SGI deployment with DPR scoring**

**for** *each pruning stage prescribed by SGI* **do**

  Decode current candidate previews $X$ and obtain SGI's target size $B_{\mathrm{stage}}$

  $\{F_i\}_{i=1}^{|X|} \leftarrow R_\theta(X)$

  $\mathcal{J}_X(\mathcal{S}) \leftarrow \sum_{i\in\mathcal{S}} q(F_i) + \lambda \sum_{i<j\in\mathcal{S}} D(F_i, F_j)$

  $\mathcal{S} \leftarrow \mathrm{SGISelect}(\mathcal{J}_X, B_{\mathrm{stage}})$

  Retain the selected trajectories and their conditioning, and continue SGI sampling

Apply SGI's completion and final output-selection rules

Here $K$ is the training pool size and $B \leq K$ is the rollout length and output batch size. Each pool loss is computed before averaging over the minibatch. For deployment, DPR supplies $q(F_i)$ and $D(F_i, F_j)$ to SGI's subset objective, retaining its pruning schedule, trajectory updates, and final output-selection procedure. SGISelect denotes the same greedy approximation to SGI's quadratic subset objective used in both scorer arms. The objective $\mathcal{J}_X$ uses raw $q(F_i)$ and $D(F_i, F_j)$ with the deployed selection weight from Appendix A.9. For the single-step generator, selection is applied once to the completed candidate pool.

### A.8 Evaluation details

**Metric interpretation.** DPR learns from HPD v2 human annotations, whereas the DivGen baseline directly uses HPSv2 predictions as its optimization reward. DPR and HPSv2 share a training data source, so HPS is not an independent measure of preference generalization. We also report PickScore, a separately trained evaluator, to assess whether preference trends extend beyond HPS. Neither host directly optimizes the diversity evaluation metrics, while CLIPScore also serves as SGI's original quality term.

**Paired uncertainty.** We estimate confidence intervals for the paired scorer differences in Tables 1 and 4. We resample prompt batches using the same indices across scorer arms, grouping identical captions together, and compute percentile intervals from 10,000 bootstrap replicates. These pointwise intervals describe prompt variation for fixed checkpoints and sampling runs, without adjustment for multiple comparisons. They do not measure variation across ranker-training seeds or establish equivalence when an interval includes zero.

The intervals are shown below the DPR means in Tables 1 and 4.

### A.9 Selection deployment details

**Pretrained models and selection baseline.** We use the released PixArt-$\alpha$-DMD, SANA-Sprint, SDXL-Turbo, and FLUX.1-schnell models, keeping their weights frozen. For SGI (Parmar et al., 2026), we retain CLIP similarity and DINO distance as the baseline scoring criteria. We select

$B$=5 outputs from $M$=30 candidates and use $\lambda$=1.5 for the baseline. Ours uses the same candidate pool, with $\lambda$ tuned on a held-out split. We use the same frozen DPR checkpoint across all evaluated backbones. Selection uses raw DPR readouts in Eq. (1), without the pool normalization used during training. For the selection results in Table 1, DPR uses $\lambda$=7.1 on PixArt-$\alpha$-DMD and FLUX.1-schnell, 7.8 on SANA-Sprint, and 2.6 on SDXL-Turbo.

### A.10 Further application: optimization deployment

This section details the optimization application evaluated in Sec. 3.4. We use the same frozen DPR checkpoint as in subset selection. Algorithm 2 uses the same field readouts $q(F_i)$ and $D(F_i, F_j)$ as Algorithm 1 at each optimization step.

**Algorithm 2:** DPR for optimization in DivGen

**Input:** Frozen ranker $R_\theta$, frozen generator $G$, prompt $c$, initial noises $Z$

**for** *each optimization step* **do**

$X \leftarrow G(Z, c), \quad \{F_i\}_{i=1}^{B} \leftarrow R_\theta(X)$

$\mathcal{L}_Z \leftarrow -\frac{w_q}{B} \sum_i q(F_i) - \frac{2w_D}{B(B-1)} \sum_{i<j} D(F_i, F_j) + \mathcal{R}(Z)$

$Z \leftarrow \mathrm{HostUpdate}(Z, \nabla_Z \mathcal{L}_Z)$

Table 8: **Optimization deployment settings.** Configurations for Table 4. Both arms use $B$=5 and the same iteration budget within each backbone. DPR weights are rounded for presentation.

| Backbone | Iterations | Ours $w_q$ | Ours $w_D$ |
|---|---|---|---|
| PixArt-$\alpha$-DMD | 10 | 2.1 | 3420 |
| SANA-Sprint | 10 | 1.9 | 3687 |
| SDXL-Turbo | 15 | 1.9 | 5728 |

**Optimization baseline.** We evaluate DivGen (Harrington et al., 2026) with HPSv2 preference rewards and mean pairwise DINO distance, matching the diversity objective used with DPR. The baseline uses HPSv2 reward weight 15 across backbones and DINO diversity weight 20 on PixArt-$\alpha$-DMD and SDXL-Turbo, and 6 on SANA-Sprint. All comparisons use the first 1,000 evaluation prompts for both arms, with baseline generation seed 90002 on PixArt-$\alpha$-DMD and 42 on SANA-Sprint and SDXL-Turbo. The baseline uses learning rate 6.0. Optimization uses SGD with momentum 0.9, gradient clipping at 0.1, and regularization weight 0.01. We adjust the output set size to $B$=5 and use the iteration budgets in Table 8. Both DivGen variants share the initial noises, generator, sampling schedule, and optimization budget within each backbone.

**DPR deployment.** For DivGen, DPR maximizes $w_q\overline{q}+w_D\overline{D}$, where $\overline{q}$ and $\overline{D}$ denote mean preference and mean pairwise field distance. DPR weights were calibrated from the original DPP–HPSv2 configuration using baseline-to-DPR score standard deviation ratios before optimization. The resulting $w_q$ and $w_D$ remain fixed in all comparisons with the mean pairwise baseline. As in selection, optimization uses raw DPR readouts without the pool normalization used during training. Gradients pass through the frozen ranker and generator to the initial noises, while $\mathcal{R}(Z)$ retains the host noise regularizer.

#### A.10.1 Gradient conflict during optimization

For the representation-sharing variants in Table 2, we measure diversity and preference gradients with respect to the initial noises in DivGen on PixArt-$\alpha$-DMD. We report their mean cosine and the fraction of logged steps with negative cosine. These noise-space measurements complement the shared-feature analysis in Appendix A.12. Table 9 shows a more negative mean cosine and a higher conflict frequency for the unified field than for separate networks. The unified field nevertheless achieves higher diversity and HPS in the optimization ablation (Table 2). These results show that unified scoring can improve optimization outcomes while retaining competing diversity and preference gradients.

Table 9: **Gradient conflict in DivGen.** PixArt-$\alpha$-DMD with the representation-sharing configurations in Table 2. Cosine is measured between diversity and preference gradients with respect to the initial noises. Conflict is the fraction of logged steps with negative cosine.

| Scoring architecture | Mean cosine | Conflict |
|---|---|---|
| (i) Two separate nets (2×3.0M) | $+0.006$ | $48.6\%$ |
| (ii) Shared encoder, decoupled heads | $-0.048$ | $59.1\%$ |
| (iii) DPR (unified field) | $-0.071$ | $59.0\%$ |

### A.11 Complexity

We compare scorer parameter counts, the scoring share of total FLOPs, and peak process GPU memory within each host and generator. We sample memory with `nvidia-smi` at the interval specified in Table 10. The DivGen measurements use the HPSv2–DINO baseline with its DPP objective. Table 10 shows that DPR reduces scorer parameters and the scoring share of FLOPs, and lowers peak GPU memory across all evaluated pairs. Image generation accounts for most computation, especially in SGI. DPR therefore provides a compact scoring alternative for both selection and optimization, reducing scoring overhead within the existing generation procedures.

Table 10: **Cost decomposition.** Scorer share is the percentage of per-prompt FLOPs attributable to scoring. Peak VRAM is the maximum GPU process memory observed with `nvidia-smi`, sampled every $0.5$ seconds and converted from MiB to GiB. Hardware and supported hosts follow Secs. 3.1 and 3.4. DivGen cost measurements use the DPP–HPSv2 baseline configuration. Lower values within each host are bold.

| | | | Scorer share of total FLOPs (%) | | | | Peak VRAM (GiB) | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Host | Scorer | Params | PixArt | SANA | SDXL | FLUX | PixArt | SANA | SDXL | FLUX |
| SGI | CLIP+DINO | 238M | 1.77 | 0.52 | 1.44 | 0.10 | 47.9 | 37.6 | 23.7 | 41.4 |
| | **DPR** | **6.1M** | **0.019** | **0.006** | **0.016** | **0.001** | **47.1** | **36.7** | **22.8** | **38.8** |
| DivGen | HPS+DINO | 719M | 12.83 | 3.10 | 4.66 | n/a | 24.9 | 94.0 | 21.4 | n/a |
| | **DPR** | **6.1M** | **0.030** | **0.007** | **0.010** | n/a | **21.6** | **92.1** | **18.1** | n/a |

We also measure end-to-end runtime within each host on the same GPU, using matched prompts and seeds with $B{=}5$. Selection uses $M{=}30$ and the deployed weights in Appendix A.9. Optimization uses the mean pairwise objective in both arms, with the weights and iteration budgets in Appendix A.10. Each scorer arm runs in a separate process, with two warm-up prompts followed by ten measured prompts. We synchronize CUDA before and after each prompt and include prompt encoding, generation, scoring, selection or noise updates, and image decoding. Timing excludes model loading, disk writes, independent evaluation metrics, and between-prompt cleanup, while retaining the optimization host's reporting of enabled objective terms. Table 11 shows lower mean runtime for Ours across all evaluated pairs, with the largest relative reductions on PixArt-$\alpha$-DMD and SDXL-Turbo. The compact scorer thus reduces execution time in both subset selection and optimization.

### A.12 Gradient alignment in the shared representation

To examine our argument that diversity and preference rely on interrelated visual features, we compare their loss gradients with respect to the shared features. We evaluate the frozen DPR checkpoint on all 400 official HPD v2 test pools, whose prompts are excluded from training. Each pool contains nine or ten distinct images. We form preference pairs from all non-tied individual annotations, retaining repeated comparisons as separate votes. We use the training DINOv3 preprocessing for spatial targets and run the ranker in evaluation mode.

Let $\mathbf{H} = \{h_{i,l}\}_{i,l}$ collect the spatial projection outputs before normalization, with $f_{i,l} = h_{i,l}/\|h_{i,l}\|_2$. We flatten all candidate and cell dimensions within each pool and compute

$$c_H = \frac{\langle \nabla_{\mathbf{H}} \mathcal{L}_{\mathrm{div}}, \nabla_{\mathbf{H}} \mathcal{L}_{\mathrm{qual}} \rangle}{\|\nabla_{\mathbf{H}} \mathcal{L}_{\mathrm{div}}\|_2 \, \|\nabla_{\mathbf{H}} \mathcal{L}_{\mathrm{qual}}\|_2}. \tag{22}$$

Table 11: **End-to-end runtime.** Mean seconds per prompt over ten prompts after two warm-up prompts on the RTX PRO 6000 GPU used in our experiments. Both arms retain the same host and output $B$=5 images. Selection uses $M$=30, while optimization uses matched mean pairwise objectives and iteration budgets. Reduction is relative to the baseline within each host. Lower latency within each pair is bold.

| Host | Backbone | Baseline (s/prompt) | Ours (s/prompt) | Reduction |
|---|---|---|---|---|
| SGI | PixArt-$\alpha$-DMD | 1.609 | **1.475** | 8.4% |
| | SANA-Sprint | 7.263 | **6.969** | 4.1% |
| | SDXL-Turbo | 3.397 | **3.144** | 7.4% |
| | FLUX.1-schnell | 31.575 | **31.248** | 1.0% |
| DivGen | PixArt-$\alpha$-DMD | 14.979 | **11.975** | 20.1% |
| | SANA-Sprint | 43.439 | **41.634** | 4.2% |
| | SDXL-Turbo | 41.866 | **36.923** | 11.8% |

Both losses are minimized, so negative cosine indicates conflicting descent directions. We also report $c_F$, computed directly with respect to the normalized field $\mathbf{F} = \{f_{i,l}\}_{i,l}$.

We evaluate $\tau_{\mathrm{q}} = 0.05$ and include $\tau_{\mathrm{q}} = 0.1$ as a sensitivity check. At the former temperature, we randomly permute candidate identities in the preference pairs while keeping features and spatial targets fixed. We average cosine over 32 permutations per pool, preserving the annotation structure under each permutation. Means weight pools equally, and confidence intervals use 2,000 bootstrap resamples of pools.

Table 12: **Supervision gradient alignment on HPD v2 test pools.** Negative denotes the fraction of pools with negative cosine. Shuffled reports the mean under permuted labels. Paired excess is the within-pool human-minus-shuffled difference.

| Space | $\tau_{\mathrm{q}}$ | Mean cosine | 95% CI | Negative | Shuffled |
|---|---|---|---|---|---|
| $H$ | 0.05 | $+0.0293$ | $[0.0274, 0.0311]$ | 5.25% | +0.0005 |
| $H$ | 0.1 | $+0.0297$ | $[0.0278, 0.0316]$ | 4.50% | – |
| $F$ | 0.05 | $+0.0163$ | $[0.0148, 0.0178]$ | 14.00% | -0.0141 |
| $F$ | 0.1 | $+0.0242$ | $[0.0227, 0.0257]$ | 6.00% | – |
| Paired $H$ excess at $\tau_{\mathrm{q}} = 0.05$: $+0.0287$, 95% CI $[0.0268, 0.0306]$. | | | | | |

Table 12 shows positive mean cosine in both feature spaces at both temperatures. Before normalization, gradients from human comparisons exhibit greater alignment than the shuffled-label control. This relationship is measured within the representation learned under both supervision signals. The excess alignment over shuffled labels provides evidence consistent with our argument that diversity and preference draw on interrelated visual features.

### A.13 Limitations

DPR computes preference scores from visual features without directly encoding the prompt. Its preference readout therefore does not explicitly model the correspondence between a candidate and its conditioning text. A natural extension is to incorporate text features into the shared ranker, enabling prompt-conditioned preference scoring alongside pool-aware diversity estimation. Future work can examine how this extension affects prompt alignment and the diversity–preference trade-off.

### A.14 ADDITIONAL QUALITATIVE RESULTS




Figure 6: **Additional selection results on PixArt-$\alpha$-DMD.** Each prompt is shown with Baseline (vanilla sampling), CADS, CNO, SGI, and Ours, in that order. Each row contains all $B$=5 outputs. SGI and Ours select from the same candidate pool using the configurations in Table 1.




Figure 7: **Additional selection results on SANA-Sprint.** Each prompt is shown with Baseline (vanilla sampling), CADS, CNO, SGI, and Ours, in that order. Each row contains all $B$=5 outputs. SGI and Ours select from the same candidate pool using the configurations in Table 1.




Figure 8: **Additional selection results on SDXL-Turbo.** Each prompt is shown with Baseline (vanilla sampling), CADS, CNO, SGI, and Ours, in that order. Each row contains all $B{=}5$ outputs. SGI and Ours select from the same candidate pool using the configurations in Table 1.

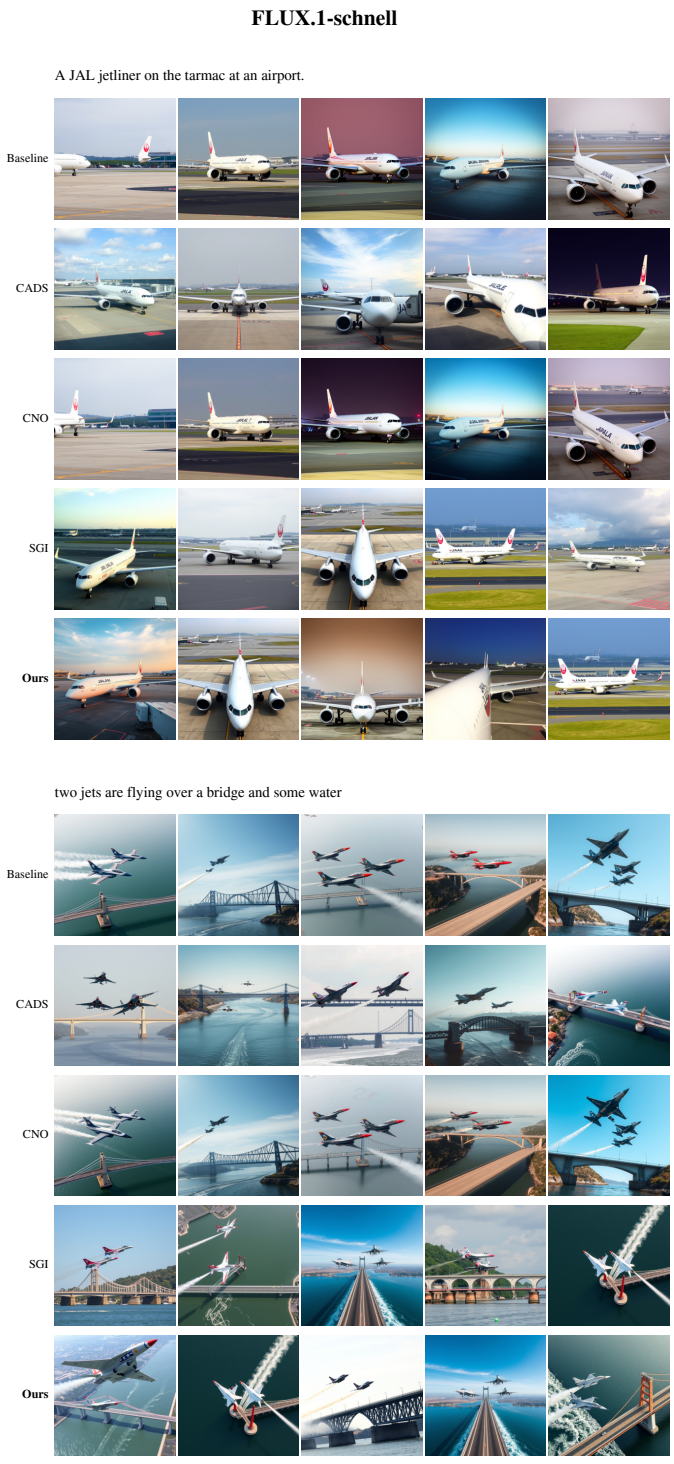



Figure 9: **Additional selection results on FLUX.1-schnell.** Each prompt is shown with Baseline (vanilla sampling), CADS, CNO, SGI, and Ours, in that order. Each row contains all $B$=5 outputs. SGI and Ours select from the same candidate pool using the configurations in Table 1.

### A.15 ADDITIONAL OPTIMIZATION RESULTS

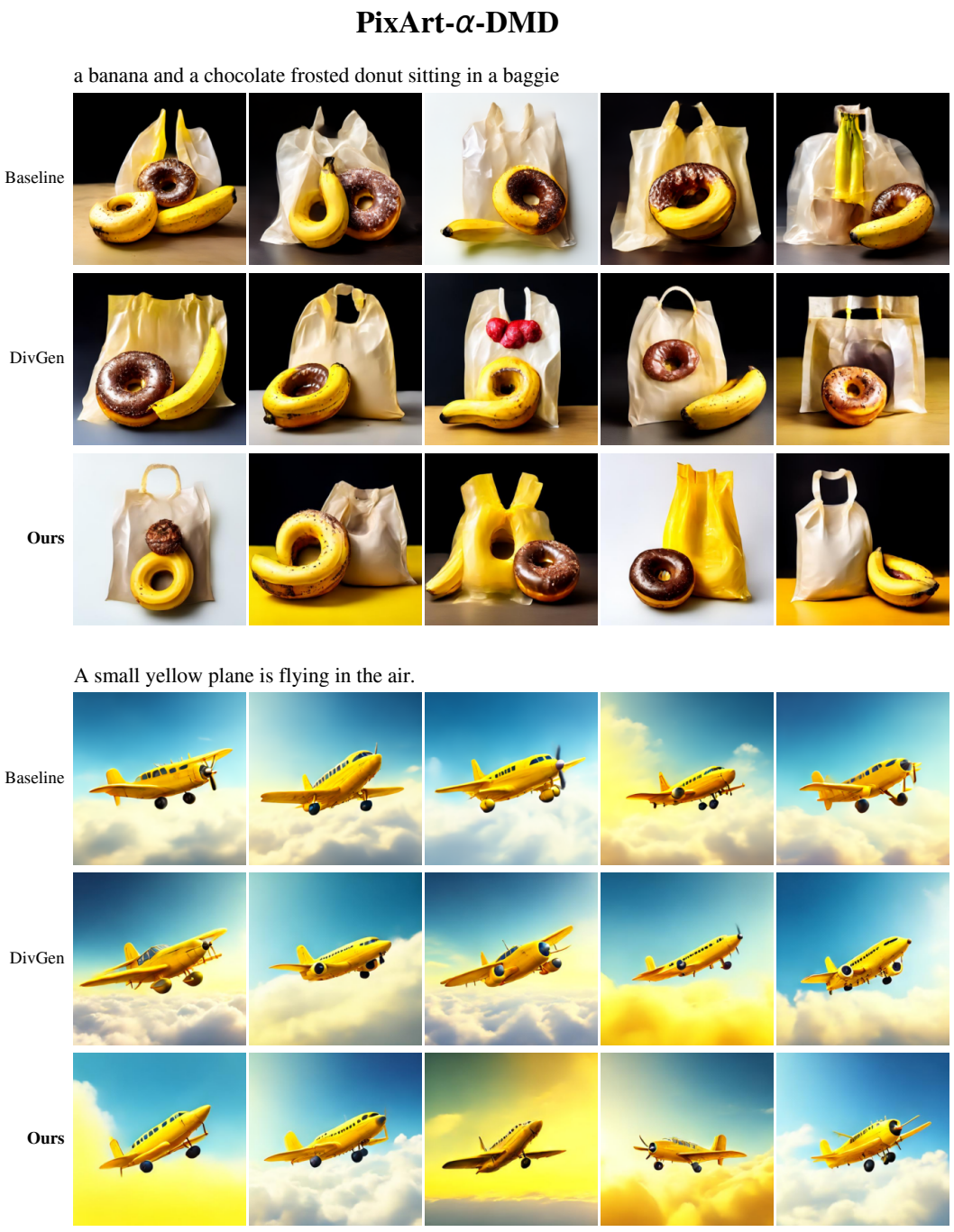



Figure 10: **Additional optimization results on PixArt-$\alpha$-DMD.** Each prompt is shown with Baseline (vanilla sampling), DivGen, and Ours, in that order. Each row contains all $B{=}5$ outputs. DivGen uses HPSv2–DINO scoring, while Ours uses DPR. Both optimization configurations use mean pairwise distance as the diversity objective, following Table 4.




Figure 11: **Additional optimization results on SDXL-Turbo.** Each prompt is shown with Baseline (vanilla sampling), DivGen, and Ours, in that order. Each row contains all $B$=5 outputs. DivGen uses HPSv2–DINO scoring, while Ours uses DPR. Both optimization configurations use mean pairwise distance as the diversity objective, following Table 4.

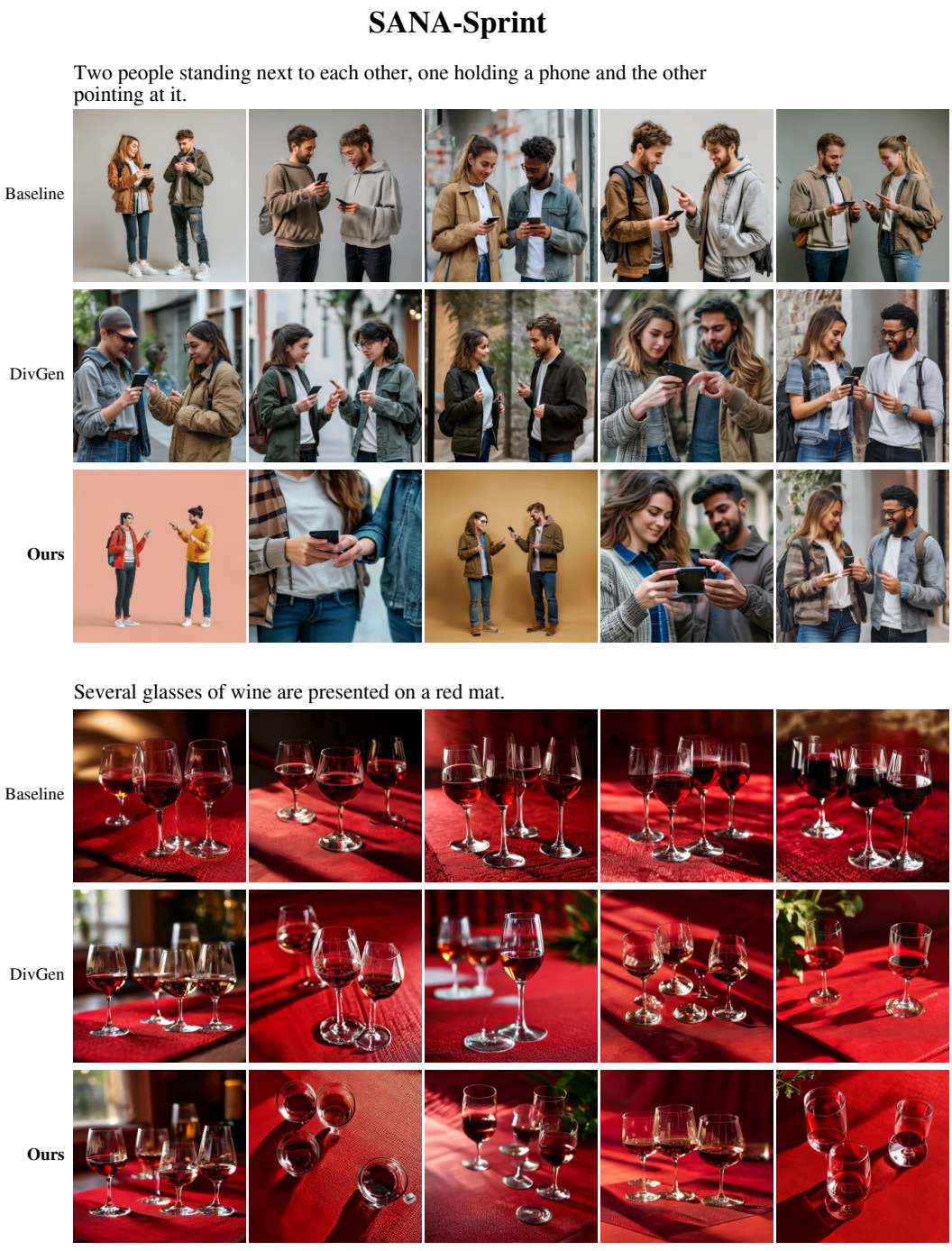



Figure 12: **Additional optimization results on SANA-Sprint.** Each prompt is shown with Baseline (vanilla sampling), DivGen, and Ours, in that order. Each row contains all $B{=}5$ outputs. DivGen uses HPSv2–DINO scoring, while Ours uses DPR. Both optimization configurations use mean pairwise distance as the diversity objective, following Table 4.