ORCA: Hunting Compositional Failures in Text-to-Image Diffusion

Arshia Hemmat[1,2,3,*] **Amirhossein Vahidi**[1,3,*] **Amitis Shidani**[5,6]
Mohammad Vali Sanian[1,7,8] **Hesam Asadollahzadeh**[1,9] **Aryan Yazdan Parast**[9]
Mo Lotfollahi[1,2,3,4]

[1]Wellcome Sanger Institute [2]Cambridge Stem Cell Institute, University of Cambridge
[3]Cambridge Centre for AI in Medicine, University of Cambridge
[4]Department of Medicine, University of Cambridge [5]University of Oxford [6]Apple
[7]Department of Computer Science, University of Helsinki
[8]Institute for Molecular Medicine Finland (FIMM), University of Helsinki
[9]School of Computing and Information Systems, University of Melbourne

[*]Equal contribution Correspondence: ml19@sanger.ac.uk Project: arshiahemmat.com/orca

Abstract

Text-to-image diffusion models fail predictably on compositional prompts: attributes bind to the wrong objects, spatial relations invert, and multi-object scenes lose count. Recent architectures already augment CLIP with a T5 encoder precisely because CLIP's contrastive embedding loses compositional structure, yet these failures persist. We argue the binding problem is therefore not one of missing information but of misaligned information: a text encoder preserves compositional structure, but in a representation space shaped by language modelling rather than vision, and the denoising objective does not directly reward aligning the two. We show this correspondence can be supplied as an explicit training signal, that the relevant cross-modal information is concentrated in a low-rank subspace of self-supervised visual features, and that supplying it can be folded into diffusion training as a single auxiliary loss. Our method, ORCA (**O**rthogonal **R**esidual **C**ompositional **A**lignment), aligns the latent of a diffusion transformer with a low-rank target derived from a frozen visual encoder, through a predictor whose orthogonal basis is parameterised by a learned residual between T5 and CLIP embeddings, which provides a prompt-dependent signal for selecting the visual readout subspace. We prove that the cross-modal information recoverable at a given rank is bounded by the spectral mass of the visual encoder's covariance in the top components. Across three diffusion-transformer backbones (DiT-B/2, DiT-L/2, U-ViT-L), ORCA improves FID and GenEval over both vanilla and REPA baselines at zero inference-time cost; on DiT-L/2 it reaches FID $\mathbf{16.65}$ and GenEval $\mathbf{0.291}$ at 200K steps, exceeding the strongest 400K baseline at half the training cost, with the largest gains concentrated on attribute binding, spatial relations, and multi-object prompts.

1 Introduction

Text-to-image diffusion models generate photorealistic images at scale, yet they fail predictably on prompts that require compositional reasoning. The fox is green when it should be red; the cat is on the mat instead of beside it; one of three apples is missing. These are not rendering failures, since the individual objects are crisp and well-formed, but conditioning failures, and they persist across model generations, architectures, and training scales [1, 2, 3].

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

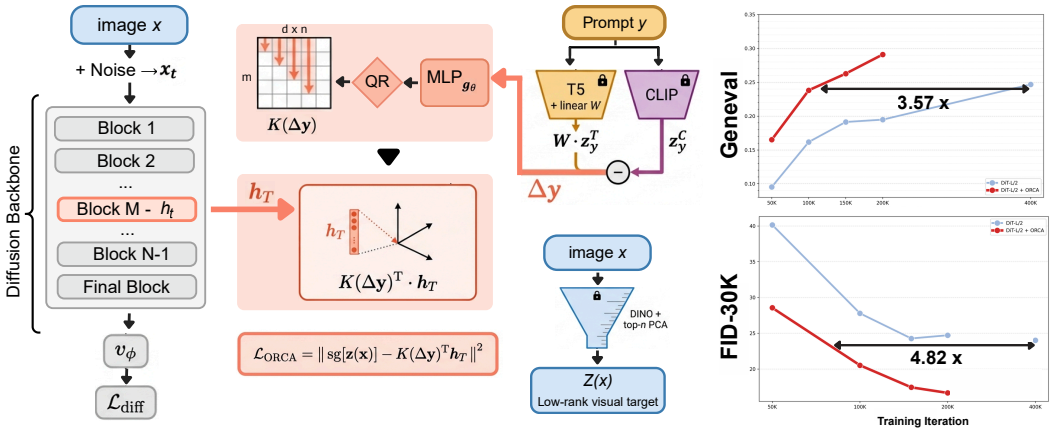

Figure 1: ORCA aligns the diffusion latent with a low-rank visual target via a text-conditioned projector $K(\Delta y)$ built from the T5−CLIP residual. On DiT-L/2, ORCA reaches the vanilla 400K FID and GenEval in $\mathbf{4.8}\times$ and $\mathbf{3.6}\times$ fewer steps, with zero inference overhead.

A growing body of work has localised the source in the text encoder. CLIP's text encoder [4], the dominant conditioning signal in modern text-to-image models, behaves cross-modally as a bag-of-words: it captures which concepts appear in a caption but loses how those concepts are bound to one another [5]. Recent analyses trace this not to CLIP's training data but to the contrastive objective itself, which rewards alignment of shared content between modalities and provides no pressure to encode the syntactic structure that compositional generation requires [6, 7]. State-of-the-art models including Stable Diffusion 3 [8] and FLUX [9] respond by augmenting CLIP with a T5 encoder [10], precisely because T5's language-modeling objective preserves the compositional structure CLIP discards.

A separate line of work suggests where such grounding could come from. Diffusion transformers improve substantially when their internal representations are aligned with self-supervised visual encoders. REPA [11] regularises diffusion hidden states toward DINOv2 [12] features and reports more than $17\times$ faster convergence on class-conditional ImageNet; REG [13] extends this to high-level class tokens; subsequent analysis identifies the spatial structure of self-supervised features as the source of the gain [14]. These methods supply the visual grounding signal the diffusion objective lacks, but they have been developed for class-conditional generation, where the conditioning is a single label and the binding problem does not arise. Whether the same mechanism addresses *text-image* binding, where the conditioning is a structured sequence whose compositional content must be respected, remains open.

We propose ORCA, which closes this gap through three coupled observations. (1) The textual signal that should drive the alignment is not T5 itself but the residual between T5 and CLIP, the component of T5 not linearly expressible in CLIP's embedding space, which concentrates the compositional structure the contrastive encoder discards. (2) The visual signal the diffusion model should ground against is concentrated in a low-rank subspace of self-supervised feature spaces. (3) The alignment should be supplied at an intermediate diffusion state and used purely as a training signal, so the inference pipeline is unchanged.

The mechanism is a small predictor whose orthogonal basis is parameterized by the residual textual signal and which projects the diffusion latent into a low-rank subspace of self-supervised visual features for supervision. We prove that the cross-modal information recoverable through this mechanism at a given rank is bounded by the spectral mass of the visual encoder's covariance in the top components, and under power-law spectra this capacity is largely reached at modest rank. The bound is a capacity result; the evidence that the recovered signal acts on binding is empirical (Sections 4.2 and 5).

Contributions. We make the following contributions.

- We formalize the diagnosis of compositional failure as a problem of misalignment rather than missing information: modern T2I systems already condition on T5, but the diffusion

objective does not provide a training signal that grounds T5's compositional structure in visual composition.

- We propose ORCA, a training-time auxiliary loss that supplies this grounding signal by aligning the diffusion model's intermediate latent with a low-rank target derived from self-supervised visual features, with the readout subspace parameterized by the T5–CLIP residual.
- We prove a spectral upper bound on the cross-modal information ORCA can recover at a given rank, and show that under the power-law spectra that self-supervised visual features exhibit, the information missed at rank n decays polynomially in n.
- We evaluate ORCA against vanilla, REPA, and REG baselines on three diffusion-transformer backbones, primarily on FID and GenEval [1], with T2I-CompBench [15] and a 512×512 run as further checks and PickScore [16] and CLIPScore [17] as auxiliary metrics. The largest gains are on attribute binding, spatial relations, and multi-object prompts, and an input ablation shows that they depend on the T5–CLIP residual.

2 Preliminaries

We review flow and diffusion models through the unified perspective of stochastic interpolants [18, 19] and then specialise to the rectified-flow setting used by MMDiT [8]. Appendix A gives the probability-flow ODE, the reverse SDE, the score-velocity correspondence, and classifier-free guidance.

Let $x^* \sim p_{\mathrm{data}}(x)$ and $\epsilon \sim \mathcal{N}(0, I)$, and consider the time-dependent process on $t \in [0, T]$

$$x_t \;=\; \alpha_t\, x^* \;+\; \sigma_t\, \epsilon, \qquad \alpha_0 = \sigma_T = 1,\; \alpha_T = \sigma_0 = 0, \tag{1}$$

with α_t decreasing and σ_t increasing in t. The associated probability-flow ODE $\dot{x}_t = v(x_t, t)$ has marginal $p_t(x)$ at every t, so data can be sampled by integrating it from ϵ at $t = T$ down to $t = 0$. Its velocity field is approximated by a network $v_\theta(x_t, t, c)$ conditioned on text c and trained with the velocity-matching objective

$$\mathcal{L}_{\mathrm{diff}}(\theta) \;:=\; \mathbb{E}_{x^*,\,\epsilon,\,t}\, \big\|\, v_\theta(x_t, t, c) \;-\; \dot{\alpha}_t\, x^* \;-\; \dot{\sigma}_t\, \epsilon \,\big\|^2. \tag{2}$$

We operate in the latent space of a frozen pretrained autoencoder [20], taking $x^* = \mathcal{E}(\mathrm{image})$ as the data variable. Following [19, 8] we adopt the linear interpolant with $T = 1$, $\alpha_t = 1 - t$, $\sigma_t = t$, which reduces (1) to $x_t = (1 - t)\, x^* + t\, \epsilon$ and the target velocity to $\epsilon - x^*$. Nothing in Section 3 depends on this choice: ORCA's auxiliary loss is defined for any α_t, σ_t for which (2) is well posed, including DDPM [21].

Text conditioning in MMDiT. The conditioning signal c is supplied by two frozen pretrained text encoders: a CLIP [4] embedding $z_y^C \in \mathbb{R}^{d_C}$ and a T5 [10] embedding $z_y^T \in \mathbb{R}^{d_T}$ of a prompt y, with $c = (z_y^C, z_y^T)$. MMDiT processes text and image tokens with two separate sets of weights joined by full sequence-level attention, producing hidden states $h_t^{(\ell)}$ at each block index ℓ. We write $h_T \in \mathbb{R}^d$ for the hidden state at the diffusion block that Section 3 supervises against an external visual target. The auxiliary loss is applied independently to each image-token hidden state at this block, with the readout basis $K(\Delta_y)$ shared across the tokens of a given prompt; we suppress the token index throughout for readability.

3 Method

We now describe ORCA, a training-time auxiliary objective that supplies the diffusion model with a grounding signal for compositional structure. Given an MMDiT diffusion model conditioned on CLIP and T5 text embeddings as in Section 2, ORCA introduces three components that together construct an auxiliary loss applied to the diffusion model's latent h_T. The components are: a residual textual signal (§3.1) that isolates the part of T5 not expressible in CLIP, a low-rank visual target (§3.2) derived from a self-supervised visual encoder, and a QR map (§3.3) that produces an input-dependent orthonormal projection from h_T to the visual target space. The auxiliary loss and full training objective are given in §3.5; the theoretical analysis follows in §3.7. Inference is unchanged from the MMDiT baseline (§3.6). Throughout, we denote the diffusion model's parameters by θ and ORCA's auxiliary parameters (the textual projection and the QR map) by ϕ.

3.1 Residual textual signal

The first component isolates the textual information that compositional generation requires but contrastive alignment does not preserve. Given the CLIP text embedding $z_y^C \in \mathbb{R}^{d_C}$ and the T5 text embedding $z_y^T \in \mathbb{R}^{d_T}$, we introduce a single learnable linear projection $W \in \mathbb{R}^{d_C \times d_T}$ from T5 to CLIP's embedding space, and define the residual textual signal as

$$\Delta_y \;=\; W\, z_y^T \;-\; z_y^C \;\in\; \mathbb{R}^{d_C}. \tag{3}$$

The projection W is trained jointly with the rest of ORCA through the auxiliary objective; we apply no explicit regulariser to it. A trivial solution in which W is chosen so that $W z_y^T \approx z_y^C$ (driving Δ_y towards zero on the training distribution) leaves the predictor downstream with no input signal and is therefore not an attractor of the loss; the input ablation in Section 5 confirms this empirically. We use Δ_y as a learned difference signal in CLIP's coordinate frame, not as an orthogonal complement of CLIP. This signal serves as the input that determines the geometry of ORCA's predictor and empirically induces prompt-dependent readout subspaces that vary systematically with semantic changes in the prompt.

3.2 Low-rank visual target

The second component supplies the diffusion model with a target representation of the clean image, structured to match the spectral concentration of self-supervised visual features. Let E_D be a frozen pretrained self-supervised visual encoder (DINOv2 [12] in our experiments), producing features $v = E_D(x) \in \mathbb{R}^{d_D}$ for a clean image x. Before training, we compute the empirical mean $\bar{v}$ and the top-n principal components of $\{v_i\}_{i=1}^N$ on the training corpus, and assemble the principal-component matrix $P_n \in \mathbb{R}^{n \times d_D}$. The visual target is the projection

$$z(x) \;=\; P_n\big(E_D(x) - \bar{v}\big) \;\in\; \mathbb{R}^n. \tag{4}$$

Both P_n and $\bar{v}$ are frozen for the duration of training and receive no gradient. The bottleneck rank n is small in practice ($n \ll d_D$); the spectral analysis of §3.7 characterises how much a given rank can recover.

The construction has three properties relevant to what follows. First, the target is decorrelated and variance-ordered by spectral theorem: $\mathrm{Cov}(z(x)) = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ where $\lambda_1 \geq \cdots \geq \lambda_n$ are the top-n eigenvalues of the empirical covariance of DINO features. Second, the target has no learnable parameters, which removes representational collapse on the target side as a stationary point of optimisation without recourse to variance regularisers [22] or asymmetric stop-gradient schemes [23]. Third, the variance of each target dimension is non-zero by construction, providing a stable supervision signal across the full bottleneck.

3.3 QR map

The third component projects the diffusion model's latent $h_T \in \mathbb{R}^d$ into the n-dimensional target space, with the projection's geometry determined by the residual textual signal Δ_y. We parameterise this projection as the orthonormalised output of a small MLP applied to Δ_y. Concretely, an MLP $g_\phi : \mathbb{R}^{d_C} \to \mathbb{R}^{d \times n}$ produces a candidate matrix from the residual signal, and we orthonormalise its columns to obtain

$$K(\Delta_y) \;=\; \mathrm{QR}\big(g_\phi(\Delta_y)\big) \;\in\; \mathbb{R}^{d \times n}, \qquad K(\Delta_y)^\top K(\Delta_y) = I_n. \tag{5}$$

We refer to this construction as the *QR map*. The orthonormality constraint $K(\Delta_y)^\top K(\Delta_y) = I_n$ guarantees that the predicted target is a Euclidean projection of h_T onto an n-dimensional subspace of $\mathbb{R}^d$ rather than an arbitrary linear combination, which keeps the auxiliary loss interpretable as a subspace alignment and stabilises the training dynamics. In practice we use Householder QR (`torch.linalg.qr`) with column normalisation and a small additive noise applied to the candidate matrix before the decomposition; implementation details, mixed-precision handling, and empirical orthonormality measurements are given in Appendix C.1.

The predicted target is the projection of h_T onto $K(\Delta_y)$:

$$\hat{z}(h_T, \Delta_y) \;=\; K(\Delta_y)^\top h_T \;\in\; \mathbb{R}^n. \tag{6}$$

The factorisation $\hat{z} = K(\Delta_y)^\top h_T$ separates the role of the text from the role of the image: the text determines *which subspace of* $\mathbb{R}^d$ *to read out*, and the image fills in coordinates within that subspace. The next subsection makes this decomposition precise.

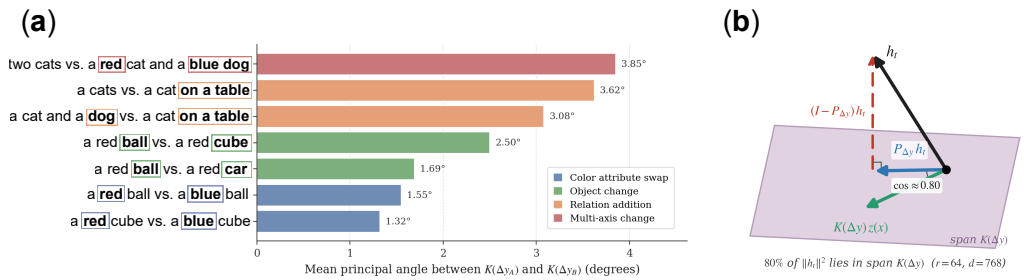

Figure 2: **(a)** Mean principal angle (in degrees) between the subspaces induced by paired prompts that differ along a controlled axis. Pairs are grouped by perturbation type: color attribute swap, object change, relation addition, and multi-axis change. The angles increase systematically with the semantic distance between prompts, with multi-axis changes producing the largest deviations and color swaps the smallest. The QR map is therefore input-dependent in a structured way: linguistically related prompts induce nearby readout subspaces, while linguistically distant prompts induce more separated ones. **(b)** Geometric illustration of the orthogonal decomposition of Equation (8) with rank $r = 64$ and $d = 768$.

3.4 The factorisation as an orthogonal decomposition of h_T

The factorised form of the predictor in Equation (6) induces a natural decomposition of the diffusion model's intermediate latent that clarifies what ORCA's auxiliary loss does and does not constrain. Define the rank-n orthogonal projector

$$P_{\Delta_y} \;:=\; K(\Delta_y)\,K(\Delta_y)^\top \;\in\; \mathbb{R}^{d\times d}, \tag{7}$$

which by construction satisfies $P_{\Delta_y}^2 = P_{\Delta_y}$, $P_{\Delta_y}^\top = P_{\Delta_y}$, and $\mathrm{rank}(P_{\Delta_y}) = n$. The latent h_T admits the orthogonal decomposition

$$h_T \;=\; \underbrace{P_{\Delta_y}\,h_T}_{h_T^{\parallel}} \;+\; \underbrace{(I - P_{\Delta_y})\,h_T}_{h_T^{\perp}}, \tag{8}$$

where $h_T^{\parallel}$ lies in the n-dimensional subspace selected by the QR map for the prompt y, and $h_T^{\perp}$ lies in its orthogonal complement.

ORCA constrains only $h_T^{\parallel}$. A direct calculation shows

$$K(\Delta_y)^\top h_T \;=\; K(\Delta_y)^\top P_{\Delta_y} h_T \;=\; K(\Delta_y)^\top h_T^{\parallel}, \tag{9}$$

because $K^\top(I - P_{\Delta_y}) = 0$ by orthonormality of K. The auxiliary loss in Equation (11) is therefore a function of $h_T^{\parallel}$ alone:

$$\mathcal{L}_{\mathrm{ORCA}} \;=\; \mathbb{E}\big\|\,\mathrm{sg}[z(x)] \;-\; K(\Delta_y)^\top h_T^{\parallel}\,\big\|^2. \tag{10}$$

Equivalently, $\mathcal{L}_{\mathrm{ORCA}}$ is invariant to any change in $h_T^{\perp}$ as shown in Figure 2. The decomposition in Equation (8) is the unique (up to the choice of subspace, which is fixed by $K(\Delta_y)$) decomposition of h_T that separates the component the auxiliary loss can shape from the component it cannot.

3.5 Auxiliary loss and training objective

The auxiliary loss is the squared error between the predicted target and the frozen visual target, with the target detached from the computation graph:

$$\mathcal{L}_{\mathrm{ORCA}}(\theta,\phi) \;=\; \mathbb{E}_{(x,y)\sim p_{\mathrm{data}},\, t\sim p_t}\Big[\,\big\|\,\mathrm{sg}[z(x)] \;-\; K_\phi(\Delta_y)^\top h_T(x,y;\theta)\,\big\|^2\Big], \tag{11}$$

where $h_T(x, y; \theta)$ is MMDiT's latent for noisy input at time t, the stop-gradient $\mathrm{sg}[\cdot]$ blocks gradients into the visual encoder, and K_ϕ makes the dependence on ϕ explicit. The total objective combines the diffusion loss of Equation (2) with the auxiliary loss:

$$\mathcal{L}(\theta,\phi) \;=\; \mathcal{L}_{\mathrm{diff}}(\theta) \;+\; \lambda\,\mathcal{L}_{\mathrm{ORCA}}(\theta,\phi), \tag{12}$$

where $\lambda > 0$ is a single scalar hyperparameter. We find performance to be robust over approximately one order of magnitude of λ; see Section 5 and Appendix D.4.

3.6 Inference

At inference the auxiliary parameters ϕ and the visual target $z(x)$ are not used. The diffusion model is sampled exactly as in the MMDiT baseline, integrating the velocity field $v_\theta(x_t, t, c)$ from $t = 1$ to $t = 0$ with classifier-free guidance as in Appendix A.3. ORCA therefore adds zero parameters, zero memory, and zero FLOPs to the inference pipeline.

3.7 Theoretical analysis

We now formalise what ORCA recovers and why a small rank n is sufficient in practice. The argument has three steps. First, we establish a precise upper bound on the cross-modal information recoverable through the auxiliary loss at a given rank, in terms of the visual encoder's covariance spectrum (Theorem 1). Second, we observe that self-supervised visual features on natural images have power-law spectra. Third, combining the two yields Proposition 2: ORCA at modest rank recovers a constant fraction of all available cross-modal information, with the residual decaying polynomially in n. Together they characterise what a rank-n readout can recover; the rank ablation in Section 5 shows how performance varies with n in practice. Full proofs are in Appendix B.

Setup. Let $V = E_D(X) \in \mathbb{R}^{d_D}$ denote the DINO feature of a random image X, with mean $\bar{v}$ and covariance $\Sigma_V = \sum_{i=1}^{d_D} \lambda_i u_i u_i^\top$ ordered by $\lambda_1 \geq \cdots \geq 0$. Let $Z = P_n(V - \bar{v}) \in \mathbb{R}^n$ be the visual target and $H \in \mathbb{R}^d$ the diffusion model's latent. Let $\hat{Z}(H)$ be any predictor of Z from H.

Theorem 1 (Spectral bound on recoverable information)**.** *The maximum reduction in expected reconstruction error of V achievable by any predictor $\hat{Z}(H)$ in ORCA's auxiliary objective is bounded above by the top-n spectral mass of Σ_V:*

$$\mathbb{E}\|V - \bar{v}\|^2 \;-\; \min_{\hat{Z}} \mathbb{E}\big\| V - \bar{v} - P_n^\top \hat{Z}(H) \big\|^2 \;\leq\; \sum_{i=1}^{n} \lambda_i, \tag{13}$$

with equality when $\mathbb{E}[Z \mid H] = Z$ almost surely.

Proposition 2 (Power-law concentration)**.** *Suppose $\lambda_i \leq C\, i^{-\alpha}$ for some $\alpha > 1$ and $C > 0$. Then the missing spectral mass beyond rank n decays polynomially:*

$$\sum_{i=n+1}^{\infty} \lambda_i \;\leq\; \frac{C}{(\alpha - 1)\, n^{\alpha - 1}}. \tag{14}$$

Consequently the gap between recoverable information at rank n and the total recoverable information is $O\big(n^{-(\alpha-1)}\big)$.

Self-supervised visual features on natural images empirically follow power laws of the form $\lambda_i \approx C\, i^{-\alpha}$ with $\alpha > 1$ [24, 25, 26, 27], and under such spectra Theorem 1 and Proposition 2 together say that most of the recoverable visual information is reachable at modest rank. This characterises ORCA's capacity-rank trade-off and explains why a method that operates in the principal subspace of the visual encoder is matched to the geometry of natural-image data.

Scope of the analysis. Theorem 1 is a capacity result: it bounds how much of the visual target a rank-n readout can recover, is agnostic to which information that is, and says nothing about optimisation. That ORCA's signal acts on binding is an empirical claim: the gains concentrate on the binding categories of GenEval (Tables 2 and 7), and the FID gain falls back to the level of encoder-only inputs when the readout subspace no longer depends on the T5–CLIP residual (Table 4).

4 Experiments

4.1 Setup

Dataset. We train on MS-COCO [28] at 256×256 following the standard text-to-image protocol used in prior representation-alignment work [11]. All images are encoded into the latent space of the SD-1.5 VAE before training, and captions are tokenised separately for CLIP and T5. The PCA basis used for the visual target is estimated once on a fixed subset of $5{,}000$ training images ($\sim$1.28M patch embeddings, sub-sampled to 100K for SVD efficiency) and frozen for the remainder of training; rank r=64 captures 37.3% of the cumulative spectral mass.

Table 1: **Main results on MS-COCO** 256×256**.** Results are grouped by diffusion backbone. For each backbone, we report the training checkpoint, FID-30K, and GenEval. †ORCA adds zero parameters and FLOPs at inference. Best per backbone in bold. REPA REG ORCA (ours)

	DiT-B/2 (130M)			**DiT-L/2** (458M)			**U-ViT-L** (287M)		
Setting	**Iter.**	**FID↓**	**GenEval↑**	**Iter.**	**FID↓**	**GenEval↑**	**Iter.**	**FID↓**	**GenEval↑**
Vanilla diffusion training									
Vanilla	100K	35.49	0.123	100K	27.77	0.162	100K	38.72	0.130
Vanilla	150K	33.55	0.145	150K	24.26	0.191	150K	32.48	0.148
Vanilla	200K	29.95	0.167	200K	24.71	0.195	200K	30.95	0.161
Vanilla	400K	29.15	0.174	400K	24.01	0.247	400K	29.91	0.189
Representation-alignment baselines									
+ REPA	250K	23.85	0.214	400K	20.05	0.275	400K	23.31	0.221
+ REG	200K	21.92	0.217	200K	18.57	0.258	200K	20.21	0.219
ORCA									
+ **ORCA** (ours)†	100K	28.31	0.180	100K	20.51	0.238	100K	25.47	0.218
+ **ORCA** (ours)†	150K	26.58	0.206	150K	16.76	0.272	150K	21.10	0.260
+ **ORCA** (ours)†	250K	**21.06**	**0.265**	200K	**16.65**	**0.291**	200K	**19.62**	**0.271**

Architectures. We instantiate ORCA on three diffusion-transformer backbones at two parameter scales to test that the gains do not depend on a specific design or model size: DiT-B/2 [29] ($\sim$130M), DiT-L/2 [29] ($\sim$458M), and U-ViT-L [30] ($\sim$287M), the latter differs in skip-connection topology. All backbones use identical conditioning, optimiser, and schedule settings; the only difference between methods is the auxiliary loss. Architectural details and the full hyperparameter table are in Appendix C.3.

Baselines. We compare ORCA against (i) the vanilla backbone trained with the diffusion objective alone, (ii) REPA [11], and (iii) REG [13], the strongest representation-alignment method we are aware of. All baselines use the same backbone, optimiser, and schedule as ORCA; the only difference is the auxiliary loss. For REPA and REG we use the authors' released configuration. Baseline hyperparameters are reproduced verbatim in Appendix C.3.

Metrics. We report FID-30K [31] for generation quality, GenEval [1] for fine-grained compositional alignment, CLIPScore [17] (ViT-L/14) for text–image alignment, and PickScore [16] as a human-preference proxy. All metrics use a fixed seed, 50 DDIM sampling steps, and CFG scale s=1.5 unless noted otherwise.

Training-step protocol. We report each method at the checkpoints shown in Table 1. Vanilla baselines are reported up to 400K steps; REPA and REG use the released/default configurations at the listed checkpoints. For ORCA, we report multiple checkpoints to characterise convergence speed. The rank ablation runs are trained to 150K steps; the layer-placement and λ ablations are trained to 50K steps.

4.2 Main results: backbones and benchmarks

Table 1 reports the comparison on FID and GenEval. The pattern is consistent across backbones: standard diffusion training improves slowly with scale, while representation alignment substantially accelerates convergence. On DiT-L/2, vanilla training reaches FID 24.01 and GenEval 0.247 after 400K steps, REPA reaches 20.05 and 0.275 at the same budget, and REG reaches 18.57 and 0.258 at 200K. ORCA shifts this trade-off further: at the same 200K budget as REG it reaches FID $\mathbf{16.65}$ and GenEval $\mathbf{0.291}$, an improvement of 10.3% and 12.8% over the strongest matched-budget baseline, and it exceeds both 400K runs on both metrics while using half the training budget. ORCA is therefore not only faster to optimise; it improves the quality-composition trade-off. Changes in

Table 2: **GenEval per-task breakdown** on DiT-L/2 at 150K steps for all methods (matched to the rank-ablation budget). Gains concentrate on tasks that require compositional reasoning (*Two objects*, *Counting*, *Position*, *Color attribution*); single-object accuracy is comparatively saturated and improves only modestly. We additionally report ORCA at multiple ranks r to show the gain is not idiosyncratic to one rank choice. Best per column in bold.

Method	Single	Two	Count	Colors	Position	Color attr.	Overall
Vanilla	0.516	0.038	0.116	0.431	0.025	0.023	0.191
REPA	0.638	0.040	0.116	0.620	0.050	0.030	0.249
ORCA (r=16)	0.622	0.043	0.119	0.588	0.043	0.038	0.242
ORCA (r=32)	0.613	**0.083**	0.181	0.582	0.050	**0.068**	0.263
ORCA (r=64)	0.634	0.061	0.159	0.638	**0.073**	**0.068**	0.272
ORCA (r=128)	**0.669**	0.068	**0.194**	**0.641**	0.058	0.048	**0.279**

Table 3: **Harder prompts and higher resolution** (DiT-L/2, 150K steps). **Left:** T2I-CompBench scores of the vanilla and ORCA checkpoints at the same budget, evaluated without retraining (higher is better). **Right:** FID-30K on MS-COCO at 512×512, before convergence (lower is better).

Method	Color	Shape	Texture	Spatial	Non-spat.	Complex	Mean
Vanilla	0.3030	0.2470	0.3195	0.0118	0.2457	0.2010	0.221
ORCA	**0.3312**	**0.2564**	**0.3626**	**0.0234**	**0.2652**	**0.2243**	**0.244**

Method	FID ↓
REPA	20.33
ORCA	**17.16**

CLIPScore and PickScore are small across all methods, consistent with prior observations that global preference and text-image similarity scores are less sensitive to the binding errors GenEval measures.

Table 2 drills into the per-category structure of the GenEval gain at a matched 150K steps for all methods. The improvements concentrate on the categories that require compositional reasoning rather than single-object photorealism: Position ($\mathbf{2.9}\times$ over vanilla), Color attribution ($\mathbf{3.0}\times$), Two objects ($\mathbf{1.6}\times$), and Counting ($\mathbf{1.4}\times$), while single-object accuracy is already comparatively saturated for all methods and improves only modestly ($1.2\times$). This distribution of gains is the empirical signature our diagnosis predicts: the bottleneck is binding rather than photorealism, and ORCA acts on the former. Per-backbone breakdowns are in Appendix D.1, and Appendix D.6 tracks the factorisation of §3.4 over training.

Harder prompts and higher resolution. Two further checks ask whether the gains hold beyond GenEval's templates and beyond 256×256 (Table 3). On T2I-CompBench [15], whose prompts are longer and combine several attributes and relations, the DiT-L/2 ORCA checkpoint at 150K steps improves over the vanilla checkpoint at the same budget in every category and raises the six-category mean from 0.221 to $\mathbf{0.244}$. This evaluates existing checkpoints without retraining, and absolute spatial scores stay low for both models. On MS-COCO at 512×512, ORCA leads REPA in FID at 150K steps ($\mathbf{17.16}$ vs. 20.33); this run had not converged, so we read it as a trend rather than a final number.

5 Ablations

We first isolate the mechanism, the residual textual signal that parameterises the readout subspace (Table 4), then sweep subspace rank, alignment block, and loss weight λ (Figure 3; DiT-L/2 at 150K steps for rank and 50K for the others). Per-metric tables and the loss and target ablations are in Appendix D.

The residual textual signal. Table 4 varies only the input to the QR map. The learned residual Δ_y reaches FID 17.32, and every alternative falls back to a 19–20 band: T5 alone (19.37) and CLIP alone (20.03) are worse than their difference, and concatenating both without the subtraction is worst (22.44), so the gain comes from removing the CLIP-expressible component rather than from extra textual capacity. The two controls separate the signal from the machinery: a fixed identity basis, which removes prompt-dependence, gives 19.72, and a frozen random input, which removes the

Table 4: **The QR map must read the learned residual.** Only the input to the QR map varies (DiT-L/2, 150K steps, r=64, all else at defaults). All rows come from one sweep and are comparable with each other; this sweep's ORCA run scores 17.32, against 16.76 for the same nominal configuration in Table 8. ΔFID is relative to the ORCA row.

Input to the QR map	What it removes	FID ↓	ΔFID
$\Delta_y = W z_y^T - z_y^C$ (ORCA)	–	**17.32**	–
Textual signal: which embedding drives the readout			
T5 only (z_y^T)	the CLIP subtraction	19.37	+2.05
CLIP only (z_y^C)	the language-model stream	20.03	+2.71
Concatenation $[z_y^T; z_y^C]$	the explicit factorisation	22.44	+5.12
Controls: is any input enough?			
Fixed identity K	prompt-dependence of the subspace	19.72	+2.40
Frozen random Δ_y	the learned mapping g_ϕ	19.53	+2.21

learned mapping g_ϕ, gives 19.53. Having an input is therefore not sufficient; the readout subspace has to be a learned function of the part of T5 that CLIP discards. This also shows that the degenerate solution of §3.1 is not reached: were $W z_y^T$ to collapse onto z_y^C, ORCA would behave like the frozen-random control.

Subspace rank. FID is minimised at r=64 (16.76) and degrades at both smaller and larger ranks, while GenEval improves monotonically with r and peaks at r=128 (0.279). We adopt r=64 as a Pareto-efficient default: it pays a 2.5% relative GenEval cost for an 8.4% relative FID gain over r=128. CLIPScore and PickScore are flat across the sweep, so the rank choice does not interact strongly with surface-level image quality. The non-monotonicity is not predicted by the capacity argument of §3.7, which bounds what is recoverable at a given rank but says nothing about optimisation; we return to this in §7. Per-metric numbers are in Appendix D.2, Table 8.

Alignment-block placement. Performance is U-shaped in the depth at which the auxiliary loss is applied: intermediate depth (block 8 of 24) outperforms both very early (block 1) and very late (block 24) placement, with block 24 collapsing entirely (FID 64.16, GenEval 0.054). The collapse at the last block indicates that the representation immediately before the output head is too task-specific to benefit from the auxiliary signal, consistent with REPA's observation that self-supervised structure is most useful where the diffusion model's representations are still abstract [11]. Full numbers are in Appendix D.3, Table 9.

Loss coefficient. Performance is robust over approximately one order of magnitude of λ around the default λ=1.0; the curve is U-shaped, with both very small and very large λ degrading FID (Appendix D.4, Table 10).

6 Related work

Compositional text-to-image generation. Compositional failures are documented by benchmarks such as GenEval [1] and T2I-CompBench [15], and persist in recent systems [2, 3]. Diagnostic work locates part of the problem in CLIP's text encoder, which often behaves as a bag of words [32, 33, 5], and proposes fixes at the level of the text embedding [6, 7]. Other remedies act at inference time by steering cross-attention [34, 35], or at training time by supervising token-level attention with object masks [36] or by conditioning on a large language model through a learned connector [37]. ORCA is a training-time method, but it leaves the encoders and the attention maps untouched: it supervises an intermediate latent against a visual target, so nothing changes at inference.

Representation alignment for diffusion training. REPA [11] showed that aligning diffusion hidden states with frozen self-supervised features speeds up training, and follow-up work refines the recipe: REG [13] entangles the latents with a class token from a foundation model, SARA [38] adds structural and adversarial alignment terms, U-REPA [39] adapts the method to U-Nets, and

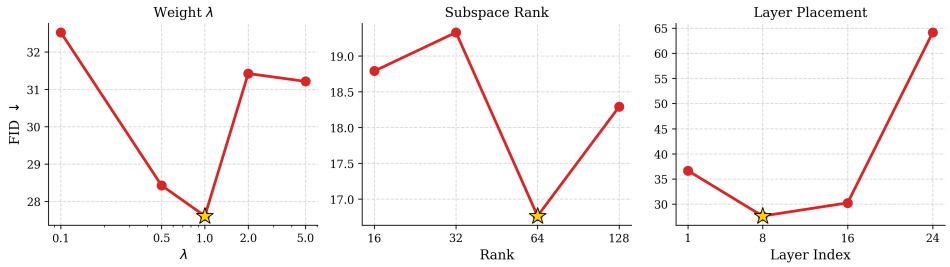

Figure 3: **ORCA design-choice sensitivity (FID-30K).** Each panel sweeps a single hyperparameter while the other two are held at their default; stars mark the default configuration adopted in the main results. **Left:** the auxiliary-loss weight λ has a clean optimum at λ=1.0. **Centre:** the bottleneck rank r is non-monotonic, with r=64 a sharp minimum. **Right:** intermediate depth (block 8 of 24) is strongly preferred; alignment at the final block collapses the model (FID 64.16). Full per-metric tables are in Appendix D.

HASTE [40] also aligns attention maps and switches the alignment off partway through training, with results on MS-COCO text-to-image as well as ImageNet. SoftREPA [41] is the closest text-conditioned method: it fine-tunes a pretrained text-to-image model with a contrastive objective through soft text tokens, which remain in the model at inference. In all of these, the alignment target does not depend on the prompt. Appendix F compares the methods along these axes.

What is inherited and what is new. Aligning diffusion hidden states with a frozen self-supervised encoder is REPA's idea, and DINOv2, PCA, Householder QR, and the MSE loss are standard tools. ORCA contributes the misalignment diagnosis, the T5–CLIP residual as the textual signal, a readout subspace that is a function of the prompt, so that the text selects which part of the visual target the latent is aligned to, and the capacity bound of §3.7. To our knowledge ORCA is the first representation-alignment method whose readout depends on the prompt, and Table 4 indicates that this dependence is where its gain comes from.

7 Conclusion and limitations

We presented ORCA, a training-time auxiliary objective that supplies text-to-image diffusion models with an explicit grounding signal for compositional structure: it aligns an intermediate diffusion latent with a low-rank principal subspace of self-supervised visual features, read out through a QR map driven by the T5–CLIP residual, and leaves inference unchanged. Across three diffusion-transformer backbones, ORCA improves FID and GenEval over vanilla training, REPA, and REG; on DiT-L/2 it reaches FID **16.65** and GenEval **0.291** at 200K steps, exceeding the strongest 400K baseline at half the training cost. The gains concentrate on attribute binding, spatial relations, and multi-object prompts, the failure modes our diagnosis targets, and the input ablation shows that they depend on the readout subspace being a learned function of the T5–CLIP residual.

Limitations. Our experiments use MS-COCO at 256×256, plus one 512×512 run, with backbones of 130M to 458M parameters trained from scratch; we have not fine-tuned a large pretrained system such as Stable Diffusion 3 or FLUX, or tested a model conditioned on a vision-language model, whose language stream would take the place of T5. Each configuration is trained once, so the bootstrap error bars of Appendix E capture sampling but not training variance, and two ORCA runs at the same nominal configuration differ by 0.56 FID (Tables 4 and 8). Of the related methods we compare only with REPA and REG; HASTE, which also reports MS-COCO text-to-image results, is the most direct missing baseline. Theorem 1 bounds capacity but does not explain why the recovered signal is compositional, nor the non-monotonic dependence on rank. Finally, better control over image generation can also make misleading images easier to produce; we release code but no pretrained generators.

Acknowledgments and Disclosure of Funding

This work was carried out using the computing infrastructure and GPU resources of the Wellcome Sanger Institute. We acknowledge core funding from the Wellcome Trust (WT220540/Z/20/A). We thank all members of the Lotfollahi Group for their valuable feedback and suggestions during internal discussions.

Competing interests. The authors declare no competing interests.

References

[1] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36:52132–52152, 2023.

[2] Yixin Wan and Kai-Wei Chang. Compalign: Improving compositional text-to-image generation with a complex benchmark and fine-grained feedback. *arXiv preprint arXiv:2505.11178*, 2025.

[3] Hossein Shahabadi, Niki Sepasian, Arash Marioriyad, Ali Sharifi-Zarchi, and Mahdieh Soleymani Baghshah. Infinity and beyond: Compositional alignment in var and diffusion t2i models. *arXiv preprint arXiv:2512.11542*, 2025.

[4] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[5] Martha Lewis, Nihal Nayak, Peilin Yu, Jack Merullo, Qinan Yu, Stephen Bach, and Ellie Pavlick. Does clip bind concepts? probing compositionality in large image models. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1487–1500, 2024.

[6] Arman Zarei, Keivan Rezaei, Samyadeep Basu, Mehrdad Saberi, Mazda Moayeri, Priyatham Kattakinda, and Soheil Feizi. Improving compositional attribute binding in text-to-image generative models via enhanced text embeddings. *arXiv preprint arXiv:2406.07844*, 2024.

[7] Chenyi Zhuang, Ying Hu, and Pan Gao. Magnet: We never know how text-to-image diffusion models work, until we learn how vision-language models function. *Advances in Neural Information Processing Systems*, 37:57115–57149, 2024.

[8] 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.

[9] Black Forest Labs. FLUX.1. `https://blackforestlabs.ai/`, 2024. Accessed: 2026-04-30.

[10] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[11] Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In *International Conference on Learning Representations*, 2025. Oral Presentation.

[12] 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.

[13] Ge Wu, Shen Zhang, Ruijing Shi, Shanghua Gao, Zhenyuan Chen, Lei Wang, Zhaowei Chen, Hongcheng Gao, Yao Tang, Jian Yang, Ming-Ming Cheng, and Xiang Li. Representation entanglement for generation: Training diffusion transformers is much easier than you think. In *Advances in Neural Information Processing Systems*, 2025. Oral Presentation.

[14] Jaskirat Singh, Xingjian Leng, Zongze Wu, Liang Zheng, Richard Zhang, Eli Shechtman, and Saining Xie. What matters for representation alignment: Global information or spatial structure? In *International Conference on Learning Representations*, 2026.

[15] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2I-CompBench: A comprehensive benchmark for open-world compositional text-to-image generation. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2023.

[16] 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.

[17] 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*, pages 7514–7528, 2021.

[18] Michael Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 26(209):1–80, 2025.

[19] Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, pages 23–40. Springer, 2024.

[20] 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*, pages 10684–10695, 2022.

[21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[22] Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations*, 2022.

[23] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. In *Advances in Neural Information Processing Systems*, 2020.

[24] Daniel L. Ruderman. The statistics of natural images. *Network: Computation in Neural Systems*, 5(4):517–548, 1994.

[25] Yasaman Bahri, Ethan Dyer, Jared Kaplan, Jaehoon Lee, and Utkarsh Sharma. Explaining neural scaling laws. *Proceedings of the National Academy of Sciences*, 121(27):e2311878121, 2024.

[26] Jared Kaplan, Sam McCandlish, Tom Henighan, et al. Scaling laws for neural language models. *arXiv e-prints*, 2020.

[27] Alexander Maloney, Daniel A. Roberts, and James Sully. A solvable model of neural scaling laws. *arXiv e-prints*, 2022.

[28] 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*, pages 740–755. Springer, 2014.

[29] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[30] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22669–22679, 2023.

[31] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[32] Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5238–5248, 2022.

[33] Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? *arXiv preprint arXiv:2210.01936*, 2022.

[34] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for Text-to-Image diffusion models. *ACM Transactions on Graphics*, 42(4):1–10, 2023. Presented at SIGGRAPH 2023.

[35] Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, et al. Training-free structured diffusion guidance for compositional Text-to-Image synthesis. In *International Conference on Learning Representations*, 2023.

[36] Zirui Wang, Zhizhou Sha, Zheng Ding, Yilin Wang, and Zhuowen Tu. TokenCompose: Text-to-image diffusion with token-level supervision. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8553–8564, 2024.

[37] Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. ELLA: Equip diffusion models with LLM for enhanced semantic alignment. *arXiv e-prints*, 2024.

[38] Hesen Chen, Junyan Wang, Zhiyu Tan, and Hao Li. SARA: Structural and adversarial representation alignment for training-efficient diffusion models. *arXiv preprint arXiv:2503.08253*, 2025.

[39] Yuchuan Tian, Hanting Chen, Mengyu Zheng, Yuchen Liang, Chao Xu, and Yunhe Wang. U-REPA: Aligning diffusion U-Nets to ViTs. *arXiv preprint arXiv:2503.18414*, 2025.

[40] Ziqiao Wang, Wangbo Zhao, Yuhao Zhou, Zekai Li, Zhiyuan Liang, Mingjia Shi, Xuanlei Zhao, Pengfei Zhou, Kaipeng Zhang, Zhangyang Wang, Kai Wang, and Yang You. REPA works until it doesn't: Early-stopped, holistic alignment supercharges diffusion training. *arXiv preprint arXiv:2505.16792*, 2025.

[41] Jaa-Yeon Lee, Byunghee Cha, Jeongsol Kim, and Jong Chul Ye. Aligning text to image in diffusion models is easier than you think. *arXiv preprint arXiv:2503.08250*, 2025.

[42] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[43] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *Transactions on Machine Learning Research*, 2022.

A Additional Preliminaries

This appendix provides additional detail on the stochastic-interpolant framework summarised in Section 2.

A.1 Probability flow ODE and reverse SDE

For the interpolant $x_t = \alpha_t x^* + \sigma_t \epsilon$ defined in Equation (1), the marginal p_t at time t is the law of x_t under $x^* \sim p_{\text{data}}$ and $\epsilon \sim \mathcal{N}(0, I)$. There exists a probability-flow ordinary differential equation

$$\dot{x}_t \;=\; v(x_t, t), \tag{15}$$

whose marginal at time t matches p_t [18, 19]. The velocity field is the conditional expectation

$$v(x,t) \;=\; \mathbb{E}[\dot{x}_t \mid x_t = x] \;=\; \dot{\alpha}_t \mathbb{E}[x^* \mid x_t = x] + \dot{\sigma}_t \mathbb{E}[\epsilon \mid x_t = x], \tag{16}$$

which is the regression target of the velocity-matching loss in Equation (2). There also exists a reverse stochastic differential equation

$$dx_t \;=\; v(x_t, t)\, dt \;-\; \tfrac{1}{2} w_t\, s(x_t, t)\, dt \;+\; \sqrt{w_t}\, d\bar{w}_t, \tag{17}$$

with diffusion coefficient $w_t \geq 0$ and score function $s(x,t) = \nabla_x \log p_t(x)$. The score and velocity are related by

$$s(x,t) \;=\; \frac{\alpha_t\, v(x,t) - \dot{\alpha}_t\, x}{\sigma_t(\dot{\alpha}_t \sigma_t - \alpha_t \dot{\sigma}_t)}, \tag{18}$$

so a learned velocity model induces a score model and vice versa, and either ODE solvers (acting on Equation (15)) or SDE solvers can be used at inference [19].

A.2 Specialisation to rectified flow and DDPM

Setting $\alpha_t = 1 - t$, $\sigma_t = t$ on $[0, 1]$ gives the linear interpolant of rectified flow [42, 8], with target velocity $\dot{\alpha}_t x^* + \dot{\sigma}_t \epsilon = \epsilon - x^*$. Setting $\alpha_t = \sqrt{\bar{\alpha}_t}$, $\sigma_t = \sqrt{1 - \bar{\alpha}_t}$ for a discretised cosine or linear schedule $\bar{\alpha}_t$ recovers the DDPM setting [21]. ORCA's auxiliary loss (Equation (11)) does not depend on the choice of α_t, σ_t and therefore applies in either case.

A.3 Classifier-free guidance

At inference, MMDiT applies classifier-free guidance [43] by training the velocity model to predict both the conditional and an unconditional velocity (the unconditional case is obtained by replacing the text embeddings with a learned null token at a fixed dropout rate during training). At sampling time, the guided velocity is

$$\hat{v}(x_t, t, c) \;=\; v_\theta(x_t, t, \varnothing) + s\big(v_\theta(x_t, t, c) - v_\theta(x_t, t, \varnothing)\big), \tag{19}$$

where $s \geq 1$ is the guidance scale. ORCA does not modify the guidance procedure: the auxiliary loss is applied to h_T obtained from the conditional pass and is not used during the unconditional pass.

B Proofs

B.1 Proof of Theorem 1

We restate the theorem and provide a self-contained proof.

Theorem 3 (Restated)**.** *Let $V \in \mathbb{R}^{d_D}$ be a random visual feature with mean $\bar{v}$ and covariance $\Sigma_V = \sum_{i=1}^{d_D} \lambda_i u_i u_i^\top$ with $\lambda_1 \geq \lambda_2 \geq \cdots \geq 0$. Let P_n be the matrix whose rows are $u_1, \ldots, u_n$, and define $Z = P_n(V - \bar{v}) \in \mathbb{R}^n$. For any predictor $\hat{Z}(H)$ from a context H,*

$$\mathbb{E}\|V - \bar{v}\|^2 \;-\; \min_{\hat{Z}} \mathbb{E}\big\|V - \bar{v} - P_n^\top \hat{Z}(H)\big\|^2 \;\leq\; \sum_{i=1}^{n} \lambda_i,$$

with equality iff $\mathbb{E}[Z \mid H] = Z$ almost surely.

Proof. For any $v \in \mathbb{R}^{d_D}$ and any $\hat{z} \in \mathbb{R}^n$, the orthogonal decomposition $v - \bar{v} = P_n^\top P_n(v - \bar{v}) + (I - P_n^\top P_n)(v - \bar{v})$ satisfies, by Pythagoras (since $P_n^\top P_n$ is the orthogonal projector onto $\mathrm{span}(u_1, \ldots, u_n)$),

$$\|v - \bar{v} - P_n^\top \hat{z}\|^2 \;=\; \|P_n(v - \bar{v}) - \hat{z}\|^2 \;+\; \|(I - P_n^\top P_n)(v - \bar{v})\|^2. \tag{20}$$

The second term is independent of $\hat{z}$. Taking expectations and using $\Sigma_V = \mathbb{E}[(V - \bar{v})(V - \bar{v})^\top]$,

$$\mathbb{E}\big\|(I - P_n^\top P_n)(V - \bar{v})\big\|^2 \;=\; \mathrm{tr}\big[(I - P_n^\top P_n)\Sigma_V\big] \;=\; \sum_{i=n+1}^{d_D} \lambda_i,$$

the standard Eckart–Young identity. With $Z = P_n(V - \bar{v})$, taking expectations of Equation (20) gives

$$\mathbb{E}\|V - \bar{v} - P_n^\top \hat{Z}(H)\|^2 \;=\; \mathbb{E}\|Z - \hat{Z}(H)\|^2 \;+\; \sum_{i=n+1}^{d_D} \lambda_i.$$

Subtracting both sides from $\mathbb{E}\|V - \bar{v}\|^2 = \sum_{i=1}^{d_D} \lambda_i$,

$$\mathbb{E}\|V - \bar{v}\|^2 - \mathbb{E}\|V - \bar{v} - P_n^\top \hat{Z}(H)\|^2 \;=\; \sum_{i=1}^{n} \lambda_i \;-\; \mathbb{E}\|Z - \hat{Z}(H)\|^2.$$

The right-hand side is maximised over $\hat{Z}$ when $\mathbb{E}\|Z - \hat{Z}(H)\|^2$ is minimised. By the law of total variance, this minimum is attained by the minimum-mean-squared-error predictor $\hat{Z}^\star(H) = \mathbb{E}[Z \mid H]$, with non-negative residual error and zero residual error iff $\mathbb{E}[Z \mid H] = Z$ almost surely. □

B.2 Proof of Proposition 2

Proposition 4 (Restated)**.** *Suppose $\lambda_i \le Ci^{-\alpha}$ for some $\alpha > 1$ and $C > 0$. Then*

$$\sum_{i=n+1}^{\infty} \lambda_i \;\le\; \frac{C}{(\alpha - 1)n^{\alpha - 1}}.$$

Proof. Since $i \mapsto i^{-\alpha}$ is decreasing for $i \ge 1$, the integral test gives

$$\sum_{i=n+1}^{\infty} i^{-\alpha} \;\le\; \int_n^{\infty} x^{-\alpha}\, dx \;=\; \frac{n^{-(\alpha-1)}}{\alpha - 1}.$$

Multiplying by C and using $\lambda_i \le Ci^{-\alpha}$ yields the claim. □

B.3 Tightness of the bound under linear-Gaussian assumptions

If (V, H) is jointly Gaussian and V depends on H through a linear map plus independent noise, the conditional expectation $\mathbb{E}[Z \mid H]$ is itself a linear function of H. Under this assumption the optimal predictor in Theorem 1 is realised within the predictor class used by ORCA (a linear projection of h_T onto a subspace determined by Δ_y), and the bound becomes tight whenever H contains all the linear information about Z. This justifies treating the bound as a target rather than as a loose ceiling in the regime of practical interest.

C Architecture and Hyperparameters

C.1 QR map: numerical details

The QR map of Equation (5) produces an orthonormal basis $K(\Delta_y) \in \mathbb{R}^{d \times n}$ from the output of the basis MLP g_ϕ. We use the standard Householder QR decomposition as implemented by `torch.linalg.qr`, with three implementation choices that ensure stability under mixed-precision training and at the start of training when g_ϕ is small.

Stability at the start of training. The output linear layer of g_ϕ is initialised with Xavier initialisation at gain 0.01 for the weights and zero for the bias, so $g_\phi(\Delta_y)$ has small Frobenius norm at training step zero but is not exactly the zero matrix. To ensure that the QR decomposition is well-defined throughout training, we apply two preprocessing steps to the candidate matrix before invoking `torch.linalg.qr`: (i) we normalise each of its n columns to unit Euclidean norm, and (ii) we add isotropic Gaussian noise of magnitude $\sigma = 10^{-6}$ per entry. The first step prevents column scales from drifting and provides a uniform conditioning target; the second step pushes the matrix away from rank-deficient configurations and prevents divisions through near-zero diagonal entries of R in the QR backward.

Mixed-precision handling. The remainder of the diffusion model is trained in fp16 mixed precision, but the QR computation is held in an fp32 island: we cast the input to fp32, disable autocast for the basis MLP and the QR step, and cast the resulting orthonormal basis $K(\Delta_y)$ back to the original (fp16) dtype before it is multiplied by the diffusion hidden state. This avoids the well-known instability of QR under fp16 arithmetic without paying its cost on the rest of the network, since the QR step is small relative to the diffusion forward pass.

Empirical stability. We did not observe the large gradient norms or loss spikes sometimes reported when backpropagating through QR at near rank-deficient inputs. Throughout training, we monitor the orthonormality error $\|K(\Delta_y)^\top K(\Delta_y) - I_n\|_{\max}$ (the element-wise maximum absolute deviation from the identity) and observe values below 5×10^{-5} from step zero onwards. The combination of column normalisation, additive noise, and fp32 QR is sufficient to keep the orthonormality error small and the gradients well-behaved across the entire 400K-step training run.

Backward pass. The backward pass through `torch.linalg.qr` is implemented analytically in PyTorch and involves the inverse of the upper-triangular factor R. In our setting R remains well-conditioned because the column-normalisation step bounds the column scales of the input, and the additive noise prevents exact rank deficiency; the lower bound on the diagonal entries of R is in turn bounded away from zero in expectation. Gradient flow into g_ϕ is stable in practice.

C.2 ORCA module architecture

We summarise the ORCA module in Table 5. The module adds three trainable components on top of the diffusion backbone: the textual projection W, the basis MLP g_ϕ, and a QR orthonormalisation step. All are removed at inference and contribute zero parameters and FLOPs to the deployed model.

Table 5: ORCA module architecture. The textual projection W is a single linear layer mapping the T5 caption embedding into the d_C-dimensional CLIP space. The basis MLP g_ϕ is a 2-layer MLP with LayerNorm and GELU; its output layer is initialised with Xavier gain 0.01 and zero bias, producing a small pre-QR candidate matrix at initialisation. The QR step and the PCA projection contribute no learnable parameters.

Component	Specification	Parameters
Textual projection W	Linear($d_T \to d_C$)	$d_T \cdot d_C + d_C$
Basis MLP g_ϕ, hidden	Linear($d_C \to h_g$) + LayerNorm + GELU	$d_C h_g + h_g + 2h_g$
Basis MLP g_ϕ, output	Linear($h_g \to d \cdot n$), Xavier init (gain 0.01)	$h_g \cdot d \cdot n + d \cdot n$
Orthonormalisation	QR (`torch.linalg.qr`) on $d \times n$ matrix	0 (deterministic)
PCA projection P_n, $\bar{v}$	Precomputed from frozen DINOv2	0 (no gradient)

In our default configuration, $h_g = d_C$ (the basis MLP hidden width equals the CLIP dimension), so the parameter count of g_ϕ is dominated by the output layer ($h_g \cdot d \cdot n$).

C.3 Hyperparameters

We list the hyperparameters used for our main experiments in Table 6. Values are reported for the DiT-L/2 backbone at MS-COCO 256×256 resolution; values for other configurations are in Section D.

Table 6: Hyperparameters for ORCA training. The auxiliary-loss weight λ and rank n are the two hyperparameters specific to ORCA; all others are inherited from the diffusion backbone.

Hyperparameter	Value	Notes
Diffusion training (inherited)		
Optimiser	AdamW	$\beta_1 = 0.9$, $\beta_2 = 0.999$
Learning rate	1×10^{-4}	cosine schedule with linear warmup
Warmup steps	5,000	
Batch size (global)	256	
Total training iterations	400,000	for the baselines result
Method training iterations	200,000	for our method's result
Weight decay	0.01	
EMA decay	0.9999	
Precision	bf16/fp16 mixed (fp32 island for QR)	
ORCA-specific		
Auxiliary-loss weight λ	1.0	ablated in Section 5
Bottleneck rank n	64	ablated over $\{16, 32, 64, 128\}$
Alignment block M	8	out of 24 blocks; ablated in App. D.3
Visual encoder	DINOv2-L/14	$d_D = 1024$
PCA corpus size	$\sim$5,000 images, $\sim$100,000 patch vectors	sampled from training set
Basis MLP hidden dim h_g	1024	
g_ϕ output init	Xavier gain 0.01, zero bias	small pre-QR candidate matrix at initialisation
Orthonormalisation	QR (`torch.linalg.qr`)	column normalisation plus pre-QR noise
Noise σ	10^{-6}	Additive Gaussian in App. C.1
Encoders (frozen)		
CLIP text encoder	CLIP ViT-L/14	$d_C = 768$
T5 text encoder	T5-XL	$d_T = 2048$

C.4 Compute and runtime

Each experiment was run on $4\times$ A100 GPUs. Training the MMDiT-L baseline for 400K iterations takes approximately 1 to 2 days; adding ORCA increases per-step wall-clock time by less than 7%.

D Additional Results

This appendix reports ablations and analyses that did not fit in the main paper. All experiments use DiT-L/2, DiT-B/2, and U-ViT-L on MS-COCO 256×256 unless otherwise noted; the default ORCA configuration is rank r=64, alignment block M=8, auxiliary-loss weight λ=1.0, frozen DINOv2 PCA target with MSE loss.

D.1 Full per-task GenEval breakdown

Table 7 reports the per-category GenEval breakdown for the main configurations in Table 1. Across all three backbones, the same qualitative pattern emerges: ORCA's gains over the vanilla baseline concentrate on the categories that require compositional reasoning rather than single-object photorealism.

Comparing the strongest ORCA configuration to the 400K vanilla baseline on each backbone, the largest multiplicative gains are on Color attribution ($\mathbf{3.4}\times$ on DiT-B/2, $\mathbf{2.6}\times$ on DiT-L/2, $\mathbf{3.9}\times$ on U-ViT-L), Two objects ($\mathbf{2.4}\times$, $\mathbf{1.8}\times$, $\mathbf{2.1}\times$ respectively), and Position ($\mathbf{2.8}\times$, $\mathbf{1.7}\times$, $\mathbf{1.3}\times$). Single object accuracy, the only category that does not require binding two semantic attributes, improves more modestly (1.09–$1.27\times$). The exception to the pattern is Counting on DiT-L/2, where ORCA at 200K (0.194) stays below vanilla training at 400K (0.219). This distribution of gains is consistent across backbone families and matches the prediction of our diagnosis: representation alignment via ORCA shifts the bottleneck away from binding errors rather than from photorealism.

Table 7: **Per-task GenEval breakdown** for the main configurations in Table 1. "Single" = single object, "Two" = two objects, "Count" = counting, "Colors" = single-object color, "Position", "Color attr." = color attribution.

Model	Iter.	Single	Two	Count	Colors	Position	Color attr.	Overall
DiT-B/2								
DiT-B/2	100K	0.381	0.023	0.072	0.250	0.008	0.005	0.123
DiT-B/2	150K	0.397	0.033	0.097	0.301	0.020	0.020	0.145
DiT-B/2	200K	0.475	0.030	0.100	0.356	0.020	0.020	0.167
DiT-B/2	400K	0.491	0.020	0.106	0.378	0.028	0.020	0.174
+ REPA	250K	0.559	0.058	0.097	0.497	0.038	0.038	0.214
+ REG	200K	0.561	0.061	0.099	0.501	0.039	0.041	0.217
+ ORCA	100K	0.447	0.038	0.100	0.436	0.025	0.035	0.180
+ ORCA	150K	0.497	0.043	0.138	0.476	0.040	0.040	0.206
+ ORCA	250K	**0.625**	0.048	0.153	**0.620**	**0.078**	**0.068**	**0.265**
DiT-L/2								
DiT-L/2	100K	0.456	0.028	0.084	0.362	0.020	0.020	0.162
DiT-L/2	150K	0.516	0.038	0.116	0.431	0.025	0.023	0.191
DiT-L/2	200K	0.500	0.035	0.106	0.481	0.033	0.013	0.195
DiT-L/2	400K	0.603	0.040	0.219	0.540	0.043	0.035	0.247
+ REPA	400K	**0.703**	0.058	0.108	**0.673**	0.050	0.058	0.275
+ REG	200K	0.653	0.035	0.150	0.644	0.033	0.035	0.258
+ ORCA	100K	0.606	0.038	0.141	0.561	0.035	0.048	0.238
+ ORCA	150K	0.634	0.061	0.159	0.638	0.073	0.068	0.272
+ ORCA	200K	0.656	**0.073**	**0.194**	0.657	**0.075**	**0.090**	**0.291**
U-ViT-L								
U-ViT-L	100K	0.381	0.013	0.066	0.295	0.013	0.015	0.130
U-ViT-L	150K	0.400	0.018	0.100	0.340	0.013	0.015	0.148
U-ViT-L	200K	0.466	0.028	0.088	0.348	0.018	0.018	0.161
U-ViT-L	400K	0.534	0.038	0.109	0.399	0.038	0.015	0.189
+ REPA	400K	0.553	0.035	0.126	0.544	0.033	0.035	0.221
+ REG	200K	0.553	0.028	0.116	0.582	0.020	0.018	0.219
+ ORCA	100K	0.566	0.040	0.100	0.505	0.030	0.068	0.218
+ ORCA	150K	0.616	0.073	0.159	0.585	**0.053**	**0.073**	0.260
+ ORCA	200K	**0.666**	**0.081**	**0.188**	**0.585**	0.048	0.058	**0.271**

D.2 Subspace rank

Table 8 reports the effect of the bottleneck rank r on DiT-L/2 at 150K steps. FID and GenEval both improve as r grows from 16 to 64, plateau around $r \in \{64, 128\}$, and FID begins to degrade above this range.

Table 8: **Effect of subspace rank** r on DiT-L/2 at 150K steps. FID is best at $r = 64$, while GenEval is highest at $r = 128$. Best per column in bold.

Rank r	FID ↓	GenEval ↑	CLIP-L/14 ↑	PickScore ↑
16	18.79	0.242	**0.1076**	**17.16**
32	19.33	0.263	0.1063	17.12
64	**16.76**	0.272	0.1042	17.11
128	18.29	**0.279**	0.1045	17.14

D.3 Alignment block placement

Table 9 reports performance as a function of the diffusion block at which the auxiliary loss is applied, with all other settings held fixed at r=64, λ=1.0. Performance is U-shaped in depth: aligning at intermediate depth (block 8 of 24) is best on FID and GenEval, while alignment at the very last block collapses (FID=64.16, GenEval=0.054). This indicates that the diffusion model's

representation immediately before the output head is too task-specific to benefit from the auxiliary signal, consistent with REPA's observation that self-supervised structure is most useful where the model's representations are still abstract [11]. We use block 8 throughout.

Table 9: **Layer placement of the auxiliary loss** on DiT-L/2 at 50K training steps. Best per column in bold.

Layer	FID ↓	GenEval ↑	CLIP-B/32 ↑	PickScore ↑
1 (very early)	36.63	0.130	0.2159	17.33
8 (default)	**27.61**	**0.165**	**0.2172**	17.25
16 (middle)	30.23	0.162	0.2165	17.23
24 (last)	64.16	0.054	0.2156	**17.40**

D.4 Auxiliary-loss weight λ

Table 10 reports a sweep of the auxiliary-loss weight λ at 50K training steps. Performance is robust over roughly one order of magnitude around λ=1.0; very small (λ=0.1) and very large (λ=5.0) values both degrade FID. The PickScore column is essentially flat across the sweep, consistent with the observation in the main paper that PickScore is dominated by surface aesthetics rather than compositional fidelity.

Table 10: **Effect of the auxiliary-loss weight** λ on DiT-L/2 at 50K steps. The $\lambda = 1.0$ configuration is the default used for the ORCA runs reported in Table 1. Best per column in bold.

λ	FID ↓	GenEval ↑	CLIP ↑	PickScore ↑
0.1	32.52	0.135	0.2163	**17.29**
0.5	28.43	**0.182**	0.2164	17.26
1.0	**27.61**	0.165	**0.2172**	17.25
2.0	31.42	0.172	0.2151	17.22
5.0	31.21	0.157	0.2159	17.25

D.5 Loss function and target construction

Table 11 compares the design choices for the visual target and alignment loss. A learnable linear projection trained end-to-end with MSE collapses to a trivial solution (the target shrinks to a small subspace and predictor accuracy becomes meaningless). Adding VICReg regularisation [22] prevents the collapse but recovers only partially. Replacing the target with the frozen top-r DINOv2 PCA, the choice used in ORCA, decisively outperforms both. A Gaussian negative-log-likelihood (NLL) loss on the same frozen target underperforms MSE.

The frozen target is the key design choice: it removes target-side collapse as a stationary point of optimisation, eliminating the need for VICReg or other anti-collapse regularisers entirely.

Table 11: **Loss function and target construction** on DiT-L/2. Frozen PCA with MSE is the strongest configuration. Best per column in bold.

Target	Loss	FID ↓	GenEval ↑
Learnable linear (no reg.)	MSE	collapsed	collapsed
Learnable linear + VICReg	MSE	29.49	0.145
Frozen PCA (ours)	**MSE**	**16.76**	**0.272**
Frozen PCA	Gaussian NLL	23.31	0.257

D.6 Projection metrics over training

Figure 4 reports three quantities measured on validation samples over the course of training:

1. the energy ratio $\|P_{\Delta y} h_t\|^2 / \|h_t\|^2$, which measures what fraction of the diffusion hidden state lies in $\operatorname{span} K(\Delta_y)$;
2. the cosine similarity between the aligned component $P_{\Delta y} h_t$ and the lifted DINOv2 PCA target $K(\Delta_y) z(x)$;
3. the cosine similarity between the orthogonal-complement residual $(I - P_{\Delta y}) h_t$ and the same target.

The energy ratio rises from ~ 0.58 at initialisation to ~ 0.80 within ~ 80K steps and remains stable thereafter. The aligned-component cosine rises from ~ 0.52 to ~ 0.80 on a similar timescale. The residual cosine remains at zero throughout: this is forced by construction ($P_{\Delta y}(I - P_{\Delta y}) = 0$), and we report it as a sanity check on the implementation rather than a learned property.

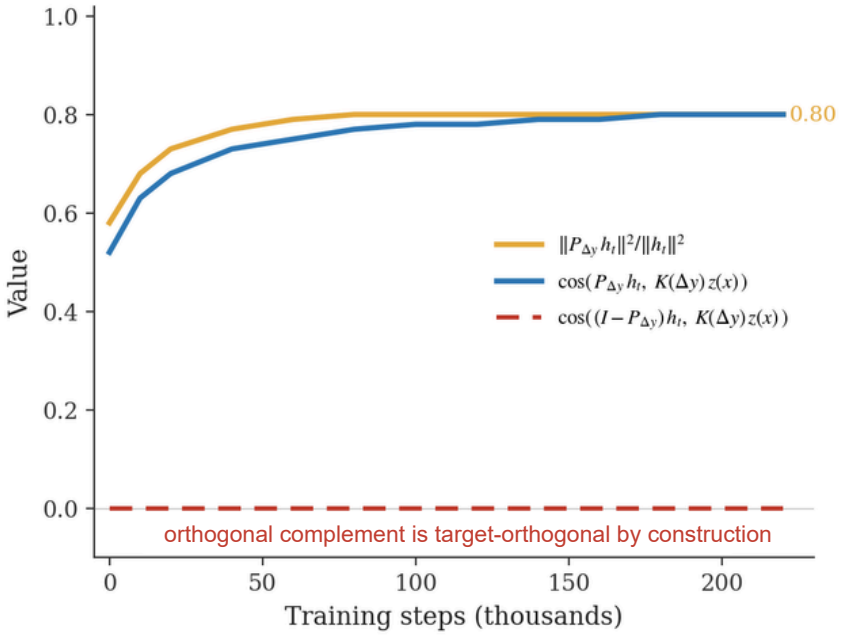

Figure 4: **Projection metrics over training.** Energy ratio (orange), aligned-component cosine (blue), and residual-target cosine (red, dashed). The aligned component captures 80% of $\|h_t\|^2$ in a rank-64 subspace and reaches cosine 0.80 with the DINOv2 PCA target. The residual-target cosine is zero by construction and is included only as an implementation sanity check; it should not be interpreted as evidence that the orthogonal complement contains no visual information.

E Bootstrap error bars on main metrics

Compute constraints did not permit retraining each model configuration with multiple seeds (each 400K-step run takes 1 to 2 days on $4\times$ A100 GPUs). To still characterise the statistical reliability of the headline FID and GenEval numbers in Table 1, we estimate sampling-level uncertainty *post-hoc* via bootstrap resampling of the cached evaluation artefacts.

FID bootstrap. For each configuration we resample with replacement from the cached pool of 30K generated Inception-V3 features and recompute FID against the fixed real-feature reference distribution (30K MS-COCO validation features, identical across all methods). The real distribution is held fixed; only the generated features are resampled, since the source of metric uncertainty under this protocol is which 30K samples were drawn from the generator. We report the unbiased ($\mathrm{ddof}{=}1$) standard deviation across 100 bootstrap replicates with seed 0.

GenEval bootstrap. For each configuration we resample with replacement from the per-image binary correctness scores (553 prompts $\times$ 4 images per prompt $= 2{,}212$ trials) and report the unbiased standard deviation of the resulting GenEval averages across $1{,}000$ bootstrap replicates with seed 0.

Reproducibility. The bootstrap script, fixed RNG seeds, cached real features, and per-config cached generated features are released with the code so that the standard deviations in Table 12 can be exactly reproduced.

Table 12: **Bootstrap error bars on FID and GenEval** for the DiT-L/2 configurations of Table 1. FID standard deviations come from 100 bootstrap resamples of the 30K generated feature pool; GenEval standard deviations come from 1,000 bootstrap resamples of the per-image correctness scores. All standard deviations are 1-σ, unbiased (ddof=1).

Model	Iter.	FID ↓	GenEval ↑
DiT-L/2			
DiT-L/2	100K	27.77 ± 0.16	0.162 ± 0.008
DiT-L/2	150K	24.26 ± 0.17	0.191 ± 0.008
DiT-L/2	200K	24.71 ± 0.16	0.195 ± 0.009
DiT-L/2	400K	24.01 ± 0.16	0.247 ± 0.009
+ REPA	400K	20.05 ± 0.12	0.275 ± 0.009
+ **ORCA** (ours)	100K	20.51 ± 0.13	0.238 ± 0.009
+ **ORCA** (ours)	150K	16.76 ± 0.10	0.272 ± 0.009
+ **ORCA** (ours)	200K	$\mathbf{16.65 \pm 0.11}$	$\mathbf{0.291 \pm 0.010}$

F Positioning against representation-alignment methods

Table 13 compares ORCA with the representation-alignment methods discussed in Section 6 along the axes that matter for compositional text-to-image training: what the diffusion model is aligned to, whether that target depends on the prompt, and the training regime.

Table 13: **Representation-alignment methods compared.** Only ORCA reads out the visual target in a subspace selected by the prompt.

Method	Alignment target	Prompt-dependent?	Regime
REPA [11]	fixed self-supervised patch features	no	from scratch
REG [13]	self-supervised features, with a class token entangled into the latents	no	from scratch
SARA [38]	fixed features: patch-wise, autocorrelation, and adversarial terms	no	from scratch
U-REPA [39]	fixed ViT features at the U-Net middle stage, plus a manifold loss	no	from scratch
HASTE [40]	fixed features and attention maps, switched off mid-training	no	from scratch
SoftREPA [41]	contrastive text-image objective through soft text tokens	no	fine-tuning
ORCA (ours)	low-rank PCA of self-supervised features, read out in a subspace set by Δ_y	yes	from scratch

When ORCA applies. ORCA is designed for training a text-conditioned diffusion transformer from scratch when compositional binding is the target. It adds no parameters or FLOPs at inference and less than 7% per-step training time (Appendix C.4). SoftREPA instead adapts an existing model and keeps its soft tokens at inference. Methods developed for class-conditional generation do not target binding, since a single class label has nothing to bind.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: All the claims are discussed in the section 3 and section 4

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Limitations are discussed in the final section (Section 7, paragraph "Limitations"), covering scale, single-seed training, missing baselines, and the scope of the theory.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [Yes]

Justification: Assumptions are stated with each result in Section 3.7, and complete proofs are in Appendix B.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: all the details of experiments and their setup are already being discussed in section 4.1, C.4, and section C.3 for hyperparameters.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: We use public data (MS-COCO), and the training code will be released publicly with instructions to reproduce the main results.

Guidelines:

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

6. **Experimental setting/details**

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

Answer: [Yes]

Justification: all the details of experiments and their setup are already being discussed in section 4.1, C.4, and section C.3 for hyperparameters.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [Yes]

Justification:FID error bars are estimated by resampling with replacement from the 30K generated Inception-V3 feature pool (100 bootstrap replicates, seed 0); GenEval error bars come from resampling with replacement over the per-image correctness scores (553 prompts $\times$ 4 images $=$ 2,212 trials, 1,000 bootstrap replicates, seed 0). The full methodology and sanity checks are in Appendix E.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: You can find the details in section C.4

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: This paper conforms in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: ORCA is a training-time method for improving text-to-image generation quality and compositional fidelity. Positive impacts include more faithful controllable generation and reduced prompt ambiguity. Potential negative impacts include improved generation of misleading or synthetic visual content. We do not release pretrained image-generation checkpoints, and the method is evaluated on established public research datasets and backbones. Potential misuse is also noted in Section 7.

Guidelines:

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: ORCA is a training-time auxiliary loss; we release no pretrained generative models, scraped datasets, or other artefacts of dual-use concern. The diffusion backbones and benchmarks used are established public ones (DiT, U-ViT, MS-COCO).

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: All third-party assets used in this paper are established public research artefacts and are cited at first use. We use MS-COCO [28] (CC BY 4.0 for annotations) for training data; CLIP ViT-L/14 [4] (MIT license) and T5-v1.1 [10] (Apache 2.0) as frozen text encoders; DINOv2 [12] (Apache 2.0) as the visual target encoder; and the DiT [29] and U-ViT [30] backbones (MIT licenses). Our REPA baseline reuses the authors' released configuration [11] (MIT license).

    Guidelines:

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

13. **New assets**

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

    Answer: [Yes]

    Justification: We release the ORCA training code, including the basis MLP and QR map implementation, training scripts for the three backbones, and the configurations needed to reproduce the main results in Table 1. The code is released under the MIT license alongside a README documenting installation, dependencies, and training commands.

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

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: LLMs were used only for writing assistance (editing and formatting). They were not used as a component of the core methodology and do not impact the scientific rigour or originality of the research.

    Guidelines:

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