BEHAVIOR-ALIGNED TRUST REGIONS FOR SVG REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Reinforcement learning from verifiable rewards (RLVR) for SVG generation typically constrains policy updates in token space. Yet token-space constraints can be poorly aligned with visual behavior: the same drawing may admit many syntactically distinct programs, while a small textual edit can substantially alter the rendering. This mismatch creates opportunities for reward hacking, where the policy exploits shortcuts that preserve a proxy reward while degrading the drawing. We study whether a behavior-aligned representation space can complement token-level constraints in SVG RLVR, and instantiate this space with a variational latent model built on a frozen SVG-finetuned LLM encoder. The Gaussian posterior provides a probabilistic local representation with closed-form divergences and uncertainty estimates, while a lightweight metric adapter calibrates the local geometry using programmatic SVG transformations. We first show that frozen encoder hidden states do not reliably track rendering similarity, motivating the learned metric. We then combine the calibrated behavior representation with a token-KL syntax anchor in a hybrid constraint, using an uncertainty-adaptive local radius. On SVG RLVR, the resulting method reduces the two studied collapse behaviors (viewBox and length collapse) while preserving or improving generation quality, and the representation remains useful on unseen sources and out-of-domain SVGs, while performance is lower on held-out transformation families.

1 INTRODUCTION

Scalable Vector Graphics (SVG) have emerged as a compelling testbed for reinforcement learning from verifiable rewards (RLVR). Unlike free-form text, an SVG program can be *executed* (rendered into pixels), which makes the task self-verifying: a renderer acts as a cheap, deterministic judge of whether a generated graphic is structurally valid and visually faithful to its specification. Recent work exploits this property to post-train large language models (LLMs) on SVG generation with RLVR, rendering sampled programs and rewarding the policy with composite signals over validity, perceptual similarity, and code efficiency (Rodriguez et al., 2025b; Xing et al., 2025a; Wang et al., 2026). CTRL-S (Wang et al., 2026) is a representative framework that unifies multi-task, multi-reward RL training for Text-to-SVG, Image-to-SVG, and SVG code refinement. Building on this foundation, we explore and improve the design of the trust region. To isolate the contribution of the constraint space itself, we compare against both a rendering-space behavior anchor built from perceptual features the reward already computes, and a reward-engineering control that patches the two documented hacking modes directly.

Yet a structural mismatch motivates a closer look at the constraint space. An SVG is a *parametric program*: the mapping from source text to rendered image is many-to-one and highly non-isometric. The same drawing admits infinitely many syntactically distinct programs (paths may be split or merged, coordinates expressed absolutely or relatively, elements reordered), while a single-token edit to a coordinate or the `viewBox` can catastrophically alter the rendering. Standard RLVR, however, constrains policy updates in *token space*: trust regions are formed by PPO-style clipping and token-level KL divergence against a reference policy (Schulman et al., 2017; Shao et al., 2024). Because token-space distance does not track visual behavior, such constraints leave the policy free to wander in directions that corrupt the drawing while keeping the measured distribution close. Prior SVG RLVR systems, including CTRL-S, document exactly this failure: reward-hacking behaviors such as `viewBox` collapse and length collapse, in which the policy shrinks the canvas or degen-

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.

erates the program to inflate a proxy reward (Rodriguez et al., 2025b; Skalse et al., 2022). This suggests that token-space constraints alone may not capture the behavioral changes induced by policy updates.

We therefore investigate a *representation space aligned with SVG rendering behavior*, with the goal that local distances correspond more closely to changes in the rendered drawing than to changes in source syntax. The question is where such a space comes from. A natural first guess, namely the hidden states of a frozen, SVG-finetuned LLM, turns out to fail: we show that these hidden states are *not* isometric to visual similarity, mapping equivalent parameterizations of the same graphic farther apart than single-token edits that destroy it. Raw hidden-state distances are therefore poorly aligned with the behavioral distinctions of interest. We seek a space that is *rendering-aligned* (distance reflects visual rather than purely syntactic change), *locally regular* (a radius is stable enough to calibrate), and *cheap* to evaluate on every rollout without additional rasterization beyond what the reward pipeline already incurs.

We ground the trust region in the *variational latent space* of a VAE built on the frozen SVG-finetuned encoder. This choice is deliberate: unlike a point-estimate embedding, a Gaussian posterior affords closed-form divergences (KL, 2-Wasserstein) and per-sample uncertainty, both of which are directly usable for defining and calibrating a trust region. However, a variational latent space trained for generation does not by itself provide the sharp equivalence invariance that a local behavior constraint benefits from: the Kullback-Leibler regularizer that organizes the manifold also smooths over the very distinctions a constraint must detect. We therefore attach a light *metric adapter*, a lightweight calibration layer over the *frozen* VAE posterior that sharpens equivalence-radius stability while leaving the generative manifold untouched. The adapter is trained with a two-part objective, namely an explicit equivalence-tightening term and a separate negative-repulsion term, using only programmatic transformations of SVG code with rendering verification. The adapter does not optimize the task reward or receive RL reward gradients, so its calibration is performed independently of the policy optimization objective.

On top of this space we construct a *hybrid trust region* that couples a token-KL *syntax anchor*, which guards against format collapse, with a latent-distance *behavior anchor* that bounds how far the rendered drawing may drift from the reference policy, with a radius that adapts to the posterior's per-sample uncertainty. Because the latent metric itself requires no rasterization, enforcing it adds no rendering cost beyond the reward's, and it can be applied on every rollout; a lightweight rendering-based calibration loop runs only periodically to detect whether the policy has begun to exploit the latent metric itself. In our SVG experiments, the resulting design reduces the two studied reward-hacking behaviors without degrading generation quality, and, more broadly, takes a first step toward clarifying when a generative latent representation can serve as a useful local behavior constraint for reinforcement learning over executable, structured code. Program synthesis, molecular design, and other domains where syntax and behavior diverge share this structure and motivate the study; our empirical evidence, however, is limited to SVG.

Contributions.

- We identify and measure the mismatch between token-space constraints and SVG rendering behavior, showing that frozen LLM hidden states do not reliably provide a rendering-aligned metric.
- We build a calibrated behavior representation from a variational latent space over a frozen SVG-finetuned encoder, together with a lightweight metric adapter and uncertainty-aware local radius, while leaving the generative latent model frozen during calibration and RL.
- We contribute a probe-pair benchmark (equivalent-parameterization / near-text-far-rendering / length-multiplied pairs) and a ranking-consistency protocol for evaluating rendering-aligned representations; the benchmark, construction pipeline, and evaluation scripts will be released.

2 Related Work

SVG representation and generation. Early learning-based methods model vector graphics as structured sequences: a VAE over font renderings decoded by an LSTM (Lopes et al., 2019), sketch-RNN for strokes (Ha & Eck, 2018), and the hierarchical DeepSVG (Carlier et al., 2020). Subsequent

work reduces vector supervision via differentiable rasterization (Reddy et al., 2021; Li et al., 2020), fuses image and sequence encoders (Wang & Lian, 2021; Wang et al., 2023), adapts latent diffusion (Xing et al., 2024; Rombach et al., 2022; Oquab et al., 2024), or models continuous geometric parameters (Cao et al., 2023). A second line treats SVG as code (Rodriguez et al., 2025a; Wu et al., 2025; Chen et al., 2025), with geometry-aware or stroke-level tokenizations (Wu et al., 2023; Xing et al., 2025b; Yang et al., 2025; Wang et al., 2025; Tang et al., 2024; van den Oord et al., 2017). Generation quality is usually optimized in token or reconstruction space; whether the resulting representation reflects *rendering behavior* rather than syntax remains unclear, a gap that perceptual metrics such as LPIPS (Zhang et al., 2018), DreamSim (Fu et al., 2023), and DinoScore (Rodriguez et al., 2025a) expose from the evaluation side. We instead probe the hidden states of a frozen SVG-finetuned LLM and learn a representation metric aligned with visual behavior.

Reinforcement learning for verifiable SVG generation. Modern LLM post-training relies on trust-region-style policy updates, from TRPO and PPO (Schulman et al., 2015; 2017) to KL-regularized RLHF (Ouyang et al., 2022) and GRPO (Shao et al., 2024); SVG is a natural verifiable-reward setting because programs can be rendered and scored automatically. RLRF (Rodriguez et al., 2025b) applies online RL with pixel, perceptual, and efficiency rewards; Reason-SVG (Xing et al., 2025a) combines GRPO with structural, semantic, and aesthetic rewards; and CTRL-S (Wang et al., 2026) unifies Text-to-SVG, Image-to-SVG, and code refinement under multi-task, multi-reward GRPO. However, these pipelines constrain updates in token space through clipping or token-level KL, leaving them vulnerable when the syntax-to-rendering map is many-to-one. RLRF indeed reports reward-hacking behaviors such as viewBox and length collapse (Rodriguez et al., 2025b; Skalse et al., 2022). Our approach complements these token-level constraints with a behavior-aligned latent trust region.

Sequence-level trust regions. GRPO-style work targets token-vs-sequence mismatch. GSPO Zheng et al. (2025b) uses sequence-level ratios/clipping; DAPO Yu et al. (2025) decouples clipping and uses token-level loss; CISPO MiniMax et al. (2025) clips importance weights, not updates. Granularity, not geometry: constraints remain in likelihood space, where viewBox collapse can be visually catastrophic despite likelihood-smoothness. Our orthogonal behavior anchor constrains execution behavior, combinable with sequence-level clipping.

Latent-space design and positioning. Representation autoencoders (Zheng et al., 2025a) pair frozen semantic encoders with trained decoders and motivate our use of a frozen SVG encoder; FlowMo highlights the risk of valid but semantically inaccurate reconstructions (Sargent et al., 2025; Lipman et al., 2023; Tschannen et al., 2024; Li et al., 2024); and hyperspherical VAEs show that latent geometry must match the intended similarity structure (Davidson et al., 2018). The closest prior works are RLRF (Rodriguez et al., 2025b), Reason-SVG (Xing et al., 2025a), CTRL-S (Wang et al., 2026), and RAE (Zheng et al., 2025a). We position our work around a complementary question: whether a representation used as an RL constraint can better reflect rendering behavior than source-level proximity.

3 METHOD

Our goal is to construct a local behavior-space constraint for SVG RLVR whose geometry reflects rendering behavior rather than purely source-level change. The method has four components (Fig. 1): a frozen SVG encoder (§3.1), a variational latent space (§3.2), a lightweight metric adapter (§3.3), and a hybrid behavior-space constraint for GRPO (§3.4). Implementation details are deferred to Appendix A.

3.1 FROZEN ENCODER FOR SVG PROGRAMS

We use an SVG-finetuned LLM (Wang et al., 2025) strictly as a *frozen* encoder, extracting and pooling mid-layer hidden states into a single representation h. Freezing is essential: the trust-region metric must remain a fixed ruler while the policy moves. Distances on these raw hidden states rank equivalent parameterizations of one graphic farther apart than single-token edits that destroy it (§4.1), motivating the learned behavior metric used below.

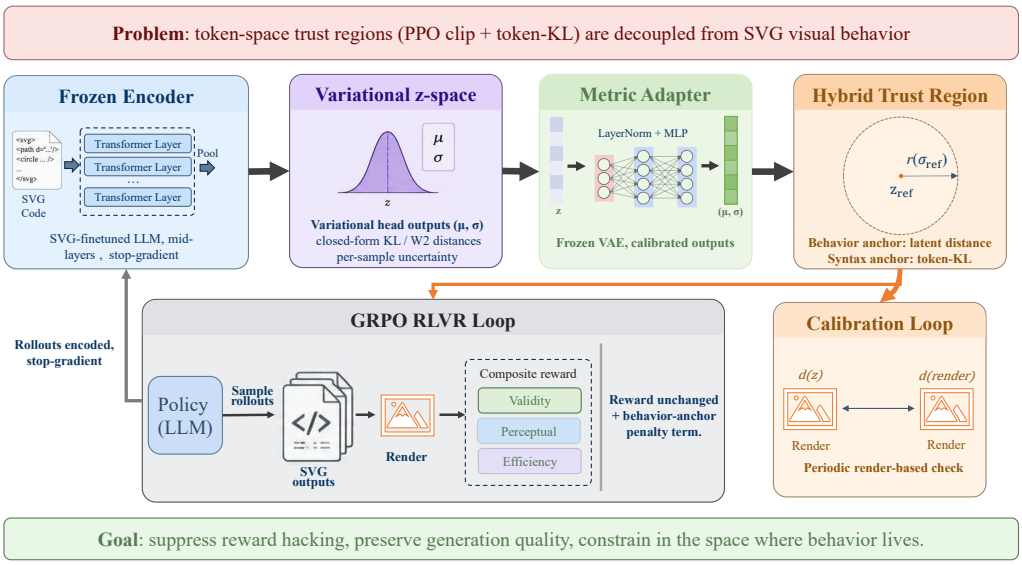

Figure 1: Overview of our method.

3.2 Variational Latent Space over the Frozen Encoder

A variational head maps h to a diagonal Gaussian posterior $q(z \mid h) = \mathcal{N}(\mu(h), \operatorname{diag}\sigma^2(h))$, from which we sample via $z = \mu + \sigma \odot \epsilon$. A light decoder is trained to reconstruct DINOv2 features (Oquab et al., 2024) of the rendering:

$$\mathcal{L}_{\text{VAE}} = \underbrace{\big(1 - \cos(\,\text{dec}(z),\; g(\text{render}(x)))\big)}_{\mathcal{L}_{\text{recon}}} + \beta\, D_{\text{KL}} q(z \mid h)\mathcal{N}(0, I). \tag{1}$$

The perceptual target avoids the degenerate blank-image solution induced by pixel reconstruction on sparse SVG renderings (§4.3). The KL term both organizes a smooth local manifold and yields a Gaussian posterior, providing closed-form divergences and per-sample uncertainty for the trust region.

3.3 Metric Adapter over the Frozen VAE

A generative VAE does not by itself provide the local equivalence invariance useful for a behavior-space constraint: the KL regularizer that smooths the manifold can also blur distinctions that the constraint must detect. We therefore attach a LayerNorm and small MLP to the *frozen* posterior, producing calibrated Gaussian parameters (μ_t, σ_t) and a trust-region representation z_t. This externalizes discriminative calibration while leaving the VAE's reconstruction and neighborhood structure unchanged.

The adapter is trained with a two-part objective,

$$\mathcal{L}_{\text{inv}} = \lambda_e \underbrace{\mathbb{E}_{(a,b)\sim\mathcal{E}}\, d(\mu_t^a, \mu_t^b)^2}_{\text{equivalence MSE-to-zero}} + \underbrace{\mathcal{L}_{\text{InfoNCE}}(\text{far/corrupt negatives})}_{\text{negative repulsion}}, \tag{2}$$

where $\mathcal{E}$ contains label-free equivalent-parameterization pairs constructed programmatically and verified by rendering. Separating equivalence tightening from the InfoNCE denominator preserves a sharp equivalence radius (§4.2), while separating calibration from task-reward optimization reduces a potential Goodhart channel. Distance preservation and uncertainty calibration encourage the calibrated space to retain the local structure of the frozen latent space; behavior-space distances use closed-form 2-Wasserstein or Euclidean distances on μ_t rather than variance-sensitive KL.

3.4 HYBRID TRUST REGION FOR GRPO

We integrate the calibrated latent metric into GRPO (Shao et al., 2024) as a behavior anchor while retaining the standard token-level syntax anchor. We use the term *behavior-space trust region* operationally: the latent distance defines a local radius for sampled behaviors, rather than replacing the policy-level trust region mechanisms of GRPO/PPO. For each rollout, the frozen encoder and adapter produce $q(z_t \mid \cdot)$ in one stop-gradient pass. The task reward is penalized only when latent drift from the reference rollout exceeds a radius:

$$R = R_{\text{task}} - \lambda_z \max\big(0, d(z_t, z_t^{\text{ref}}) - r\big). \tag{3}$$

The reference latent z_{ref} is an exponential moving average of the reference policy's rollout latents, refreshed only in frozen windows of $K = 10$ steps so that the policy cannot chase a slowly moving target, with the frozen initial-SFT latent as fallback. The radius $r_0 = 0.05$ is calibrated as $P95 + 2\sigma$ of the equivalence-pair distance distribution on the rollout set (false-positive rate 0.42% on equivalence pairs; details in §E.4). Eq. 3 is a soft hinge penalty rather than a hard TRPO-style constraint. Unless stated otherwise, Ours (Full) uses λ_z=0.5 and γ=1.0.

The token-KL anchor guards against format collapse, whereas the latent anchor bounds visual drift; this complementarity matters because code-efficiency rewards can be blind to structural distortion and may even favor viewBox collapse. In the uncertainty-adaptive variant, $r_{\text{ref}} = r_0\,(1 + \gamma\,\bar{\sigma}_{\text{ref}})$: widening the radius where the reference itself is uncertain preserves benign exploration that a fixed radius miscalibrates on, and the constraint falls back to a fixed radius whenever the learned σ_t fails its calibration gate (§F.4); a tightening variant is left to future work. Section 4.6 ablates adaptive versus fixed radii. The encoder and adapter remain frozen during RL, with periodic rendering-based calibration checking that latent distances continue to agree with rendering distances.

4 EXPERIMENTS

We evaluate four questions, one per component: **(EQ1)** do frozen hidden states already provide a rendering-aligned SVG metric (§4.1); **(EQ2)** does a learned adapter recover rendering-aligned structure (§4.2); **(EQ3)** does the variational space supply useful local geometry and uncertainty for the behavior constraint (§4.3); **(EQ4)** does the constraint improve RLVR over token-space constraints (§4.4).

Setup. Our primary evaluation uses SVGs sampled from the SFT policy, matching the distribution on which the trust region operates. The probe-pair protocol covers equivalent parameterizations, near-text/far-rendering edits, and length multiplications. We report ranking consistency with rendering distance (Spearman ρ) and equivalent-versus-far AUROC, with held-out transformations and SArena used to assess generalization. Dataset construction and implementation details are deferred to Appendices D and E. The VAE and adapter are trained on SAgogeWang et al. (2025) and evaluated out-of-domain on SArenaWang et al. (2025); the RL experiments use SVG-Sophia Wang et al. (2026). We compare against frozen hidden states, token KL, and DinoScore Rodriguez et al. (2025a).

4.1 EQ1: FROZEN HIDDEN STATES ARE NOT RENDERING-ALIGNED

Across all layer, aggregation, and model configurations on 26,637 probe pairs, frozen hidden-state distances fail to calibrate against rendering distance: equivalent parameterizations are mapped farther apart than single-token edits that destroy the graphic (merged Spearman $\rho < 0$; equivalent-versus-far AUROC below chance in every one of the 33 configurations, only 0.28–0.31 even for the best aggregation per layer, though within the far group alone, ordering retains moderate signal; Table 10). These results indicate that the representation remains sensitive to source parameterization even after SVG fine-tuning, motivating the learned behavior metric.

4.2 EQ2: RECOVERING A METRIC WITH SEPARATED OBJECTIVES

Training the adapter highlights two requirements. First, decoupling equivalence tightening from negative repulsion preserves a sharp equivalence radius; a single shared InfoNCE allows the neg-

ative pool to blunt invariance. Second, programmatically sampled far pairs contain visually near-identical examples, so negatives must be filtered by rendering distance. The resulting discriminative head reaches rollout $AUROC \approx 1.0$ on the evaluated gross-corruption probes, including the stroke-amplify single-token family and structural mutation families. A stratified analysis over corruption magnitude (D.5) shows the learned distance responds monotonically to corruption strength down to the $[0.1\%, 0.5\%)$ pixel-diff bucket, where DinoScore is flat (D.2, R11): the metric remains informative in a regime where the perceptual metric changes only weakly. Discrimination on held-out transformation families is lower (AUROC 0.807, D.3), indicating a clear limitation in transformation-level generalization.

4.3 EQ3: The Variational Latent Space

Perceptual DINOv2 reconstruction is useful in this setting: pixel reconstruction collapses toward the near-blank mean image, whereas the perceptual target yields a smoother local manifold with more useful neighborhoods. For the variational posterior, Euclidean and 2-Wasserstein distances align better with rendering distance than spherical distances. The native variational space already provides useful local discrimination, but its equivalence radius is less stable than that of the calibrated discriminative head, motivating the external metric adapter. Global interpolation remains unreliable; the present use is local, where the constraint is calibrated around the rollout distribution.

4.4 EQ4: Behavior-Space Constraints for GRPO

Main RLVR results. Table 1 reports SVG-Sophia code refinement under similar reward and compute budget. RL post-training improves the SFT model on all quality metrics (DINO $0.871 \rightarrow 0.949$, SSIM $0.633 \rightarrow 0.779$, LPIPS $0.251 \rightarrow 0.163$, success $89.44\% \rightarrow 99.63\%$), and Ours (Full) further improves over CTRL-S by 0.014 DINO, 0.026 SSIM, and 0.023 LPIPS with comparable output length.

Table 1: SVG refinement results on the SVG-Sophia Code Refinement test set. Ours (Full) denotes the complete hybrid behavior-space constraint with an uncertainty-adaptive radius. CTRL-S is evaluated under the same reward and compute budget.

Model	DINO ↑	SSIM ↑	LPIPS ↓	SR	Tokens
Vision-Language Models					
GPT-5.6 Luna	0.921	0.711	0.242	99.30%	988
Claude-Sonnet-4.5	0.796	0.579	0.401	84.58%	790
Gemini-3.1-Pro	0.897	0.697	0.225	92.13%	987
Deepseek-v4-pro	0.891	0.723	0.201	91.24%	890
KIMI-K3	0.922	0.735	0.184	95.45%	1102
CTRL-S	0.951	0.765	0.180	99.79%	866
Ours					
Ours(SFT)	0.871	0.633	0.251	89.44%	2.1k
Ours(SFT+RL)	0.949	0.779	0.163	99.63%	847
Ours(Full)	**0.965**	**0.791**	**0.157**	**99.81%**	891

Table 2 shows the same trend on SArena-Icon (condensed; full comparison in Table 6, Appendix C). Relative to Ours (SFT+RL), Ours (Full) reduces Text-to-SVG FID from 11.425 to 9.368 (9.37 ± 0.28 over 3 seeds, Table 5) and improves CLIP-T2I from 25.181 to 26.103; InternSVG-8B retains a lower FID (8.715) but with weaker semantic alignment, lower success, and much longer outputs. Differences within seed noise (below twice the per-seed standard deviation) are reported as on par. Overall, behavior-aligned latent constraints improve visual fidelity without sacrificing validity.

4.5 Qualitative Results

Figure 2 provides a representative qualitative comparison for Text-to-SVG; the corresponding Image-to-SVG and refinement comparisons are deferred to Appendix G (Figures 6 and 7). Across tasks, Ours better preserves small semantic components, color ordering, spatial relations, and local geometry than the baselines, and in refinement retains the draft's global composition while correct-

Table 2: SVG generation results on SArena-Icon (main comparison; full table in Appendix C, Table 6). SR denotes Success Rate. Ours (SFT), Ours (SFT+RL), and Ours (Full) denote the SFT model, the standard RL baseline, and the complete hybrid latent trust-region method, respectively.

Model	Data	Text-to-SVG					Image-to-SVG				
		FID ↓	CLIP-T2I ↑	CLIP-I2I ↑	SR	Tokens	DINO ↑	SSIM ↑	LPIPS ↓	SR	Tokens
Traditional SVG Methods											
VectorFusion	–	16.594	22.992	70.308	100.0%	33k	–	–	–	–	–
LIVE	–	–	–	–	–	–	0.973	0.986	0.024	100.0%	18k
VTracer	–	–	–	–	–	–	0.966	0.875	0.054	100.0%	4.4k
General-Purpose LLMs											
Gemini-3.1-Pro	–	13.701	26.001	84.426	95.88%	377	0.951	0.744	0.199	97.27%	755
Deepseek-v4-pro	–	12.131	25.882	82.266	97.32%	455	0.951	0.782	0.177	97.72%	884
KIMI-K3	–	11.761	25.851	82.133	98.36%	431	0.964	0.811	0.122	98.17%	893
SVG-LLMs											
LLM4SVG-7B	580K	21.939	19.458	70.726	90.95%	705	0.748	0.472	0.409	95.53%	485
InternSVG-8B	16M	**8.715**	23.916	80.911	97.24%	1.0k	0.949	0.811	0.127	94.45%	1.3k
CTRL-S	145K	11.584	25.944	82.291	99.85%	**346**	0.980	0.835	0.098	**99.93%**	**512**
Ours											
Ours (SFT)	131K	19.213	23.135	77.152	93.08%	1.1k	0.890	0.705	0.190	94.00%	1.1k
Ours (SFT+RL)	145K	11.425	25.181	81.793	99.81%	370	0.962	0.818	0.113	99.83%	508
Ours (Full)	145K	9.368	**26.103**	**82.313**	**99.86%**	362	**0.985**	**0.891**	**0.089**	**99.91%**	534

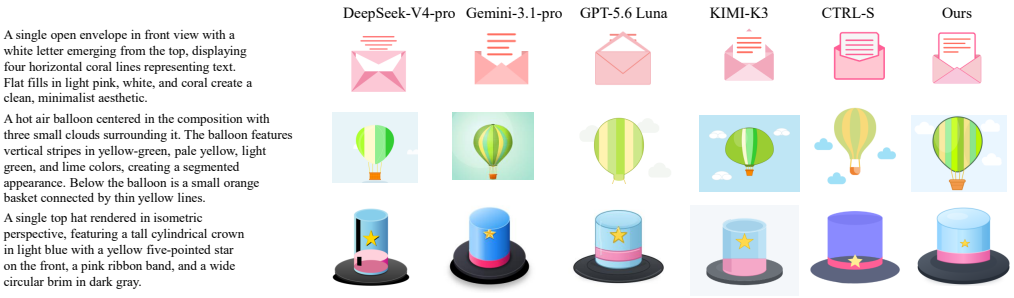

Figure 2: Qualitative comparison for Text-to-SVG.

ing details toward the reference, supporting the claim that the latent anchor constrains visual drift rather than merely code similarity.

4.6 Ablations

Tables 3, 4, and 5 isolate the components of the full method: (i) separated versus shared InfoNCE training, (ii) the distance-preservation regularizer, (iii) the closed-form divergence, (iv) the VAE reconstruction target, (v) adaptive versus fixed trust-region radius, and (vi) the syntax and behavior anchors. The first table reports discrimination and manifold quality; the latter two report collapse incidence, training reward, and unseen-domain success.

Setup. All ablations share the EQ4 conditions: the frozen InternSVG-8B encoder (layers $[15, 16]$, mean-pooled to $h \in \mathbb{R}^{7168}$), GRPO under a fixed composite reward $R = R_{\text{format}}\,(2R_{\text{DINO}} + R_{\text{LCLIP}} + R_{\text{eff}})$ and a fixed compute budget, unless the ablated component is itself varied. Discrimination and manifold metrics (Table 3) are evaluated on the rollout distribution; SR is the unseen-domain Text-to-SVG success rate of Table 2; "Train reward" in Tables 4–5 denotes the task reward R_{task} *excluding* the behavior-anchor penalty, so that all variants are compared on an identical scale (consistent with the reporting convention of §E.4). Collapse rates are computed over $N = 2{,}000$ rollouts per configuration with the automatic detector defined in the caption of Table 5.

Table 3: Adapter and variational-branch ablations on rollout data. Equiv-σ measures equivalence-radius stability; Recon and Neigh. ρ measure manifold quality.

Category	Variant	Discrimination		Manifold quality	
		Rollout AUROC ↑	Equiv-σ ↓	Recon ↓	Neigh. ρ ↑
	Full method (Ours)	0.99	0.01	0.15	0.75
(i) Objective	Shared InfoNCE	0.85	0.10	0.27	0.62
(ii) Distance-preservation	w/o regularizer	0.99	0.04	0.18	0.72
(iii) Closed-form divergence	KL divergence	0.90	0.14	0.22	0.63
	Euclidean on μ	0.98	0.04	0.19	0.71
(iv) Reconstruction target	Pixel reconstruction	0.75	0.20	0.80	0.30

Table 4: Behavior-space radius ablation (3 seeds, mean ± std). R_{task} excludes the anchor penalty.

Variant	VB coll.↓	Len coll.↓	R_{task} ↑	SArena SR↑
Adaptive radius (Ours)	1.0±0.4%	2.0±0.7%	2.40±0.04	99.86±0.03%
Fixed radius	3.1±0.9%	5.8±1.5%	2.15±0.07	99.84±0.04%

Table 5: Behavior-anchor ablation and constraint-space controls (EQ4). Mean ± std over 3 seeds. *Fixed-reward*: reward patched against the two documented exploits (fixed render resolution; symmetric length penalty), token-KL constraint retained. *Render-anchor*: behavior drift penalized in the DINOv2 rendering-feature space the reward already computes. z_{inv}*-only*: discriminative head replaces the variational space. Collapse: fraction of N=2,000 rollouts per run flagged by the automatic detector (viewBox: effective canvas $< 25\%$ of input resolution; length: output $< 50\%$ of ground-truth tokens). Cost: wall-clock per 1k rollouts relative to token-KL only.

Variant	VB coll.↓	Len coll.↓	R_{task} ↑	FID↓	SR↑	Cost
Token-KL only	8.1±1.2%	21.8±2.5%	2.30±0.06	11.43±0.52	99.81±0.03%	1.00×
Fixed-reward + t-KL	1.5±0.6%	2.8±1.0%	2.28±0.03	11.08±0.41	99.80±0.06%	1.00×
Render-anchor	1.8±0.7%	9.5±2.1%	2.33±0.05	10.62±0.44	99.82±0.01%	1.02×
Behavior only	2.0±0.8%	4.1±1.2%	2.25±0.01	10.91±0.38	99.83±0.04%	1.18×
z_{inv}-only	1.2±0.5%	2.4±0.9%	2.36±0.05	9.71±0.35	99.85±0.04%	1.17×
Full (hybrid, adapt.)	**1.0±0.4%**	**2.0±0.7%**	**2.40±0.04**	**9.37±0.28**	99.86±0.03%	1.19×

Controls isolating the constraint space. Two controls establish that the gains come from the latent behavior space rather than from behavioral constraining per se or from reward patching. (i) *Render-anchor*: the behavior anchor is replaced by a drift penalty computed from the DINOv2 rendering features the reward already computes, with its radius calibrated by the same $P95 + 2\sigma$ protocol. (ii) *Fixed-reward + token-KL*: the reward is patched against the two documented exploits (fixed rendering resolution; length reward penalizing under-length outputs), with the standard token-KL constraint retained; both controls share the compute budget and seeds of the main configurations.

Table 5 shows that the render anchor suppresses macroscopic drift (viewBox collapse 1.8% vs. 8.1% for token-KL only) but admits 9.5% length collapse and degraded fine-grained fidelity (FID 10.62 vs. 9.37 for the full method; R_{task} 2.33 vs. 2.40), consistent with the DinoScore blind zone (D.1): the latent anchor's advantage concentrates exactly where the reward's perceptual features are uninformative. Patching the reward reduces the two targeted collapse behaviors (1.5%/2.8%), but produces only a modest change in overall quality (FID 11.08 and R_{task} 2.28 versus the unpatched token-KL baseline). The behavior-space constraint provides a complementary mechanism that does not depend on identifying a particular collapse pattern in advance.

5 Discussion

Behavior-space constraints. A useful constraint can bound changes in behavior in addition to source syntax. For executable outputs, including programs, molecular formulas, formal specifications, and vector graphics, behavior is defined by execution, so token-level KL can be systematically

misleading when the syntax-to-behavior map is many-to-one. Our results support treating constraint-space choice as an explicit design dimension for RLVR: neither raw token distributions nor frozen task-finetuned hidden states are well aligned with the rendering behavior probed here. The empirical evidence is limited to SVG and a single encoder family (§6).

Reward engineering versus constraint-space design. The two exploits we study were previously addressed by patching the reward that enabled them (Rodriguez et al., 2025b). Our control experiment (§4.6) illustrates the distinction between these mechanisms: reward patches target specific anticipated failure modes, whereas the behavior-space constraint operates on local representation drift rather than on a predefined collapse rule. In the experiments here, this distinction is evaluated on the two studied collapse families.

Generation, metrics, and uncertainty. Generation and discrimination impose different demands on a latent space, so our adapter keeps calibration outside the frozen generative space. Perceptual metrics are fallible—DinoScore misses many single-token corruptions that the latent metric tracks—so we separate the adapter objective from task-reward optimization and validate the learned geometry through rendering-based calibration. The variational posterior further provides uncertainty for adaptive radii, giving a probabilistic way to calibrate a local behavior constraint.

6 Limitations

Our metric is most reliable near the policy's rollout distribution and may generalize poorly to distant SVG domains or held-out transformations; accordingly, we view it primarily as a local behavior representation rather than a globally valid SVG perceptual metric. The VAE provides a smooth local manifold but not reliable global interpolation. Our discrimination metrics are anchored to DinoScore-labeled references outside the pixel-diff gates, so AUROC figures partly measure agreement with DinoScore; where DinoScore is blind (99.8% of single-token mutations on SAgoge, §D.2), the metric is evaluated against pixel-diff stratification and human judgment instead (§D.5). Moreover, the VAE reconstruction target and the reward's R_{dino} share the DINOv2 feature family. Thus, the separation is at the optimization supervision level: the adapter receives no task-reward gradient, while feature-level independence is not assumed. Our collapse detectors use fixed thresholds ($25\%/50\%$) whose sensitivity is not exhaustively mapped, and an adversarial policy explicitly optimized against the frozen adapter—together with the detection power of the calibration loop—remains untested; we flag this as the most important robustness gap. Finally, our evidence is limited to SVG with the Qwen2.5/InternSVG encoder family: cross-domain validation and cross-encoder robustness remain future work—the latter a deployment prerequisite, since the encoder shares its lineage with the policy it measures and its representational blind spots are invisible to the constraint.

7 Conclusion

We study whether SVG RLVR benefits from constraining local behavior in a representation space in addition to token space. Our method combines a frozen SVG-finetuned encoder, a variational latent representation with uncertainty, a lightweight metric adapter, and a hybrid token-plus-latent behavior constraint with an adaptive radius. On SVG, the full method reduces the measured viewBox and length collapse rates from 8%/22% to 1%/2% while maintaining or improving generation quality, with the relative contribution of the constraint space examined by our controls (§4.6). The held-out transformation AUROC of 0.807 indicates that the learned geometry is not invariant to all unseen transformations, supporting a local rather than globally uniform interpretation. More broadly, the results suggest that constraint-space design is a useful dimension for RLVR over executable structured outputs.

AI USE STATEMENT

In this work, we used generative AI tools for implementing methods, specifically to assist with code generation, debugging, and refactoring for our experiments. We have not used generative AI tools for generating synthetic datasets, developing theoretical models or conceptual frameworks, formulating or proving mathematical claims, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, interpreting results, cleaning or reformatting datasets, supporting qualitative or thematic data analysis. The remaining required-disclosure tasks are not applicable to this work. Additionally, we used generative AI tools to create or edit software code, to assist with translation and to edit the research paper to improve readability, clarity, and grammar. We did not use generative AI tools to directly generate research ideas, draft the paper's technical content, discover research topics, identify research gaps, or conduct brainstorming. We have reviewed all AI-assisted work. Specifically, all AI-generated or AI-edited code was tested and verified for correctness by the authors; all AI-assisted text was reviewed, revised, and approved by the authors. We take responsibility for the final content of this work, including text, claims, code, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

All authors have read and adhere to the ICLR Code of Ethics. This work studies representation metrics and RL constraints for SVG generation; it involves no human-subjects research, and the human-review loops used for benchmark validation (§D.1, §D.5) only verified programmatically generated SVG pairs and collected no personal data. Our probe-pair benchmark is derived from publicly released SVG corpora (Wang et al., 2025; 2026); we will verify license compatibility and provide documentation before release. Proprietary models were accessed through their official APIs under the providers' terms, solely as evaluation baselines. We identify no harmful applications beyond those of general vector-graphics generation; the method constrains, rather than expands, policy behavior. Compute usage is reported per stage (Table 9); energy was not instrumented.

REPRODUCIBILITY STATEMENT

All probe pairs are generated deterministically (MD5-seeded) with render gates recorded in the build metadata (§D); architectures, training grids, optimizer settings, and per-stage compute are documented in Appendix E; the main RL ablations report mean $\pm$ std over 3 seeds. We include the core code in the supplementary material. The full code will be released after the paper is accepted. The probe-pair benchmark, construction pipeline, and evaluation scripts will be released after publication.

REFERENCES

Defu Cao, Zhaowen Wang, Jose Echevarria, and Yan Liu. SVGformer: Representation learning for continuous vector graphics using transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10093–10102, 2023.

Alexandre Carlier, Martin Danelljan, Alexandre Alahi, and Radu Timofte. DeepSVG: A hierarchical generative network for vector graphics animation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 16351–16361, 2020.

Hanqi Chen, Zhongyin Zhao, Ye Chen, Zhujin Liang, and Bingbing Ni. SVGThinker: Instruction-aligned and reasoning-driven text-to-svg generation. In *Proceedings of the 33rd ACM International Conference on Multimedia (MM)*, pp. 11004–11012, 2025.

Tim R. Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M. Tomczak. Hyperspherical variational auto-encoders. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2018.

Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lin Chai, Richard Zhang, Tali Dekel, and Phillip Isola. DreamSim: Learning new dimensions of human visual similarity using synthetic data. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023.

David Ha and Douglas Eck. A neural representation of sketch drawings. In *International Conference on Learning Representations (ICLR)*, 2018.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. NV-Embed: Improved techniques for training LLMs as generalist embedding models, 2024.

Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Tzu-Mao Li, Michal Lukáč, Michaël Gharbi, and Jonathan Ragan-Kelley. Differentiable vector graphics rasterization for editing and learning. In *ACM SIGGRAPH Asia*, 2020.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

Raphael Gontijo Lopes, David Ha, Douglas Eck, and Jonathon Shlens. A learned representation for scalable vector graphics. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

MiniMax, Aili Chen, A. Li, B. Gong, B. Jiang, et al. MiniMax-M1: Scaling test-time compute efficiently with lightning attention. *arXiv preprint arXiv:2506.13585*, 2025. URL `https://arxiv.org/abs/2506.13585`. CISPO is proposed in this technical report.

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. *Transactions on Machine Learning Research (TMLR)*, 2024.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, 2022.

Pradyumna Reddy, Michaël Gharbi, Michal Lukac, and Niloy J. Mitra. Im2Vec: Synthesizing vector graphics without vector supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Juan A. Rodriguez, Shubham Agarwal, Issam H. Laradji, Pau Rodriguez, David Vazquez, Christopher Pal, and Marco Pedersoli. StarVector: Generating scalable vector graphics code from images and text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025a.

Juan A. Rodriguez, Haotian Zhang, Abhay Puri, Aarash Feizi, Rishav Pramanik, Pascal Wichmann, Arnab Mondal, Mohammad Reza Samsami, Rabiul Awal, Perouz Taslakian, Spandana Gella, Sai Rajeswar, David Vazquez, Christopher Pal, and Marco Pedersoli. Rendering-aware reinforcement learning for vector graphics generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025b.

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 (CVPR)*, pp. 10684–10695, 2022.

Kyle Sargent, Kyle Hsu, Justin Johnson, Li Fei-Fei, and Jiajun Wu. Flow to the mode: Mode-seeking diffusion autoencoders for state-of-the-art image tokenization, 2025.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International Conference on Machine Learning (ICML)*, pp. 1889–1897, 2015.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models, 2024.

Joar Skalse, Nikolaus H. R. Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward hacking. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, 2022.

Zecheng Tang, Chenfei Wu, Zekai Zhang, Mingheng Ni, Shengming Yin, Yu Liu, Zhengyuan Yang, Lijuan Wang, Zicheng Liu, Juntao Li, and Nan Duan. StrokeNUWA: Tokenizing strokes for vector graphic synthesis. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, pp. 47830–47845, 2024.

Michael Tschannen, Cian Eastwood, and Fabian Mentzer. GIVT: Generative infinite-vocabulary transformers, 2024.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Haomin Wang, Jinhui Yin, Qi Wei, Wenguang Zeng, Lixin Gu, Shenglong Ye, Zhangwei Gao, Yaohui Wang, Yanting Zhang, Yuanqi Li, Yanwen Guo, Wenhai Wang, Kai Chen, Yu Qiao, and Hongjie Zhang. InternSVG: Towards unified SVG tasks with multimodal large language models, 2025.

Haomin Wang, Qi Wei, Qianli Ma, Shengyuan Ding, Jinhui Yin, Kai Chen, and Hongjie Zhang. Reliable Reasoning in SVG-LLMs via Multi-task Multi-reward Reinforcement Learning. In Paolo Favaro, Zuzana Kukelova, Atsuto Maki, Anna Rohrbach, Konrad Schindler, and Federico Tombari (eds.), *Computer Vision – ECCV 2026*, pp. 658–675. Springer Nature Switzerland, 2026. ISBN 978-3-032-37271-0. doi: 10.1007/978-3-032-37271-0_36.

Yizhi Wang and Zhouhui Lian. DeepVecFont: Synthesizing high-quality vector fonts via dual-modality learning. *ACM Transactions on Graphics (TOG)*, 40(6):1–15, 2021.

Yuqing Wang, Yizhi Wang, Longhui Yu, Yuesheng Zhu, and Zhouhui Lian. DeepVecFont-v2: Exploiting transformers to synthesize vector fonts with higher quality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18320–18328, 2023.

Ronghuan Wu, Wanchao Su, Kede Ma, and Jing Liao. IconShop: Text-guided vector icon synthesis with autoregressive transformers. *ACM Transactions on Graphics (TOG)*, 42(6):1–14, 2023.

Ronghuan Wu, Wanchao Su, and Jing Liao. Chat2SVG: Vector graphics generation with large language models and image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23690–23700, 2025.

Ximing Xing, Juncheng Hu, Jing Zhang, Dong Xu, and Qian Yu. SVGFusion: Scalable text-to-svg generation via vector space diffusion, 2024.

Ximing Xing, Yandong Guan, Jing Zhang, Dong Xu, and Qian Yu. Reason-SVG: Hybrid reward RL for aha-moments in vector graphics generation, 2025a.

Ximing Xing, Juncheng Hu, Guotao Liang, Jing Zhang, Dong Xu, and Qian Yu. Empowering LLMs to understand and generate complex vector graphics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19487–19497, 2025b.

Yiying Yang, Wei Cheng, Sijin Chen, Xianfang Zeng, Fukun Yin, Jiaxu Zhang, Liao Wang, Gang Yu, Xingjun Ma, and Yu-Gang Jiang. OmniSVG: A unified scalable vector graphics generation model. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, juncai liu, LingJun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Ru Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Yonghui Wu, and Mingxuan Wang. DAPO: An Open-Source LLM Reinforcement Learning System at Scale. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 113222–113244. Curran Associates, Inc., 2025. doi: 10.52202/085713-3775.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders, 2025a.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025b. URL https://arxiv.org/abs/2507.18071.

A Additional Method Details

A.1 Encoder Readout

Our encoder is InternSVG-8B (Qwen2.5-7B base) (Wang et al., 2025). A full stop-gradient forward pass over each SVG program yields token-level hidden states. We read activations from layers $[15, 16]$ and mean-pool them into $h \in \mathbb{R}^{7168}$. Mid-layer states are used because they tend to retain more transferable semantic structure than final-layer states, which are more specialized for next-token prediction (Lee et al., 2024). The encoder is never updated during adapter training or RL, preventing the trust-region ruler from drifting with the policy.

A.2 Variational Training Details

The decoder uses noise-augmented decoding to smooth the discrete support of the frozen encoder. We use a perceptual reconstruction target because SVG renderings are overwhelmingly blank (over 96% white pixels): a pixel-space objective is dominated by the background and collapses toward predicting the mean image. Reconstructing DINOv2 features instead produces a non-degenerate latent space with semantic structure. The KL term controls posterior regularity through the coefficient β in Eq. 1 and makes symmetrized KL, 2-Wasserstein, and per-sample uncertainty available in closed form.

A.3 Adapter Data and Regularization

Equivalent pairs in $\mathcal{E}$ are generated by programmatic SVG transformations, including absolute/relative coordinate conversion, path merge/split, element reordering, and numeric-precision truncation. We also construct syntactically valid but visually suspicious corruptions as negatives. Equivalence is established by construction and verified with pixel-level render-diff; no perceptual reward signal is used for adapter supervision. Unlike a single shared InfoNCE, the decoupled objective prevents a diverse negative pool from raising the loss floor and blunting equivalence sharpness.

A distance-preservation regularizer aligns the adapter's neighborhood geometry with that of the frozen VAE, preventing collapse into a discriminative but semantically scrambled space. Because the adapter outputs a Gaussian distribution, we additionally validate that the calibrated σ_t remains predictive of uncertainty before using it to adapt trust-region radii. We use 2-Wasserstein or Euclidean distance on μ_t for the final constraint, as KL is overly sensitive to variance tails and destabilizes the equivalence radius (§4.2).

A.4 RL Integration and Calibration

During RL, each sampled SVG is encoded with gradients stopped through both the encoder and the adapter. The token-KL syntax anchor remains active to prevent format collapse, while the latent behavior anchor penalizes only drift beyond the current radius. We set the reference radius as $r_{\mathrm{ref}} = r_0\, g(\sigma_{\mathrm{ref}})$, allowing uncertain reference outputs larger local movement. A periodic rendering-based calibration loop compares latent distances with rendering distances and detects whether the policy has begun to exploit the learned metric itself.

B Experimental Protocol

B.1 Encoder and Evaluation Distribution

Unless stated otherwise, the encoder is InternSVG-8B (Wang et al., 2025) (Qwen2.5-7B base) with an SFT-LoRA, frozen throughout. We read layers $[15, 16]$ and mean-pool hidden states to $h \in \mathbb{R}^{7168}$. The primary evaluation distribution consists of rollouts sampled from the SFT policy, since this is the distribution on which the trust region operates. We additionally evaluate held-out transformation families and SArena as out-of-domain references.

B.2 Probe-Pair Construction and Metrics

The probe-pair protocol contains three groups: (i) equivalent-parameterization pairs, where the same graphic is written differently; (ii) near-text/far-rendering pairs, where a single-token edit causes a large visual change; and (iii) length-multiplied pairs, where the same graphic is represented at different compression levels. Representation quality is measured by Spearman ρ between latent and rendering distances and by the AUROC for separating equivalent from far-rendering pairs. For EQ1, we sweep three layer positions, eleven distance aggregations (including last-token, mean, coordinate-weighted, path-segmented, and optimal-transport variants), and both base and SFT models over 26,637 probe pairs.

C Full Quantitative Results

Table 6 reports the complete SArena-Icon results of which Table 2 is a condensed version, adding traditional SVG methods (IconShop, VectorFusion, SVGDreamer, DiffVG, LIVE, VTracer) and further SVG-LLMs (StarVector-8B, LLM4SVG-7B, OmniSVG-3B) to the comparison. Three observations emerge. First, the monotonic improvement from Ours (SFT) to Ours (SFT+RL) to Ours (Full) holds across most metrics on both tasks: relative to the standard RL baseline, the full hybrid trust region reduces Text-to-SVG FID from 11.425 to 9.368 and improves CLIP-T2I from 25.181 to 26.103, and improves Image-to-SVG DINO/SSIM/LPIPS from 0.962/0.818/0.113 to 0.985/0.891/0.089 while retaining a 99.91% success rate. Second, among generative models, Ours (Full) achieves the best CLIP-T2I, DINO, SSIM, and LPIPS scores; InternSVG-8B retains a lower Text-to-SVG FID (8.715) but with weaker semantic alignment (CLIP-T2I 23.916), lower success (97.24%), and much longer

Table 6: Full SVG generation results on SArena-Icon. SR denotes Success Rate. Ours (SFT), Ours (SFT+RL), and Ours (Full) denote the SFT model, the standard RL baseline, and the complete hybrid latent trust-region method, respectively.

Model	Data	Text-to-SVG					Image-to-SVG				
		FID ↓	CLIP-T2I ↑	CLIP-I2I ↑	SR	Tokens	DINO ↑	SSIM ↑	LPIPS ↓	SR	Tokens
Traditional SVG Methods											
IconShop	–	32.288	20.894	70.922	86.51%	1.6k	–	–	–	–	–
VectorFusion	–	16.594	22.992	70.308	100.0%	33k	–	–	–	–	–
SVGDreamer	–	26.612	20.329	69.975	100.0%	132k	–	–	–	–	–
DiffVG	–	–	–	–	–	–	0.869	0.927	0.097	100.0%	17k
LIVE	–	–	–	–	–	–	0.973	0.986	0.024	100.0%	18k
VTracer	–	–	–	–	–	–	0.966	0.875	0.054	100.0%	4.4k
General-Purpose LLMs											
GPT-5.6 Luna	–	13.631	25.771	83.661	97.26%	644	0.941	0.751	0.190	98.71%	505
Claude-Sonnet-4.5	–	16.451	25.322	81.038	99.63%	320	0.910	0.677	0.253	97.95%	705
Gemini-3.1-Pro	–	13.701	26.001	84.426	95.88%	377	0.951	0.744	0.199	97.27%	755
Deepseek-v4-pro	–	12.131	25.882	82.266	97.32%	455	0.951	0.782	0.177	97.72%	884
KIMI-K3	–	11.761	25.851	82.133	98.36%	431	0.964	0.811	0.122	98.17%	893
SVG-LLMs											
StarVector-8B	2.2M	–	–	–	–	–	0.871	0.623	0.206	72.51%	951
LLM4SVG-7B	580K	21.939	19.458	70.726	90.95%	705	0.748	0.472	0.409	95.53%	485
OmniSVG-3B	2M	28.292	21.679	74.831	99.68%	1.8k	0.894	0.756	0.186	99.97%	2.4k
InternSVG-8B	16M	**8.715**	23.916	80.911	97.24%	1.0k	0.949	0.811	0.127	94.45%	1.3k
CTRL-S	145K	11.584	25.944	82.291	99.85%	**346**	0.980	0.835	0.098	**99.93%**	**512**
Ours											
Ours (SFT)	131K	19.213	23.135	77.152	93.08%	1.1k	0.890	0.705	0.190	94.00%	1.1k
Ours (SFT+RL)	145K	11.425	25.181	81.793	99.81%	370	0.962	0.818	0.113	99.83%	508
Ours (Full)	145K	9.368	**26.103**	**82.313**	**99.86%**	362	**0.985**	**0.891**	**0.089**	99.91%	534

outputs (1.0k vs. 362 tokens). Third, optimization-based tracing methods (LIVE, VTracer) attain strong Image-to-SVG reconstruction scores (e.g., LIVE SSIM 0.986) but are not generative models and do not apply to Text-to-SVG, so they serve as reference points rather than direct competitors. Overall, the full results corroborate the main-text finding that behavior-aligned latent constraints improve visual fidelity without sacrificing validity.

Table 7: Proprietary baseline configurations (accessed 2026-06–2026-09).

Model	Version / ID	Temp.	Max tok.
GPT-5.6 Luna	gpt-5.6-luna	0.7	20K
Claude-Sonnet-4.5	claude-sonnet-4.5	-	-
Gemini-3.1-Pro	gemini-3.1-pro	0.7	20K
Deepseek-v4-pro	deepseek-v4-pro	0.7	20K
KIMI-K3	kimi-k3	0.7	20K

D Probe-Pair Protocol and Dataset Construction

D.1 Probe-Pair Construction

All probe pairs are generated *programmatically* from source SVGs drawn from SArena (11.5k graphics across icon-generation, chemistry, illustration, and animation subsets), and are fully deterministic: any randomness is seeded from the MD5 hash of the source SVG content, so a rebuild reproduces identical pairs. Each candidate pair is verified against a rendering gate at construction time; pairs that fail a gate are rejected and never enter the dataset. All thresholds are recorded in the build metadata.

Equivalent-parameterization pairs ($\mathcal{E}$). Twelve transformation families rewrite the *same* graphic without changing its rendering: absolute↔relative coordinate conversion (`abs2rel`, `rel2abs`), path splitting and merging (`path_split`, `path_merge`), numeric-precision truncation (`round2`, `round3`), attribute and element reordering (`attr_reorder`, `elem_reorder`), group wrapping/unwrapping (`group_wrap`, `group_unwrap`), and redundancy stripping/minification (`strip_redundant`, `minify`). Every pair must satisfy a strict render-equivalence gate: $\leq 0.5\%$ pixel difference at 256×256 (a pixel counts as different if $|\Delta| > 2/255$). Two implementation details matter. First, all transformations are pure text-level edits; path `d` strings are parsed and re-serialized by a stdlib-only state machine that handles implicit lineto, S/T reflection relative to the preceding command's absolute control point, and arc flags, and never merges consecutive moveto commands (which would silently change geometry). Second, semantic edge cases—such as `path_split` changing "hole" semantics under the nonzero fill rule, or cairosvg's 0.1–0.5% anti-aliasing artifacts between absolute and relative forms—are intentionally *not* special-cased; they are caught by the pixel gate and the affected pairs are rejected, trading a small yield loss for the guarantee that every admitted pair is render-equivalent.

Near-text-far-rendering pairs ($\mathcal{F}$). Eight single-token mutation families make the smallest possible textual edit with the goal of a large visual change: `viewbox_halve_wh`, `viewbox_shift_min`, `coord_negate`, `coord_scale`, `coord_delete`, `fill_swap`, `segment_drop`, `stroke_amplify`. Admission requires a *dual* gate: $\geq 5\%$ pixel difference *and* token-level edit distance ≤ 3 (whole tags, quoted values, and numbers each count as one token), so the pairs are provably near in text and far in rendering. In addition, a set of *structural* mutation families (whole-shape duplication `all_shapes_duplicate`, global color complement `fill_global_complement`, viewBox zoom-in/out) provides multi-token far pairs for diversity diagnostics; these require $\geq 2\%$ pixel difference, since single-token mutations on complex SAgoge SVGs are overwhelmingly render-invariant (§D.2).

Length-multiplied pairs. Each pair contains the same graphic written at two compression levels (`compressed_expanded`), with a length ratio of at least 2 (observed range 2.2–4.4). The compressed side preserves compact numeric notation where possible so that re-serialization itself does not inflate length.

The full build (seed 42) yields **16,494** equivalence pairs (twelve families), **8,132** far pairs (eight single-token mutations), and **2,011** length pairs, for a total of **26,637** probe pairs with zero gate violations. A human-review loop samples 100 pairs per group (stratified by transform family) into self-contained HTML galleries (side-by side renders, red pixel-difference heatmaps, code diffs); a group is accepted only if label agreement exceeds 95% on 100 sampled pairs (a one-sided 95% Wilson lower bound of $\approx 91.4\%$ on the true agreement rate at the $96/100$ acceptance threshold; we report this interval rather than the point estimate alone)

D.2 Negative Filtering (R10)

Programmatically sampled far pairs are not automatically true visual negatives. We measured this on the 1M-anchor training corpus: all 963,626 far pairs were rendered on both sides and scored with DinoScore. Of the 960,966 pairs with valid features, 64.99% have DinoScore distance < 0.05—i.e., nearly two thirds of the sampled far pairs are visually near-identical noise (mean distance 0.0502, median 0.0376). The full histogram is given in Table 8.

Consequently, far pairs with DinoScore distance < 0.05 are *removed from the negative pool* before adapter training (the cleaned set is kept for diagnostics; a "half-margin weak negative" variant was evaluated in a pilot and dropped). A related measurement (R11) explains the underlying mechanism: on complex SAgoge SVGs, 99.8% of single-token far mutations produce a DinoScore change below 0.05—the standard rendering-based perceptual metric is *blind* to most single-token visual corruptions on this domain (vs. 38% on SArena), and `stroke_amplify` is the only single-token mutation that reliably passes a 2% pixel-diff gate. This motivates both the cleaning step and the structural-mutation far set described above.

DinoScore distance range	count	fraction
$[0.00, 0.05)$	624,517	64.99%
$[0.05, 0.10)$	216,718	22.5%
$[0.10, 0.20)$	96,468	10.0%
$[0.20, 0.40)$	23,198	2.4%
$[0.40, 1.00)$	65	$< 0.01\%$
valid pairs	960,966	100%

Table 8: R10 diagnostic: distribution of DinoScore distances over the $963{,}626$ sampled far pairs of the 1M-anchor corpus before cleaning. 64.99% of pairs fall below the 0.05 noise threshold and are removed from the negative pool before any training.

D.3 Held-Out Protocol

Generalization is measured along four independent dimensions: *transform family* $\times$ *corruption family* $\times$ *SVG domain* $\times$ *source sample*.

- **Transform family.** Six equivalence families are held out from all training: `viewbox_reformat`, `path_simplify`, `style_inline`, `redundant_attr_inject`, `tree_rotation`, `multi_space_separator`. (Note that `viewbox_reformat` only rewrites the *notation* of the viewBox; an actual rescale is not render-invariant under the fixed-canvas rendering protocol and is rejected by the pixel gate.)
- **Corruption family.** Two of the six corruption families (`out_of_viewbox_coords`, `pathological_path_values`) are held out; the other four (`repeat_elem`, `zero_area_path`, `display_none_inject`, `attr_shuffle`) train as *weak positives* (render-invariant "grey-zone" mutations), never as negatives.
- **Domain.** Training anchors are sampled from SAgoge; the out-of-domain axis uses SArena anchors, which share no source with the training set.
- **Source sample.** 5,000 anchors of the 1M corpus are withheld from training (unseen SVGs), yielding $9{,}716$ held-out pairs.

With the best metric head (R10-cleaned far labels, DinoScore as the far labeler), AUROC on the four axes is: train 0.955 / unseen-source 0.950 / out-of-domain (SArena) 0.921 / held-out-transformation 0.807. The held-out-transformation gap is reported as a limitation in the main text; the rollout distribution—the distribution on which the trust region actually operates—reaches AUROC 1.000 (§F.1).

D.4 Metrics

Let $d_{\text{rep}}(a, b)$ be the distance induced by a candidate representation, and $d_{\text{gold}}(a, b)$ the reference rendering distance.

- **Gold rendering distance.** DinoScore: $d_{\text{gold}} = \big(1 - \cos(g(\text{render}(a)),\, g(\text{render}(b)))\big)/2 \in [0, 1]$ with DINOv2-base mean-pooled features g, numerically matched to the InternSVG DINOscore implementation (cross-checked to $< 10^{-3}$).
- **Spearman ρ.** Rank correlation between d_{rep} and d_{gold} over the merged probe set (primary), plus within-group ρ on the far group (where gold distances have variance; equivalence and length groups have near-constant gold distances by construction, so their within-group ρ is noise).
- **AUROC.** Separability of the equivalence pair distribution from the far pair distribution by d_{rep}; chance is 0.5.
- **Equivalence radius stability.** Mean and standard deviation of d_{rep} over equivalence pairs—the quantity a trust-region radius must be calibrated against (the smaller and tighter, the more semantically stable the radius).
- **CKNNA.** Centered-kernel k-NN alignment: the fraction of k-nearest-neighbor sets in the representation space that agree with the k-nearest neighbors in the DINO reference space, measuring neighborhood semantic preservation.

- **Pixel-diff gate and ρ(pixel-diff).** Raw rendering difference (fraction of pixels with $|\Delta| > 2/255$ at 256×256) serves both as the construction gate and as an independent ranking reference for probing whether a representation tracks true pixel change.

D.5 Rollout Evaluation Set

The primary evaluation distribution is the *rollout distribution*—outputs sampled from the SFT policy on SFT-domain tasks (text2svg and svg-edit prompts, temperature 0.7 and 1.0 half-and-half), because this is the distribution on which the trust region operates. From **5,000** sampled rollouts (of which **1,000** are withheld as test anchors), each rollout derives one equivalence pair (from four training-visible and two held-out transform families, verified $\leq 0.5\%$ pixel difference) and one far pair (a single-token visual perturbation, verified $> 2\%$ pixel difference), yielding **7,664** pairs (3,831 equivalence / 3,833 far). All pairs are labeled offline with DinoScore. The set is split by source task: 20% of rollouts are held out for final testing, and of the remaining 80% a further 20% is reserved for constructing the distillation triplets of the ranking head (non-overlapping with the test split). A separate *diverse structural far* set of **17,849** pairs spans four families (`all_shapes_duplicate` 3,489, `fill_global_complement` 4,390, `viewbox_zoom_in` 4,995, `viewbox_zoom_out` 4,975) and is used for far-diversity diagnostics; the main rollout AUROC uses the single-token `stroke_amplify` far pairs.

Corruption-magnitude stratification. To probe sensitivity below the 2% pixel-diff admission gate, we bucket all eight single-token corruption families by realized pixel difference ([0.1%,0.5%), [0.5%,1%), [1%,2%), $\geq$2%) and report Spearman ρ between the latent distance and pixel difference within each bucket, alongside DinoScore's response as the reference. A stratified human study (500 pairs, two annotators, agreement 98.4%) provides a DinoScore-independent anchor in the two lowest buckets.

E Implementation and Reproducibility

E.1 Encoder and Pooling

The encoder is a frozen InternSVG-8B (Qwen2.5-7B base, 28 layers, hidden size 3584) with an SFT, used strictly as a feature extractor. A stop-gradient forward pass runs over the text-only `language_model` (no vision tower, no `lm_head`). Hidden states at layers $[15, 16]$ are concatenated along the feature axis to $h \in \mathbb{R}^{7168}$ and mean-pooled over real (non-padding) tokens. Because the pooled vector uses only layers ≤ 16, the forward pass is *early-stopped* after the first 17 transformer blocks (verified on a CPU mock that the selected-layer outputs match the full model exactly), reducing per-rollout cost substantially. Encoding runs in bf16; sequences are truncated at 16,384 tokens (covering 98.9% of the corpus), and batches are packed by token budget (12k tokens per sub-batch) to fit 46GB GPUs. Pooled vectors that are non-finite are dropped (120 of 3,776,572 attempted encodings, 0.0032%). Throughput is $\sim$5–8k tokens/s/GPU on an L40S. The pooled representations of the corpus are cached once ($\sim$3.78M vectors) and reused by all downstream training stages.

E.2 Variational Branch

The variational branch shares the frozen trunk of the metric head (Linear $7168\to1024$ + GELU, depth 2). On the trunk output $h_{\text{trunk}} \in \mathbb{R}^{1024}$, a `VarEncoder` maps to the posterior parameters: Linear($1024\to512$)+GELU $\to$ Linear($512\to d_z$) separately for μ and $\log\sigma$, with the reparameterization $z = \mu + \sigma \odot \epsilon$. The decoder reconstructs the *perceptual* target: Linear($d_z\to512$)+GELU $\to$ Linear($512\to768$) predicting the ℓ_2-normalized DINOv2 feature of the rendering, trained with $\mathcal{L} = (1 - \cos(\hat{f}, f_{\text{gt}})) + \beta\, D_{\text{KL}} q(z|h)\mathcal{N}(0, I)$. The perceptual target is mandatory rather than a design choice: a pixel-space decoder (128×128 RGB) collapses to the near-blank mean image—96.5% of pixels are white, and the trained decoder produces a dark-pixel fraction of 0 (ground truth 0.031) and a pixel standard deviation of 0.035 (ground truth 0.148)—while perceptual reconstruction yields non-degenerate structure. During training, Gaussian noise $z \leftarrow z + \delta\,\eta$ ($\delta \in \{0, 0.5\}$) is added to smooth the discrete support of the frozen encoder for out-of-distribution rollouts.

The training grid sweeps $d_z \in \{64, 128, 256\}$, $\beta \in \{10^{-3}, 10^{-2}, 10^{-1}\}$, $\delta \in \{0, 0.5\}$, on a *random* sample of 500k SAgoge SVGs (random rather than sequential sampling is essential: sequential sampling is dominated by a single chemical-formula domain whose renderings have pairwise DINO similarity 0.92, collapsing the decoder; random sampling reduces this to 0.60). Optimizer: AdamW, lr 10^{-3}, weight decay 10^{-5}, batch size 512, 30 epochs, bf16 autocast, gradient clipping 5.0, seed 0, one GPU per configuration ($<$ 2M trainable parameters). The selected configuration is d_z=128, β=10^{-2}, δ=0.5.

E.3 METRIC ADAPTER

Two adapter instantiations appear in this work; both use the same decoupled objective.

Adapter over frozen hidden states (route B). A dual-head module: trunk Linear(7168→1024)+GELU (depth $\in \{1, 2\}$), then two ℓ_2-normalized heads Linear(1024→256)—z_{inv} (the trust-region head, which never observes any visual signal) and z_{rank} (a verification head distilled from DinoScore order, never used inside RL). $\leq$ 9M parameters at depth 1. The decoupled loss separates equivalence-tightening from negative-repulsion: $\mathcal{L}_{\text{equiv}} = \text{mean}_i\, d(a_i, p_i)^2$ (MSE-to-zero on the spherical distance $d = (1 - \cos)/2$; its gradient grows linearly with distance, compressing both the mean *and the variance* of the equivalence radius—a hinge would only bound the mean), and $\mathcal{L}_{\text{push}}$, a supervised InfoNCE in which the anchor's equivalence partner (T=0.07) and corruption partner (T=0.12, weak positive) are the positives and all batch far pairs are negatives (64 anchors $\times$ 3 partners = 192 negative candidates). Total $\mathcal{L}_{\text{inv}} = \lambda_e \mathcal{L}_{\text{equiv}} + \mathcal{L}_{\text{push}}$ with $\lambda_e \in \{2, 5\}$; z_{rank} adds margin-ranking (m=0.1, λ_{rank}=0.5) and distance calibration (λ_{cal}=0.1) against DinoScore. Equivalence partners never enter the InfoNCE denominator—the decoupling is the point. Optimizer: AdamW, lr 10^{-4} (cosine to 10^{-6}), weight decay 0.01, gradient clipping 1.0, 20k steps, batch 64 anchors.

Training data: 1M anchors (995k train / 5k withheld) from SAgoge, 2,140,659 pairs (804,243 equivalence / 999,967 corruption / 336,449 far, the far set R10-cleaned as in §D.2), and 9,716 held-out pairs for the four-axis protocol.

Adapter over the frozen VAE posterior (E-VAE-TR2, path 4). A LayerNorm followed by a small MLP (128→256→256) maps the frozen posterior mean to calibrated Gaussian parameters (μ_t, σ_t) (sub-route (b): the output *remains a distribution*, preserving closed-form KL/Wasserstein divergences and per-sample uncertainty; the point-estimate sub-route (a) would forfeit both). The VAE is frozen throughout, so reconstruction, interpolation, and neighborhood structure are untouched by calibration. The loss is $\mathcal{L}_{\text{adapter}} = \lambda_e\, \text{MSE}(d(\mu_t^A, \mu_t^B)^2 \to 0) + \mathcal{L}_{\text{InfoNCE}}(\text{far/corrupt negatives}, T{=}0.07) + \lambda_{\text{distill}} \mathcal{L}_{\text{distill}}$, where $\mathcal{L}_{\text{distill}} = \text{MSE}(\cos\text{-dist}(z_t), \cos\text{-dist}(z_{\text{VAE}}))$ keeps the calibrated space faithful to the frozen VAE's neighborhood geometry ($\lambda_{\text{distill}} \in \{0, 0.1, 0.5\}$; λ_{distill}=0 risks a discriminative collapse that crushes equivalence pairs into a point and scrambles intra-class structure). Training: $\leq$ 0.2M parameters, AdamW lr 10^{-4}, 15k steps, cleaned far pairs; supervised by the same programmatic transformations with render-diff verification, i.e., zero visual signal in the labels. Trust-region distances are computed with W2 or Euclidean on μ_t; KL is kept for offline diagnostics only, since its variance-tail sensitivity destabilizes the equivalence radius (§F.3).

E.4 RLVR INTEGRATION

Training uses VERL 0.8.0 GRPO (Ray + FSDP2 + vLLM rollout) initialized from the SFT model. GRPO configuration: group size $G = 8$, AdamW lr 10^{-6}, token-KL loss coefficient 0.03 (the RLRF default) against the reference policy, PPO clip ratio 0.2, one PPO epoch, advantage normalization by standard deviation, bf16 throughout; rollout temperature 0.7 (greedy validation), max prompt/response length 4096 tokens.

Reward (unchanged across all trust-region configurations). $R_{\text{total}} = R_{\text{format}} \cdot \big(2\, R_{\text{dino}} + R_{\text{lclip}} + R_{\text{eff}}\big)$, where $R_{\text{format}} \in \{0, 1\}$ is a binary gate (exactly one `<think>` block plus a CairoSVG-renderable SVG block), $R_{\text{dino}} = 0.5\,(1 + \cos(g(\text{gen}), g(\text{gt})))$ with DINOv2 features of the rendered images, $R_{\text{lclip}} = \max(0, \cos(e_{\text{img}}, e_{\text{text}}))$ with LongCLIP features (the

Stage	work	budget
Probe build	3,000 sources, 8 workers	$\sim$2 min
E1 sweep	26,637 pairs $\times$ 3 layer configs $\times$ 11 metrics	$\sim$12 GPU-h
R10 cleaning	963k far pairs, 1.93M renders, 32 workers	6–8 h CPU
1M-corpus pooling	3.78M SVGs, 8 GPUs (200k / $\sim$0.5 h)	$\sim$10 GPU-h
Metric-head grid	8 configs $\times$ 3–4 h, 1 GPU each	1–1.5 GPU-day
z-VAE grid	1 GPU-day per config, $\sim$9 configs + E3	$\sim$2 GPU-weeks
E-VAE-TR	zero-training measurement + dual eval	$\sim$1.5 days
E-VAE-TR2	adapter grid (6 cfg $<$ 1 GPU-h) + ablation paths	$\sim$2 GPU-days
RLVR, one config	GRPO $\sim$50k steps, 8$\times$L40S	6–8 days
RLVR, full matrix	5 main configs + ablations	3–4 GPU-weeks

Table 9: Per-stage compute. Rollout encoding is the dominant recurring cost of the RL runs; the 17-layer early-stopped encoder keeps one stop-gradient pass per rollout at $\sim$8GB footprint amortized across reward workers.

extended-context tokenizer handles instructions beyond CLIP's 77-token limit), and $R_{\text{eff}} = 1 - \big(\max(0,\, L_{\text{gen}} - L_{\text{gt}}/2)/L_{\text{gt}}\big)^2$ penalizing code length quadratically beyond half the ground-truth length. All reported "train reward" curves and table entries use R_{task} without the anchor penalty, unless explicitly stated.

Behavior anchor. The frozen encoder + adapter produce a latent in a single stop-gradient pass per rollout; the anchor adds $R_{\text{behavior}} = -\lambda_z \max(0,\, d(z, z_{\text{ref}}) - r)$ with euclidean distance and $\lambda_z \in \{0.1, 0.5, 1.0\}$; the main configurations use λ_z=0.5. The radius $r = 0.05$ is calibrated on the equivalence-distance distribution of the rollout set: $P_{95} = 0.029$, $P_{99} = 0.042$, $P_{95} + 2\sigma = 0.0476$; at $\varepsilon = 0.05$ the false-positive rate (equivalence pairs flagged as out-of-radius) is 0.42% and the false-negative rate (far pairs admitted) is 0.0%. The reference latent z_{ref} uses a dual anchor: an EMA of the reference policy's rollouts, updated only in frozen windows of K steps (so the policy cannot exploit a slowly moving window), with a frozen initial-SFT latent as fallback. In the uncertainty-adaptive variant, the radius scales with the reference posterior's uncertainty: $r_{\text{ref}} = r_0 \cdot \big(1 + \gamma\, \overline{\sigma}_{\text{ref}}\big)$, $\gamma \in \{0.5, 1.0, 2.0\}$ (main configurations: γ=1.0)—widening the region where the reference policy is itself unconfident, a mechanism unavailable to point-estimate metrics.

Anti-coupling and calibration. The encoder and adapter are frozen and stop-grad throughout RL, and the latent enters only the constraint, never the reward. Independence is enforced at the *supervision* level: the adapter is trained on label-free programmatic transformations and never observes the reward; the VAE reconstruction target and R_{dino} do share the DINOv2 feature family, so feature-level blind spots may still be shared (§6). A periodic rendering-based calibration loop re-measures the Spearman ρ between latent distances and DINO distances on a fresh sample; sustained degradation triggers an alarm, detecting when the policy begins to exploit the latent metric itself (Goodhart on the constraint).

E.5 Computational Budget

Primary hardware: 8$\times$L40S (46GB); the E-VAE-TR stages ran on 8$\times$3090 (24GB). Per-stage budgets (GPU-hours, from the execution plans; energy was not instrumented):

F Additional Results

Stage names (E1, E2, E-VAE-TR, E-VAE-TR2) denote successive experiment series; R10/R11 denote diagnostic measurements referenced from §D.

F.1 Full Metric-Head Grid

E1: the full hidden-state sweep. Three layer positions ($[11, 12]$, $[15, 16]$, $[19, 20]$) $\times$ eleven distance aggregations (four pooling schemes $\times$ cosine, euclidean plus position-aligned, Sinkhorn-OT,

Layers	best merged ρ	best far-group ρ	best AUROC(far vs equiv)
$[11, 12]$	-0.061	$+0.570$	0.292
$[15, 16]$	-0.055	$+0.549$	0.298
$[19, 20]$	-0.095	$+0.550$	0.282

Table 10: E1 full-sweep summary on 26,637 probe pairs (best of 11 aggregations per layer position, base model). Far-group ordering carries moderate signal; inter-group calibration is inverted (AUROC < 0.5).

Metric	single-token far	diverse structural far
AUROC(equiv vs far), rollout	1.000	1.000
equiv distance, mean / σ	0.0091 / 0.0093	0.0091 / 0.0093
far distance, mean	0.255	0.631
$\rho(z_{\text{inv}}$, pixel-diff)	0.790	0.732
AUROC, train (DinoScore-labeled far)	0.955	
AUROC, unseen source	0.950	
AUROC, out-of-domain (SArena)	0.921	
AUROC, held-out transform families	0.807	
length $\lvert\rho\rvert$ (length neutrality)	0.019	
random-negative mean distance	0.992	

Table 11: E2 best head (decoupled loss, λ_e=2, depth 2, mean pooling, 20k steps). The rollout-distribution AUROC is perfect against both single-token and structural far families, and the head tracks raw pixel change ($\rho = 0.79$) where DinoScore is blind (§D.2).

and path-segment-set-OT variants) on the base and SFT models. Table 10 reports the best configuration per layer; every one of the 33 configurations yields a negative merged Spearman ρ and an equiv-vs-far AUROC below chance. The SFT-merged model is statistically indistinguishable (merged ρ -0.034 to -0.070; far-group ρ $+0.555$ to $+0.571$; AUROC 0.290–0.308), i.e., SVG fine-tuning does not change mid-layer geometry. A representative pair illustrates the inversion: a `coord_negate` far pair (visual catastrophe) has hidden distance 0.001, while an `abs2rel` equivalence rewrite has distance 0.062—a 60× inversion.

E2: the decoupled-loss grid. The 8-configuration grid ($\lambda_e \in \{2, 5\}\times$ trunk depth $\in \{1, 2\}\times$ pooling $\in$ {mean, coord-weighted}) is selected on rollout AUROC and equivalence-radius tightness; the best configuration is λ_e=2, depth 2, mean pooling. Its full evaluation is in Table 11. Decoupling is the decisive ingredient: the single-InfoNCE head (v2) reaches 0.797 source AUROC, and the decoupled head 0.950, while compressing the equivalence distance to mean 0.0091, $\sigma = 0.0093$ on the rollout distribution. The ranking head z_{rank} collapses toward z_{inv} ($\rho_{\text{far}} = 0.477$), so the trust region uses z_{inv} only; the distance-calibration stream is reported as a failed auxiliary.

F.2 Variational Space Geometry

Metric geometry depends on representation type. On the variational latent space ($z = \mu$, 5,000 samples), Spearman ρ of latent distances against DINO reference distances is 0.662 for the spherical (cosine) metric and 0.759 for Euclidean—the reverse of the discriminative head, where ℓ_2-normalization makes the sphere the design default. The Gaussian latent, pulled toward $\mathcal{N}(0, I)$ by the KL term, carries distance-from-prior information in $\|\mu\|$, which the spherical metric discards (Fig. 4).

The β trade-off. Table 12 shows the KL-regularization frontier: β=10^{-3} yields the best reconstruction (0.893) but a collapsed posterior (KL 14.8—no manifold); β=10^{-1} over-regularizes (reconstruction 0.765, near the mean-feature baseline 0.776); β=10^{-2} is the operating point (0.852 / KL 2.85). The latent dimension is insensitive (cosine 0.847–0.854 across $d_z \in \{64, 128, 256\}$) and noise augmentation has minor effect.

β	recon cosine	KL	verdict
10^{-3}	0.893	14.8	posterior collapse
10^{-2}	0.852	2.85	operating point
10^{-1}	0.765	0.05	over-regularized

Table 12: The β-VAE frontier (z-dim 128). The mean-feature baseline reconstruction cosine is 0.776.

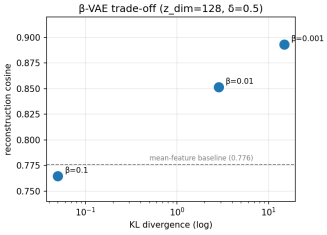

Figure 3: Reconstruction cosine vs. KL along the β frontier.

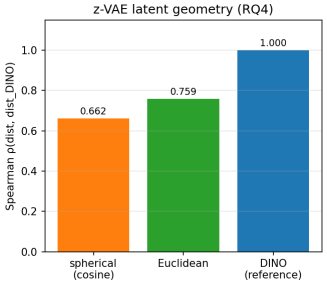

Figure 4: Spherical vs. Euclidean Spearman ρ on the variational latent space.

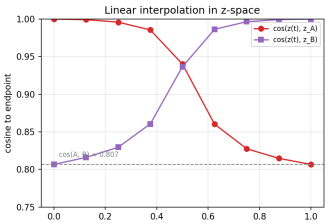

Figure 5: Interpolation midpoint cosine as a function of the interpolation coefficient t.

Rep.	distance	AUROC	equiv mean	equiv norm. σ	far mean	ρ(pixel-diff)
V0	μ Euclidean	0.9162	0.0351	1.0591	0.1394	0.6265
V1	μ Mahalanobis	0.9260	0.0454	1.0272	0.1885	0.6403
V2	sym. KL	0.9290	0.0021	2.0209	0.0262	0.6459
V3	W_2	0.9178	0.0352	1.0560	0.1402	0.6297
C	z_{inv} cosine	0.9999	0.0092	1.0122	0.2589	0.7976

Table 13: Native variational-space discrimination on the rollout set. All closed-form divergences clear the 0.85 AUROC gate without any discriminative training, but equivalence-radius stability (normalized σ) is worse than the discriminative head's—the gap the metric adapter closes. Symmetric KL achieves the best AUROC yet the worst radius stability: its ratio-of-variances term is hypersensitive to the posterior variance tails (a single small σ inflates the divergence), which is why the trust region uses W_2 or Euclidean on μ and keeps KL for offline diagnostics.

Local manifold, no global guarantee. For *equivalence* pairs (the pairs a trust region must fold together), latent midpoints are closer to both endpoints than the endpoints are to each other (midpoint cosine 0.945/0.933 to A/B vs. endpoint cosine 0.81): the manifold is smooth exactly where the trust region needs it. For *random, semantically distant* pairs, interpolation fails (midpoint cosine 0.469 vs. endpoints 0.807); this is reported as a limitation in the main text and is irrelevant for a trust region that constrains only local drift (Fig. 5).

Neighborhood semantic preservation (CKNNA). With k=20, neighbors in the variational latent space agree with DINO-feature neighbors at rate 0.792, against 0.603 for random neighbors; the dual-evaluation protocol (E-VAE-TR stage 3) confirms this on the rollout set (0.7575 vs. 0.6086).

F.3 Native VAE Discrimination (E-VAE-TR)

Before training any adapter on the VAE space, we measured how far the *native* variational posterior's closed-form divergences get on the rollout set (7,664 pairs). V0: Euclidean on μ; V1: Mahalanobis on μ with per-dimension precision averaged over the two posteriors, $d^2 = \sum_i (\Delta\mu_i)^2/\bar{\sigma}_i^2$; V2: symmetrized KL; V3: squared 2-Wasserstein, $W_2^2 = \|\mu_A - \mu_B\|_2^2 + \sum_i (\sigma_{A,i} - \sigma_{B,i})^2$; C: the discriminative z_{inv} cosine distance as the strong reference.

The dual-target evaluation (stage 3) confirms the picture: discrimination is solid (AUROC 0.9162/μ-Euclidean, 0.9178/W_2; gate 0.85), reconstruction is preserved (cosine 0.8730; gate 0.85), and local neighborhood structure holds (CKNNA 0.7575 vs. random 0.6086; gate 0.75), but the equivalence-radius sigma (0.0371/0.0372) misses the 0.02 target and global interpolation does not hold (midpoint 0.4692 $<$ endpoints 0.8067). Conclusion: the native variational space provides usable discrimination, reconstruction, and local structure, but not the radius stability a trust region requires—precisely the gap the distribution-preserving metric adapter (§E.3) is designed to close without touching the generative manifold.

F.4 Metric-Adapter Calibration (E-VAE-TR2)

E-VAE-TR2 evaluates four routes to a VAE-space trust region and selects *external calibration* (path 4). The design contrast that the experiment is built around:

- **Path 1 (uncertainty-weighted radius, zero training).** The Mahalanobis distance and the uncertainty-adaptive radius $r(\sigma_{\text{ref}}) = r_0\,(1 + \gamma\,\overline{\sigma}_{\text{ref}})$, $\gamma \in \{0.5, 1.0, 2.0\}$, evaluated purely analytically (configurations U0–U5: μ-Euclid, Mahalanobis $\times$ fixed, adaptive radius plus KL and per-pair-normalized W_2). Success criterion: equivalence $\sigma \leq 1.0$ with rollout AUROC ≥ 0.92 and false-positive rate $< 1\%$.
- **Path 2 (discriminative VAE, training).** The decoupled discriminative term added *inside* the VAE loss ($\lambda_{\text{disc}} \in \{0.3, 0.5, 1.0\}$, λ_e=2, plus a λ_e=5 variant), moving the generative space itself. Success requires the same discrimination gains without degrading reconstruction (≥ 0.85) or CKNNA (≥ 0.75).
- **Path 3 (dual-space fallback).** A z_{inv} tight discriminative radius plus the VAE's analytic uncertainty as a complementary constraint, if the VAE space cannot stand alone.
- **Path 4 (metric adapter, preferred).** The distribution-preserving adapter of §E.3: calibration is fully *external* to the frozen generative space. Sub-route (b) (preserving the Gaussian output, hence closed-form KL/W_2 and calibrated uncertainty σ_t) is chosen over the point-estimate sub-route (a), which would forfeit the variational advantages that justify the VAE space in the first place. The distance-preservation regularizer $\lambda_{\text{distill}} \in \{0, 0.1, 0.5\}$ is the main sweep, with the neighborhood-overlap check (k-NN agreement between z_t and $z_{\text{VAE}} \geq 70\%$) guarding against collapse. A dedicated calibration check verifies that the learned σ_t still tracks discrimination confidence (Spearman $\rho \geq 0.3$) before it may drive adaptive radii; otherwise the trust region falls back to fixed-radius Euclidean on μ_t.

The path-2 vs. path-4 comparison is reported as an independent finding: whether discriminative calibration belongs *inside* or *outside* the generative space. External calibration preserves the manifold by construction; the experiment tests whether internal calibration buys any discrimination that external calibration does not.

G Qualitative Examples

An equivalence pair the metric must fold. Figure 8 shows an absolute-coordinate SVG and its relative-coordinate rewrite: byte-level the programs differ substantially, yet the renderings are pixel-identical. Frozen hidden states place such a pair *farther* apart than a single-token coordinate negation that destroys the drawing ($d = 0.062$ vs. 0.001); the trained latent metric assigns it distance ≈ 0.009 (§F.1).

Reward hacking under a token-space trust region. The B0 baseline (GRPO with the token-KL syntax anchor only, i.e., a purely token-space trust region; the Ours (SFT+RL) configuration of Tables 2 and 6) provides a concrete instance of the failure the method targets. Under the efficiency term of the reward, the policy shortens its programs: on Image-to-SVG the average output drops from 1.1k tokens (SFT) to 508 tokens (Table 6), a 54% reduction. This shortening is not benign compression. Over N = 2,000 rollouts per run, the automatic detector flags 8.1 $\pm$ 1.2% of rollouts as viewBox collapse and 21.8 $\pm$ 2.5% as length collapse (Table 5): on roughly one in five rollouts the policy has found a cheaper representation of "compliance" that discards the structural detail the reference image requires. Crucially, this collapse is invisible in the aggregate metrics—overall success rate and fidelity still improve over SFT (SR 94.00%→99.83% on Image-to-SVG; Text-to-SVG FID

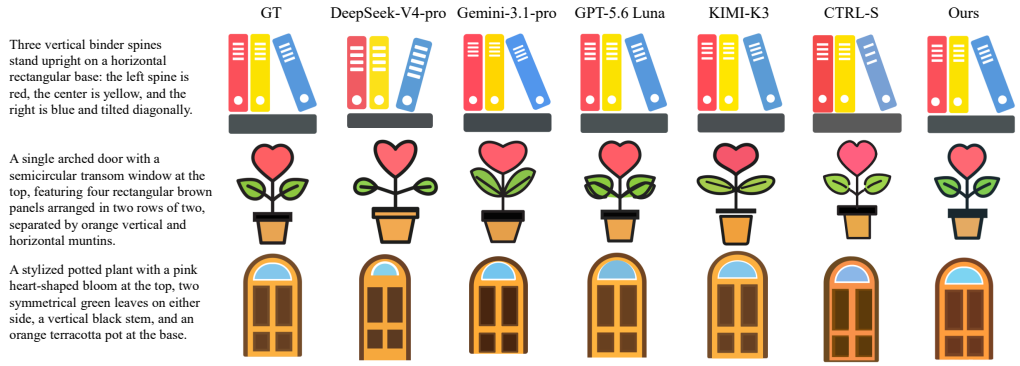

Figure 6: Qualitative comparison for Image-to-SVG.

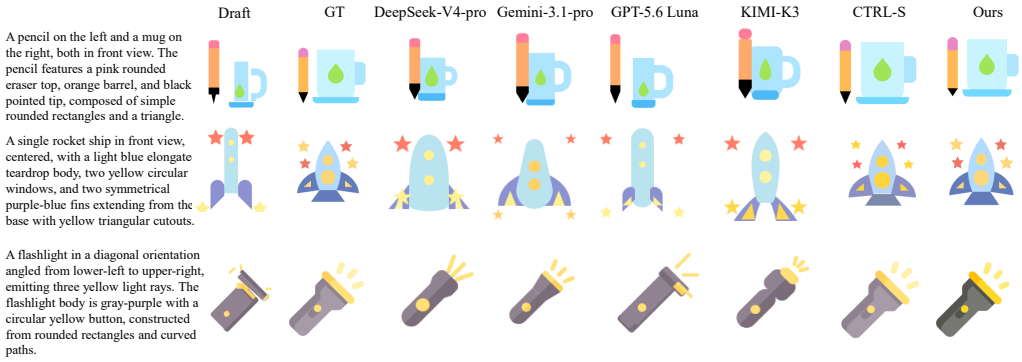

Figure 7: Qualitative comparison for SVG refinement.

Original (absolute coordinates)

```
<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 100 100">
  <path d="M 10 10 L 90 10 L 90 90 L 10 90 Z"
        fill="none" stroke="#222" stroke-width="2"/>
  <circle cx="50" cy="50" r="8" fill="#c00"/>
</svg>
```

Equivalent (relative coordinates)

```
<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 100 100">
  <path d="m 10 10 l 80 0 l 0 80 l -80 0 z"
        fill="none" stroke="#222" stroke-width="2"/>
  <circle cx="50" cy="50" r="8" fill="#c00"/>
</svg>
```

Figure 8: An equivalence pair from the `abs2rel` family: the two programs render identically (pixel difference $< 0.5\%$) but differ at the byte level. The trust-region metric must fold such pairs together, which frozen hidden states fail to do.

19.213→11.425; Tables 2 and 6)—because the collapsed rollouts are a minority that the averages absorb. Token KL does not detect this drift because the drift is semantically smooth in token space; it is only visible in rendering space and—by design—in the latent space where the behavior anchor measures it on every rollout.

Manifold evidence. The three panels of Fig. 3–5 summarize the variational-space geometry behind the method: the β-KL frontier (operating point β=10^{-2}), the Euclidean-vs-spherical metric comparison on the Gaussian latent, and the interpolation midpoint profile (local pairs well-behaved; global pairs not). The interpolation figure also documents why the trust region is designed around *local* drift: the manifold is smooth exactly at the scale the constraint operates on.