CONTRAST-OF-CONTRASTS: ON THE SECOND ORDER OF DIFFUSION AND BRIDGE GUIDANCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion and bridge models are developed to capture a complex data distribution from known prior, while learning errors and potential mismatch between training and desired target distribution limit their generation quality. Recent guidance methods address this issue by extrapolating a primary denoiser away from a reference one with quality differences at each sampling step, while a single reference contrast provides only a first-order estimate of the desired sampling direction and may leave residual bias, leading to sub-optimal performance. Here, we introduce Contrast-of-Contrasts Guidance ($\mathrm{C}^2\mathrm{G}$), a general second-order framework that extends existing guidance with multi-level reference predictions along its underlying degradation trajectories. By contrasting adjacent reference differences, $\mathrm{C}^2\mathrm{G}$ captures the variation of the first-order steering direction along the degradation trajectory, aiming to recover the residual guidance signals missed by single-level guidance. Moreover, we introduce $\mathrm{Proj}\mathrm{C}^2\mathrm{G}$, which can inherit the tuned first-order guidance scale while decomposing and dynamically modulating the second-order correction along the directions parallel and orthogonal to the first-order guidance, exploiting their complementary effects throughout sampling. Extensive experiments show that $\mathrm{C}^2\mathrm{G}$ improves the primary metrics of classifier-free guidance, auto-guidance, self-guidance, and prior guidance across diffusion and bridge models for image generation, image-to-image translation, and image-to-video generation. Complete implementation will be publicly released.

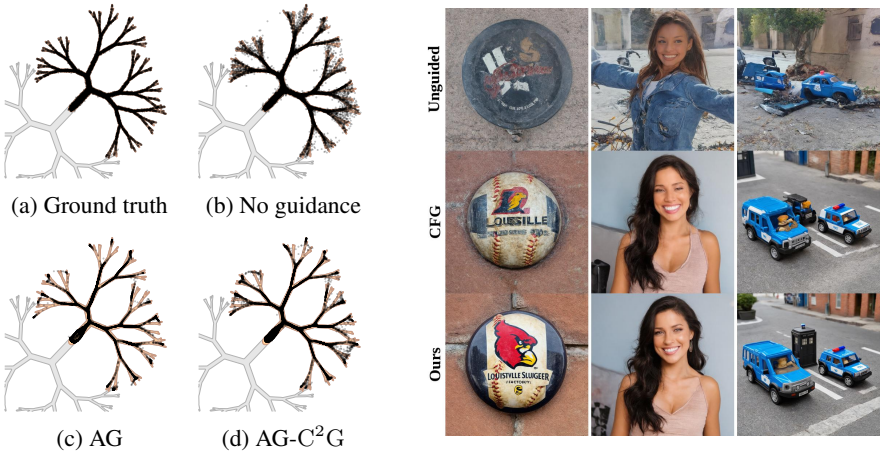
1 INTRODUCTION

Diffusion models (Ho et al., 2020; Song et al., 2021; Dhariwal & Nichol, 2021; Podell et al., 2024) learn time-dependent score functions to faithfully reconstruct a complex data distribution from a known simple prior distribution, *e.g.*, standard Gaussian, with an iterative denoising and refinement sampling process. Recently, bridge models (Liu et al., 2023; Zhou et al., 2024; Zheng et al., 2025) have been developed, where a clean prior can be directly exploited from the endpoint, enabling a data-to-data generation process and showing advantages in tasks with an informative prior, such as image-to-image translation (Chen et al., 2026b) and image-to-video generation (Wang et al., 2025).

In generation, learning errors in training and potential mismatch between the training and the desired target distribution in sampling result in a training-inference mismatch, inevitably restricting the quality of diffusion and bridge models. To address this issue, guidance methods (Ho & Salimans, 2021; Karras et al., 2024a; Ahn et al., 2024; Galashov et al., 2026) are designed to steer the generation direction at each inference step. Usually, these methods construct a reference denoising term whose quality differs from that of the primary denoising term and contrast them as a correction. For example, classifier-free guidance (CFG) (Ho & Salimans, 2021) concurrently learns an unconditional denoising term and contrasts it with the conditional one to emphasize condition alignment. Auto-guidance (AG) (Karras et al., 2024a) additionally develops a weak diffusion model, which is trained on the same task, condition information, and target distribution but with limited model capacity or training resources, and contrasts it with the strong one to enhance score accuracy. Other methods, such as self-guidance (SG) (Li et al., 2026) and prior guidance (PG) (Chen et al., 2026b), present the reference term by perturbing the model input, *i.e.*, the time step or prior information.

While these methods have constructed different reference terms to propose novel guidance mechanisms, they may still suffer from sub-optimal performance, as a single-level contrast mainly captures

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



(a) Ground truth (b) No guidance (c) AG (d) AG-C^2G

Figure 1: Qualitative motivation for Contrast-of-Contrasts Guidance (C^2G). **Left:** A toy Auto-guidance (AG) (Karras et al., 2024a) example compares ground truth, unguided generation, AG, and AG-C^2G. **Right:** Three MS-COCO (Lin et al., 2014) examples compare unguided generation, CFG results, and CFG-$\mathrm{Proj}C^2G$ results, following Self-guidance (Li et al., 2026). The CFG prompts comparison and more examples are included in Appendix F.

the first-order approximation of the desired steering direction and may leave room for further refinement. In this work, we introduce *Contrast-of-Contrasts Guidance* (C^2G), a general second-order guidance framework: C^2G refines an existing mechanism by constructing the denoising terms with multi-level quality differences along its underlying reference trajectory, without requiring any new steering objective. By contrasting adjacent reference differences, C^2G characterizes how the steering direction varies across degradation levels and aims to recover residual guidance signals missed by single-level guidance. As analyzed by an anisotropic analysis, the second-order signal enables direction-dependent refinement, whereas first-order guidance applies a uniform correction strength across eigendirections. This additional flexibility is valuable when the desired residual cannot be adequately represented by a single isotropic correction. Figure 1 illustrates our qualitative improvements in AG example and in CFG generation. Moreover, we develop $\mathrm{Proj}C^2G$, where we inherit the tuned first-order guidance scale of an existing method and decompose the second-order correction into components parallel to and orthogonal to the first-order guidance direction. By dynamically modulating these components along the sampling trajectory, $\mathrm{Proj}C^2G$ balances effective first-order strength with complementary higher-order steering, exploiting their cumulative advantages.

We evaluate C^2G across a diverse set of existing guidance mechanisms, including the training-based CFG and AG, as well as the training-free SG and PG. Together, these methods span diffusion and bridge generative frameworks, enabling a broad evaluation across image generation, text-to-image generation, image-to-image translation, and image-to-video generation tasks. Extensive experimental results demonstrate gains in the primary reported metrics across these settings, supporting multi-level reference contrast as a broadly applicable principle for second-order diffusion guidance.

2 Preliminary

2.1 Diffusion and Bridge Models: Distribution Reconstruction

Diffusion models (Ho et al., 2020; Song et al., 2021) transform a data distribution $p_0(\boldsymbol{x}_0)$ into a known prior distribution $p_T(\boldsymbol{x}_T)$, *e.g.*, standard Gaussian $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$, with pre-defined generative dynamics in a *forward* process. By learning time-dependent score functions $\boldsymbol{s}_{\boldsymbol{\theta}_{\text{diff}}}$, they are capable of reconstructing the target distribution from the simple prior distribution with an iterative denoising and refinement trajectory in the *reverse* process. The generative dynamics can be described by a forward stochastic differential equation (SDE) and its reverse-time counterpart as follows.

$$\begin{aligned} \text{forward:} \quad & \mathrm{d}\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_t, t)\mathrm{d}t + g(t)\mathrm{d}\boldsymbol{w}_t; \\ \text{reverse:} \quad & \mathrm{d}\boldsymbol{x}_t = [\boldsymbol{f}(\boldsymbol{x}_t, t) - g^2(t)\boldsymbol{s}_{\boldsymbol{\theta}_{\text{diff}}}(\boldsymbol{x}_t, t, \boldsymbol{c})]\mathrm{d}t + g(t)\mathrm{d}\bar{\boldsymbol{w}}_t, \end{aligned} \tag{1}$$

where $\boldsymbol{f}(\boldsymbol{x}_t, t)$ and $g(t)$ denote the drift and diffusion coefficients, respectively, and $\boldsymbol{c}$ denotes the condition information, *e.g.*, a text prompt in text-to-image generation. To estimate the score function required in generation, diffusion models have diverse model parameterization methods, *e.g.*, data, noise, score, and velocity (Salimans & Ho, 2022). Without loss of generality, we take data prediction as an example, where the network is optimized with

$$\arg\min_{\boldsymbol{\theta}_{\text{diff}}} \mathbb{E}_{\boldsymbol{x}_0,t,\boldsymbol{\epsilon}} \left\| \boldsymbol{x}_0 - D_{\boldsymbol{\theta}_{\text{diff}}}(\boldsymbol{x}_t, t, \boldsymbol{c}) \right\|_2^2. \tag{2}$$

Bridge models (Zhou et al., 2024; Zheng et al., 2025) have recently been designed to exploit an informative prior $\boldsymbol{x}_T$ from the beginning of the sampling trajectory, *e.g.*, a sketch in sketch-to-image generation, where the prior distribution is not restricted to additive Gaussian noise as in diffusion models. Similarly, their generative dynamics can be described by an SDE, while an additional drift term $\boldsymbol{h}(\boldsymbol{x}_t, t, \boldsymbol{x}_T)$ is incorporated to ensure that the forward process reaches a clean prior at $t = T$.

$$\begin{aligned} \text{forward:} \quad & \mathrm{d}\boldsymbol{x}_t = [\boldsymbol{f}(\boldsymbol{x}_t, t) + g^2(t)\boldsymbol{h}(\boldsymbol{x}_t, t, \boldsymbol{x}_T)]\mathrm{d}t + g(t)\mathrm{d}\boldsymbol{w}_t; \\ \text{reverse:} \quad & \mathrm{d}\boldsymbol{x}_t = [\boldsymbol{f}(\boldsymbol{x}_t, t) - g^2(t)(\boldsymbol{s}_{\boldsymbol{\theta}_{\text{bridge}}}(\boldsymbol{x}_t, t, \boldsymbol{x}_T, \boldsymbol{c}) - \boldsymbol{h}(\boldsymbol{x}_t, t, \boldsymbol{x}_T))]\mathrm{d}t + g(t)\mathrm{d}\bar{\boldsymbol{w}}_t. \end{aligned} \tag{3}$$

To estimate the bridge score function $\boldsymbol{s}_{\boldsymbol{\theta}_{\text{bridge}}}(\boldsymbol{x}_t, t, \boldsymbol{x}_T, \boldsymbol{c})$ from the clean prior $\boldsymbol{x}_T$ to the clean target $\boldsymbol{x}_0$, the network can be optimized with the same target prediction loss as diffusion models:

$$\arg\min_{\boldsymbol{\theta}_{\text{bridge}}} \mathbb{E}_{\boldsymbol{x}_0,\boldsymbol{x}_T,t,\boldsymbol{\epsilon}} \left\| \boldsymbol{x}_0 - D_{\boldsymbol{\theta}_{\text{bridge}}}(\boldsymbol{x}_t, t, \boldsymbol{x}_T, \boldsymbol{c}) \right\|_2^2. \tag{4}$$

2.2 Guidance Methods: Contrast Quality Differences

As introduced in AG (Karras et al., 2024a) and related work (Chen et al., 2026b), guidance methods of diffusion models usually construct two denoising terms with quality differences, namely a high-quality primary denoising result $D_{\boldsymbol{\theta}_{\text{good}}}(\boldsymbol{x}_t, t)$ and a low-quality reference result $D_{\boldsymbol{\theta}_{\text{bad}}}(\boldsymbol{x}_t, t)$, and extrapolate them with a scaling factor w to emphasize the desired attribute,

$$D_{\boldsymbol{\theta}}^{\text{guide}}(\boldsymbol{x}_t, t) = (1 + w)D_{\boldsymbol{\theta}_{\text{good}}}(\boldsymbol{x}_t, t) - wD_{\boldsymbol{\theta}_{\text{bad}}}(\boldsymbol{x}_t, t). \tag{5}$$

For example, CFG (Ho & Salimans, 2021) learns an unconditional score $D_{\boldsymbol{\theta}_{\text{bad}}}(\boldsymbol{x}_t, t)$, which is more difficult to learn than the conditional one $D_{\boldsymbol{\theta}_{\text{good}}}(\boldsymbol{x}_t, t, \boldsymbol{c})$ and trained with less training resources, to emphasize condition alignment. PG (Chen et al., 2026b) manually degrades the prior information to hinder pre-trained bridge models and result in a low-quality denoising result $D_{\boldsymbol{\theta}_{\text{bad}}}(\mathcal{H}(\boldsymbol{x}_t), t, \boldsymbol{c}, \boldsymbol{x}_T)$, to further encourage the advantage of bridge models on prior exploitation by contrasting it with pretrained $D_{\boldsymbol{\theta}_{\text{good}}}(\boldsymbol{x}_t, t, \boldsymbol{c}, \boldsymbol{x}_T)$. Throughout this paper, guidance weights use the primary-prediction-anchored convention in equation 5. In particular, CFG is written as $D_{\text{cond}} + w_1(D_{\text{cond}} - D_{\text{uncond}})$; the equivalent unconditional-anchored scale is $s_{\text{CFG}} = 1 + w_1$. An introduction of more guidance methods is provided in Appendix A.

3 Method

3.1 Motivation: Higher-order Guidance for More Faithful Steering

Analysis of guidance methods. As introduced in Section 2, score-based models (Sohl-Dickstein et al., 2015; Song et al., 2021) are developed to reconstruct the data distribution $p_{\text{data}} = p_0(\boldsymbol{x}_0)$ from the prior distribution $p_{\text{prior}} = p_T(\boldsymbol{x}_T)$ with learned time-dependent score functions that parameterize the reverse transition $p_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ from T to 0. Let $q_{\boldsymbol{\theta}}(\boldsymbol{x}_0)$ denote the terminal marginal induced by the full reverse path, as defined in Appendix A.3. Ideally, $q_{\boldsymbol{\theta}}(\boldsymbol{x}_0) \approx p_{\text{data}}(\boldsymbol{x}_0)$. However, finite model capacity, optimization errors, and limited training resources may introduce a distributional discrepancy: $D_{\text{KL}}(q_{\boldsymbol{\theta}}(\boldsymbol{x}_0) \| p_{\text{data}}(\boldsymbol{x}_0)) > 0$, which manifests as inaccurate samples or low-probability outliers (Karras et al., 2024a; Wang et al., 2026). Moreover, the learning objective, namely the ground-truth distribution p_{data}, may not be aligned with the desired one p_{target}, *e.g.*, when stronger condition alignment is desired. Guidance methods, as shown in equation 5, manipulate the denoising result at each inference step without changing pre-trained network parameters. Hence, they serve as post-processing algorithms to refine the denoising results, helping diffusion

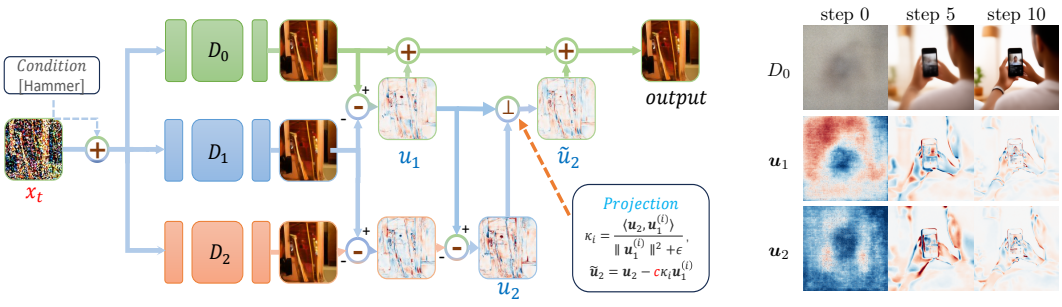


Figure 2: ProjC^{2}G mechanism and representative predictions. **Left:** The AutoGuidance pipeline forms first- and second-order contrasts from three predictions and partially projects the latter before combining them. **Right:** A CFG example at sampling steps 0, 5, and 10; rows show $D_0 = D_g$, $\boldsymbol{u}_1 = D_g - D_b$, and $\boldsymbol{u}_2 = D_g - 2D_m + D_b$. Red and blue pixels in the last two rows of the right panel indicate positive and negative residual values, respectively.

sampling approximate the target distribution more faithfully.

$$\hat{\boldsymbol{x}}_0^{\text{guide}} = D_{\boldsymbol{\theta}_{\text{good}}}(\boldsymbol{x}_t, t, \boldsymbol{c}) + w\Delta^{\text{gui}}, \quad \Delta^{\text{gui}} = D_{\boldsymbol{\theta}_{\text{good}}}(\boldsymbol{x}_t, t, \boldsymbol{c}) - D_{\boldsymbol{\theta}_{\text{bad}}}(\boldsymbol{x}_t, t, \boldsymbol{c}),$$
$$p_{\boldsymbol{\theta}}^{\text{guide}}(\boldsymbol{x}_{0:T}) := p_T(\boldsymbol{x}_T) \prod_{t=1}^{T} p_{\boldsymbol{\theta}}^{\text{guide}}\left(\boldsymbol{x}_{t-1} \mid \hat{\boldsymbol{x}}_{0,t}^{\text{guide}}, \boldsymbol{x}_t\right), \quad q_{\boldsymbol{\theta}}^{\text{guide}}(\boldsymbol{x}_0) \longrightarrow p_{\text{target}}(\boldsymbol{x}_0), \tag{6}$$

where Δ^{gui} denotes the improvement term introduced by guidance methods and $q_{\boldsymbol{\theta}}^{\text{guide}}$ is the terminal marginal induced by guidance, which we seek to align with $p_{\text{target}}(x_0)$. Hence, the mechanism of guidance methods is to construct a reference term $D_{\boldsymbol{\theta}_{\text{bad}}}$ and take its quality differences with the high-quality denoising term $D_{\boldsymbol{\theta}_{\text{good}}}$ to approximate the gap between $D_{\boldsymbol{\theta}_{\text{good}}}$ and the desired direction named D^{tar}. Extrapolating along this contrastive direction therefore steers the denoising prediction toward the target. In AG (Karras et al., 2024a), Δ^{gui} pushes the pre-trained diffusion model $D_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t, \boldsymbol{c})$ to reconstruct the target distribution more faithfully, *i.e.*, $p_{\text{target}} = p_{\text{data}}$. In CFG (Ho & Salimans, 2021; Chen et al., 2025), Δ^{gui} steers the conditional sampling process toward the desired terminal distribution $p_{\text{target}}(\boldsymbol{x}_0|\boldsymbol{c}) \propto p(\boldsymbol{x}_0|\boldsymbol{c}) \left[\frac{p(\boldsymbol{x}_0|\boldsymbol{c})}{p(\boldsymbol{x}_0)}\right]^w$; see Appendix A.3 for details.

Motivation of contrast-of-contrasts guidance. While recent methods (Karras et al., 2024a; Li et al., 2026; Chen et al., 2026b) have constructed novel low-quality denoising terms as the reference in equation 6 to boost diffusion generation quality, they usually steer the sampling process using a single contrast of quality difference. This first-order correction implicitly assumes that a single reference-induced direction is sufficient to characterize the residual from the learned high-quality denoising term to the desired target field. In practice, however, the discrepancy may vary nonlinearly across denoising quality levels: a single contrast can capture the dominant steering direction but may overlook residual bias and higher-order variation. Consequently, even an effective guidance mechanism may result in sub-optimal performance after the first-order correction. This observation motivates us to explore whether the guidance direction itself can be further refined through constructing multiple quality-level references. In the following section, we introduce C^{2}G, which exploits higher-order reference contrasts to compensate for the residual error left by first-order guidance.

3.2 Contrast-of-Contrasts Guidance

Second-order guidance mechanism. Consider three diffusion denoising results D_0, D_1, D_2 ($\boldsymbol{\theta}$ is omitted here for simplicity) sampled successively along the same degradation trajectory, from high to low quality. We define the first-order reference contrast and its second-order refinement as

$$\boldsymbol{u}_1 = D_0 - D_1, \qquad \boldsymbol{u}_2 = (D_0 - D_1) - (D_1 - D_2) = D_0 - 2D_1 + D_2. \tag{7}$$

While $\boldsymbol{u}_1$ is the conventional first-order guidance direction, $\boldsymbol{u}_2$ captures the variation of this contrast across adjacent degradation levels. We then augment the unguided prediction with both terms:

$$D^{\text{C}^2\text{G}} = D_0 + w_1\boldsymbol{u}_1 + w_2\boldsymbol{u}_2. \tag{8}$$

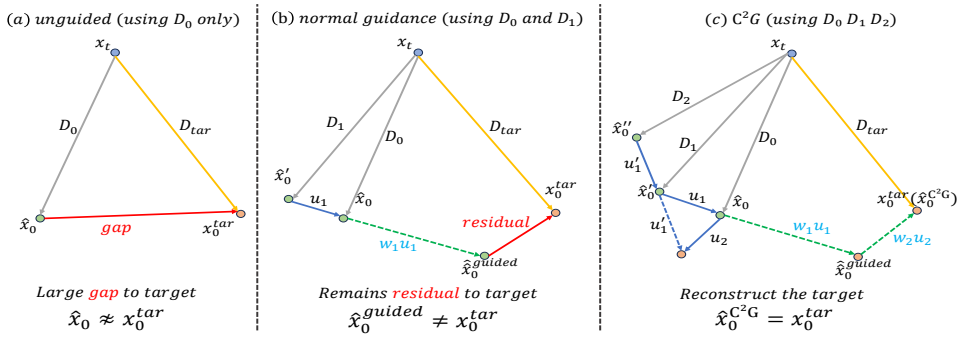


Figure 3: Conceptual geometry of C^{2}G. (a) The unguided prediction differs from the target; (b) first-order guidance reduces this gap but may leave a residual; (c) a second-order correction can further reduce the residual. Coincidence with the target in (c) illustrates an idealized case.

Here, w_1 and w_2 are scalar guidance weights. Figure 2 illustrates the mechanism and shows the corresponding first- and second-order contrasts in a CFG example. Unlike heuristic combinations of heterogeneous guidance methods, which mix guidance signals designed for different objectives or mechanisms, C^{2}G refines the reference contrast within a single underlying guidance mechanism. It is therefore conceptually orthogonal to guidance combination: the latter explores *which* steering signals are used, whereas C^{2}G improves *how* a selected steering signal is estimated. Accordingly, C^{2}G can be composed with existing methods, as shown in Section 4.

An anisotropic interpretation. To illustrate the additional flexibility introduced by second-order guidance, we consider an anisotropic Gaussian model in which different directions exhibit different degradation behaviors. In this setting, standard first-order guidance alone may not sufficiently refine the prediction. At each diffusion time step t, let $\boldsymbol{x}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_t)$, where $\boldsymbol{\Sigma}_t \succ 0$ and $\mathrm{tr}(\boldsymbol{\Sigma}_t) = d$. We fix t and consider three progressively degraded predictions $D_{\rho,t}(\boldsymbol{x}_t)$, indexed by $\rho \in \{0, 1, 2\}$:

$$D_{\rho,t}(\boldsymbol{x}_t) = D_{0,t}(\boldsymbol{x}_t) - \left[\rho \boldsymbol{I} + \frac{\rho(\rho-1)}{2}(\boldsymbol{\Sigma}_t - \boldsymbol{I})\right] \boldsymbol{x}_t. \tag{9}$$

Here, $\rho = 0$ corresponds to the strongest prediction, and a larger ρ value represents stronger degradation. The corresponding first- and second-order guidance directions are

$$\boldsymbol{u}_{1,t} = \boldsymbol{x}_t, \qquad \boldsymbol{u}_{2,t} = (\boldsymbol{I} - \boldsymbol{\Sigma}_t)\boldsymbol{x}_t, \qquad (\boldsymbol{u}_{2,t})_i = (1 - \lambda_{i,t}) x_{i,t}, \tag{10}$$

where $\lambda_{i,t}$ is the i-th eigenvalue of $\boldsymbol{\Sigma}_t$. The first-order guidance refinement applies the same scalar strength to every eigendirection, whereas the second-order refinement introduces the direction-dependent coefficient $w_2(1 - \lambda_{i,t})$. Their combined coordinate-wise response is therefore

$$(w_1 \boldsymbol{u}_{1,t} + w_2 \boldsymbol{u}_{2,t})_i = [w_1 + w_2(1 - \lambda_{i,t})]\, x_{i,t}. \tag{11}$$

Thus, while first-order guidance applies the same correction strength across eigendirections, the second-order term enables direction-dependent refinement, providing additional flexibility for anisotropic residuals. Figure 4 further validates this effect, with more details in Appendix D.

Multi-level reference contrasts. Conventional first-order guidance uses the contrast $D_0 - D_1$(or $D_0 - D_2$ in CFG) as a proxy for the residual between D_0 and the desired target field D_{tar}. However, this estimation is based on a single reference level and may therefore omit variations in the steering direction along the reference degradation trajectory. C^{2}G introduces D_2 to compare adjacent first-order contrasts. Its second-order term characterizes how the first-order estimate changes across reference levels, naturally capturing refinement information that cannot be represented by a single quality contrast alone. Figure 3 schematically illustrates how this additional contrast can address residual error after first-order guidance. To assess whether this multi-level construction provides information beyond first-order guidance, we measure the cosine similarity between $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ for the second-order extension of CFG, AG, and PG. The results in Appendix Figure 5 show that these directions are generally non-collinear, so multilevel references can provide complementary steering information. Extensions to higher-order reference contrasts are discussed in Appendix B.

3.3 Construction of the Second-order Guidance Term

Reference continuation principle. To capture a fine-grained steering direction, a meaningful second-order refinement should preserve the steering mechanism of the underlying first-order guidance rather than introduce an unrelated auxiliary objective. We therefore construct D_0, D_1, and D_2 by continuing the same degradation trajectory that defines the original guidance method: D_0 is the primary prediction, D_1 is its intermediate reference, and D_2 is a further-degraded reference. All predictions are computed for the same reverse sampling step, while only the method-specific degradation variable is varied. This construction preserves the performance of original first-order guidance when $w_2 = 0$, and uses the additional reference level solely to estimate how the original reference contrast changes along its own degradation trajectory.

CFG (Ho & Salimans, 2021). Here, we construct a progressive condition-ablation trajectory $D_0 = D(\boldsymbol{x}_t, t, \boldsymbol{c}), D_1 = D(\boldsymbol{x}_t, t, \boldsymbol{c}_{\text{vague}}), D_2 = D(\boldsymbol{x}_t, t, \phi)$, where ϕ denotes the null condition and $\boldsymbol{c}_{\text{vague}}$ retains only partial information from $\boldsymbol{c}$, *e.g.*, by removing selected attributes, objects, relations, or contextual details. For CFG, we make a method-specific adjustment to equation 7: the first-order contrast is $\boldsymbol{u}_1 = D_0 - D_2$ rather than $D_0 - D_1$, preserving the standard CFG direction. The second-order contrast remains $\boldsymbol{u}_2 = D_0 - 2D_1 + D_2$, with D_1 as the intermediate reference.

AG (Karras et al., 2024a). It constructs its first-order direction from a well-trained denoiser and an additional degraded version of the denoiser. We extend this trajectory by introducing a further-degraded diffusion model: $D_0 = D_{\text{strong}}, D_1 = D_{\text{weak}}, D_2 = D_{\text{weaker}}$. For example, the three levels can be obtained by progressively reducing the training budget, *e.g.*, $N \to N/4 \to N/16$, or by using nested model capacities, *e.g.*, XL $\to$ L $\to$ M. Since all denoisers follow the same data distribution and conditioning technique, the resulting multi-level contrasts retain the correlated-error structure exploited by AG.

SG (Li et al., 2026). It constructs the reference prediction through a shifted diffusion time step. We continue the same temporal-shift mechanism to obtain $D_0 = D(\boldsymbol{x}_t, t), D_1 = D(\boldsymbol{x}_t, t+\delta), D_2 = D(\boldsymbol{x}_t, t+2\delta)$, where the sampling state $\boldsymbol{x}_t$ is held fixed. The second-order contrast therefore captures the variation of the self-guidance direction as the temporal shift increases.

PG (Chen et al., 2026b). For bridge models, we verify C^2G with recently proposed PG, a perturbation-based guidance method. We iteratively apply the original perturbation operator $\mathcal{H}$ to form an ordered perturbation trajectory: $\boldsymbol{x}_t^{(0)} = \boldsymbol{x}_t, \boldsymbol{x}_t^{(1)} = \mathcal{H}(\boldsymbol{x}_t^{(0)}), \boldsymbol{x}_t^{(2)} = \mathcal{H}(\boldsymbol{x}_t^{(1)})$. The three predictions are then evaluated as $D_k = D(\boldsymbol{x}_t^{(k)}, t, \boldsymbol{c})$ for $k \in \{0, 1, 2\}$. For stochastic perturbations, a fresh random draw is used at each application of $\mathcal{H}$. This construction extends the original contrast between the seen prior and the perturbed prior into a multi-level perturbation trajectory.

3.4 Disentangling First-order and Contrast-of-Contrasts Guidance

Controlled projection. The momentum direction $\boldsymbol{u}_{2,t}$ generally contains a component parallel to the first-order direction $\boldsymbol{u}_{1,t}$, so directly adding it both introduces an additional refinement and changes the effective first-order correction. We estimate their overlap by $\kappa_t := \langle \boldsymbol{u}_{2,t}, \boldsymbol{u}_{1,t} \rangle / (\|\boldsymbol{u}_{1,t}\|_2^2 + \epsilon)$, which gives the fully projected direction $\boldsymbol{v}_{2,t} = \boldsymbol{u}_{2,t} - \kappa_t \boldsymbol{u}_{1,t}$.

Full projection gives the cleanest separation, but can be unnecessarily restrictive. The first-order coefficient is tuned for its standalone trade-off between improvement and guidance-induced error; once contrast-of-contrasts guidance is added, the two terms can instead be optimized jointly. We therefore introduce a projection ratio $c \in [0, 1]$ and define the **controlled projected second-order** direction as $\widetilde{\boldsymbol{u}}_{2,t}(c) = \boldsymbol{u}_{2,t} - c\kappa_t \boldsymbol{u}_{1,t}$, where $c = 0$ retains the raw momentum direction and $c = 1$ gives full projection. The resulting guidance is

$$D_{\text{C}^2\text{G}} = D_0 + w_1 \boldsymbol{u}_{1,t} + w_2 \left(\boldsymbol{u}_{2,t} - \frac{c \langle \boldsymbol{u}_{2,t}, \boldsymbol{u}_{1,t} \rangle}{\|\boldsymbol{u}_{1,t}\|_2^2 + \epsilon} \boldsymbol{u}_{1,t} \right). \tag{12}$$

In our implementation, $\epsilon = 10^{-12}$. The analysis below removes this numerical safeguard and assumes $\|\boldsymbol{u}_{1,t}\|_2 > 0$ almost surely.

An illustrative analysis. For a fixed geometry, raw and projected second-order directions span the same correction space. Their parameterizations differ, however, when one coefficient pair is

Table 1: COCO results on SD3, implemented use (Li et al., 2025) . The CFG column uses our conditional-anchored coefficient $w_1 = 2.5$, equivalent to the conventional scale $s_{\mathrm{CFG}} = 3.5$. In combined settings, $\mathrm{C}^2\mathrm{G}$ and $\mathrm{ProjC}^2\mathrm{G}$ extend only SG. We keep its original sampling and evaluation protocol unchanged and only add the second-order branch required by $\mathrm{C}^2\mathrm{G}$. HPS v2 and CLIP are multiplied by 100. Bold denotes the best value within each comparison block, including ties.

Approach	1st-order			$\mathrm{ProjC}^2\mathrm{G}$		Metrics		
	CFG	SG	PAG	w_2	c	FID↓	HPS↑	CLIP↑
CFG	2.5	–	–	0	–	27.00	30.00	31.69
CFG-$\mathrm{C}^2\mathrm{G}$	2.5	–	–	0.5	0.0	24.97	30.18	31.70
CFG-$\mathrm{ProjC}^2\mathrm{G}$	2.5	–	–	0.5	1.0	**24.73**	**30.23**	**31.72**
CFG+SG	2.5	3.0	–	0	–	25.21	30.32	31.63
CFG+(SG-$\mathrm{C}^2\mathrm{G}$)	2.5	3.0	–	−0.5	0.0	23.96	30.35	**31.66**
CFG+(SG-$\mathrm{ProjC}^2\mathrm{G}$)	2.5	3.0	–	−0.5	0.75	**23.93**	**30.36**	**31.66**
CFG+PAG+SG	2.5	3.0	0.7	0	–	24.83	30.37	31.62
CFG+PAG+(SG-$\mathrm{C}^2\mathrm{G}$)	2.5	3.0	0.7	1.0	0.0	24.59	30.44	31.62
CFG+PAG+(SG-$\mathrm{ProjC}^2\mathrm{G}$)	2.5	3.0	0.7	1.0	1.0	**24.49**	**30.48**	**31.64**

shared across the sample- and time-varying geometries of a sampling trajectory. This effect can be quantified directly for the sample-wise projection used by $\mathrm{ProjC}^2\mathrm{G}$.

Theorem 1 (Shared-coefficient risk under sample-wise projection)**.** *Let the expectation be over both the sampling timestep and sample, and define*

$$R_t := \|\boldsymbol{u}_{1,t}\|_2^2, \qquad \bar{\kappa}_R := \frac{\mathbb{E}[R_t\kappa_t]}{\mathbb{E}[R_t]}, \qquad S := \mathbb{E}\big[R_t(\kappa_t - \bar{\kappa}_R)^2\big], \qquad \bar{V} := \mathbb{E}[\|\boldsymbol{v}_{2,t}\|_2^2]. \tag{13}$$

Assume these moments are finite and $\bar{V} > 0$. For the target correction $\boldsymbol{r}_t = a\boldsymbol{u}_{1,t} + b\boldsymbol{v}_{2,t}$, the minimum risk over globally shared coefficients (w_1, w_2) is

$$\mathcal{E}_c^* = \frac{b^2(1-c)^2 S\bar{V}}{(1-c)^2 S + \bar{V}}. \tag{14}$$

This result uses the same sample-wise coefficient κ_t as $\mathrm{ProjC}^2\mathrm{G}$ and shows that the projection ratio can materially change the shared-coefficient risk. Here $c = 1$ is optimal, while more complex target corrections may prefer different projection strengths. The complete construction and proof are given in Appendix C.2. This theoretical optimization is distinct from our real dataset experiments, where the baseline w_1 is kept fixed.

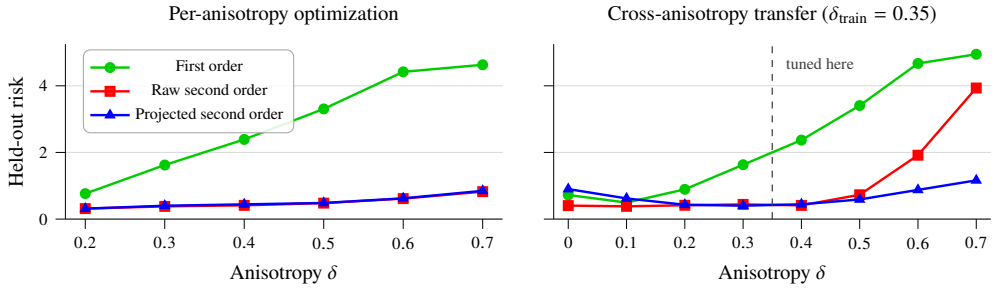


Figure 4: Controlled learned-predictor validation on the two-band anisotropic Gaussian family. **Left:** per-anisotropy optimization. **Right:** cross-anisotropy transfer from $\delta_{\text{train}} = 0.35$.

Controlled example with learned predictors. We illustrate this geometry using three independently trained MLP predictors on a two-band anisotropic Gaussian family. As shown in Figure 4, raw and projected second-order guidance perform similarly under per-anisotropy optimization, while projection is more stable under cross-anisotropy transfer. This example highlights the trade-off of choosing c. Details are provided in Appendix D

Time-varying projection. The preferred projection strength may also vary along the sampling trajectory, since both the first–second-order overlap and the locally required correction change with time. We therefore allow c_t to vary while keeping w_1 and w_2 fixed. The corresponding derivation is provided in Appendix C.4.

Table 2: Results on EDM2/ImageNet-512 at 32 sampling steps using official code(NVIDIA Research, 2024). Each metric is the minimum over three disjoint seed sets; see Appendix E.4.

FID↓						FD-DINOv2↓					
Method	w_1	w_2	c	Old	New	Method	w_1	w_2	c	Old	New
S+XS						*S+XS*					
AG	1.10	–	–	1.390	1.248	AG	1.45	–	–	37.35	35.17
AG-$\mathrm{ProjC^2G}^\dagger$	1.10	-0.1	1.0	1.368	1.234	AG-$\mathrm{ProjC^2G}^\dagger$	1.45	-0.07	0.75	37.07	34.92
AG-$\mathrm{ProjC^2G}$	1.10	-0.1	1.0	**1.359**	**1.225**	AG-$\mathrm{ProjC^2G}$	1.45	-0.05	0.25	**36.83**	**34.78**
XXL+M						*XXL+M*					
AG	1.05	–	–	1.329	1.171	AG	1.45	–	–	25.12	22.91
AG-$\mathrm{ProjC^2G}$	1.05	-0.1	1.0	**1.313**	**1.157**	AG-$\mathrm{ProjC^2G}$	1.45	-0.07	0.25	**24.57**	**22.42**

† Uses an inconsistently constructed second-order branch.

4 EXPERIMENTS

Experimental Setup. Following baseline methods, we evaluate $\mathrm{ProjC^2G}$ across CFG and SG on Stable Diffusion 3, AG with EDM2 on ImageNet-512×512, and PG or FMPG on E2H, DIODE, and image-to-video generation (Wang et al., 2025), covering both diffusion and bridge models. All pretrained models are frozen and $\mathrm{ProjC^2G}$ is applied only at inference time. For each baseline, we keep the backbone, evaluation protocol, and other baseline-specific settings unchanged. We use the reported optimal first-order coefficient w_1 when available, or tune the optimal value with the released implementation otherwise. With w_1 fixed, we tune only w_2 followed by c; search spaces and time-varying schedules are specified in Appendix E.1.

Better Refinement with $\mathrm{ProjC^2G}$. Existing first-order guidance relies on a single low-quality reference to estimate and correct reconstruction error, which can leave residual refinement unexplored. By introducing an additional quality contrast, $\mathrm{ProjC^2G}$ provides a second-order refinement that more faithfully reconstructs the target distribution beyond the first-order correction. Across the evaluated settings, $\mathrm{ProjC^2G}$ improves the most of the metrics like FID, CLIP, and HPS.

CFG, SG on COCO. Following (Li et al., 2026), Table 1 reports COCO (Lin et al., 2014) FID, HPS, and CLIP scores with Stable Diffusion 3 (Esser et al., 2024). We construct the auxiliary CFG branch as detailed in Appendix E.5. $\mathrm{ProjC^2G}$ improves or matches the three reported metrics for CFG and SG, including settings in which SG is combined with PAG.

AG on ImageNet-512. Table 2 reports AG results on 50,000 ImageNet-512 (Deng et al., 2009) samples with EDM2 (Karras et al., 2024b) in 32 sampling steps. We use the original and AlphaFlow class-balanced protocols (Zhang et al., 2026). $\mathrm{ProjC^2G}$ improves FID and FD-DINOv2 across both presets and protocols; employed checkpoints are listed in Appendix E.4.

PG, FMPG on DIODE, E2H. Table 3 evaluates $\mathrm{ProjC^2G}$ with DBIM (Zheng et al., 2025) on DIODE (Vasiljevic et al., 2019) and Edges2Handbags (Isola et al., 2017). $\mathrm{ProjC^2G}$ yields overall improvements across the PG or FMPG settings; full settings are provided in Appendix E.2.

CFG+PG on FrameBridge. Table 4 evaluates image-to-video method FrameBridge (Wang et al., 2025) on VBench (Huang et al., 2024), where both CFG and PG are employed to achieve the best performance of PG or FMPG methods (Chen et al., 2026b). $\mathrm{ProjC^2G}$ improves the primary component scores under fixed CFG+PG settings.

Generalization and ablations. The gains span CFG, SG, AG, and PG as well as diffusion- and bridge-based tasks (Tables 1–4), although some secondary metrics remain unchanged or slightly deteriorate. Table 1 compares $\mathrm{ProjC^2G}$ ($c \neq 0$) with $\mathrm{C^2G}$ ($c = 0$), and Table 5 compares static and time-varying projection; the latter improves FID in most settings, with schedules given in Appendix E.1. The †-marked EDM2 rows further show that an approximately even degradation progression is preferable to the tested nonuniform second-order branch.

Table 3: Full-evaluation results on E2H and DIODE. FMPG denotes the frequency-dependent variant of PG. Each block compares PG or FMPG with its $\mathrm{ProjC}^2\mathrm{G}$ extension; guidance parameters and schedules are omitted for compactness. Bold denotes the better value within each pair.

Method	Edges→Handbags					DIODE-outdoor				
	Steps	FID↓	IS↑	LPIPS↓	MSE↓	Steps	FID↓	IS↑	LPIPS↓	MSE↓
PG	10	1.43	3.60	0.11	0.007	10	3.76	6.13	0.24	0.033
PG-$\mathrm{ProjC}^2\mathrm{G}$	10	**1.35**	**3.61**	0.11	0.007	10	**3.71**	**6.18**	0.24	**0.028**
PG	50	0.91	3.58	0.11	0.006	50	2.08	6.05	0.20	0.019
PG-$\mathrm{ProjC}^2\mathrm{G}$	50	**0.78**	**3.60**	0.11	0.006	50	**2.06**	**6.14**	0.20	0.019
PG	3	4.57	3.71	**0.09**	**0.007**	5	6.20	6.22	0.25	0.053
PG-$\mathrm{ProjC}^2\mathrm{G}$	3	**3.77**	**3.73**	0.10	0.010	5	**6.05**	**6.23**	0.25	**0.050**
FMPG	10	1.08	3.63	0.11	0.008	10	3.43	6.05	0.24	0.034
FMPG-$\mathrm{ProjC}^2\mathrm{G}$	10	**1.03**	**3.64**	0.11	0.008	10	**3.30**	**6.07**	**0.23**	**0.031**

Table 4: VBench-I2V scores (official normalization). See Appendix E.3 for settings and metrics; our implementation will be released. Bold: best in each column.

Model / Setting	Total	I2V	Quality	I2V-SC	MS	DD
SEINE-512×320	83.49	89.62	77.37	93.97	89.47	34.31
DynamiCrafter-256	84.35	91.29	77.42	94.61	93.38	**38.69**
FrameBridge-VideoCrafter	85.37	92.83	77.92	95.60	95.75	35.77
PG (blur)	85.71	93.20	78.22	96.34	97.11	31.46
PG-$\mathrm{ProjC}^2\mathrm{G}$ (blur)	**85.83**	93.23	**78.43**	96.34	97.12	33.90
PG (pooling)	85.74	93.23	78.24	96.41	97.12	30.16
PG-$\mathrm{ProjC}^2\mathrm{G}$ (pooling)	85.78	**93.32**	78.23	**96.45**	**97.16**	30.56

Table 5: Static vs. time-varying projection (FID↓).

Benchmark	Guidance	Steps	Static	Dynamic
E2H	PG	10	1.35	**1.31**
E2H	PG	50	0.78	**0.73**
DIODE	PG	5	6.05	**5.80**
DIODE	PG	50	2.06	**2.02**
SD3	CFG	28	24.73	**24.65**
SD3	CFG+SG	28	**23.93**	**23.93**
SD3	CFG+PAG+SG	28	24.49	**24.47**

5 Related Work

Guidance methods have developed different denoising references to emphasize condition alignment (Ho & Salimans, 2021; Sun et al., 2024), score accuracy (Karras et al., 2024a), fine-grained details (Park et al., 2025; Li et al., 2026), or prior exploitation (Chen et al., 2026b). Furthermore, methods are developed to improve guidance performance by combining heterogeneous signals (gil Lee et al., 2025; Wang et al., 2026), extending CFG through Taylor expansion over continuous condition representations (Sun et al., 2024), exploring temporal effects along the diffusion sampling trajectory (Liao et al., 2026), or leveraging negative concept (Koulischer et al., 2025; Chang et al., 2026). Concurrently, several works propose time-varying (Kynkäänniemi et al., 2024; Jiang et al., 2026), learnable (Galashov et al., 2026), or frequency-dependent (Sadat et al., 2025b) guidance scales. In this work, we inherit the advantages of existing effective guidance mechanisms and present a general second-order guidance framework by constructing multiple reference-quality levels, which demonstrate stable improvements across a range of guidance methods and generation tasks. More details of related work are introduced in Appendix A.

6 Conclusion

In this work, we introduce $\mathrm{C}^2\mathrm{G}$, a general second-order framework that improves guidance steering through multi-level reference contrasts. By characterizing variations in first-order steering directions along their underlying degradation trajectories, $\mathrm{C}^2\mathrm{G}$ provides complementary refinement signals beyond single-level guidance. We further propose $\mathrm{ProjC}^2\mathrm{G}$, which preserves tuned first-order guidance scale while dynamically modulating the parallel and orthogonal components of second order corrections, enabling stable higher-order refinement throughout sampling. Experiments across diverse guidance mechanisms spanning diffusion and bridge models and multiple generative tasks demonstrate consistent gains without requiring new steering objectives, establishing $\mathrm{C}^2\mathrm{G}$ as an effective and broadly applicable framework for second-order steering. ProjMG nevertheless adds computation and memory. Future efforts include distilling its three branches into a single denoiser.

AI USE STATEMENT

We used generative AI tools in two ways. First, they assisted with precise word choice, grammar correction, and minor language polishing to improve the readability of the manuscript. Second, Qwen2.5-7B-Instruct was used as an explicit component of our experimental preprocessing pipeline to generate deliberately weakened auxiliary captions for the CFG-based $\mathrm{C}^2\mathrm{G}$ and $\mathrm{ProjC}^2\mathrm{G}$ experiments. The model, prompt, decoding configuration, and use of the resulting captions are detailed in Appendix E.5. Generative AI tools were not used to formulate mathematical claims or proofs, interpret experimental results, or draw research conclusions. All AI-assisted manuscript edits were reviewed by the authors. We take responsibility for the final content of this work, including artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The $\mathrm{C}^2\mathrm{G}$ and $\mathrm{ProjC}^2\mathrm{G}$ formulations are specified in the main paper. Complete derivations and proofs, including the terminal-marginal interpretation, higher-order construction, sample-wise projection analysis, and time-varying projection analysis, are provided in Appendices A.3–C.4. Appendix E reports the hyperparameter search procedure, exact guidance coefficients and schedules, pretrained checkpoints, dataset sizes, evaluation protocols, and random-seed aggregation used in the experiments. The datasets and pretrained models are identified by their citations, with exact checkpoint information provided where applicable. Complete implementation will be fully open-sourced.

REFERENCES

Donghoon Ahn, Hyoungwon Cho, Jaewon Min, Wooseok Jang, Jungwoo Kim, SeonHwa Kim, Hyun Hee Park, Kyong Hwan Jin, and Seungryong Kim. Self-rectifying diffusion sampling with perturbed-attention guidance. In *ECCV*, 2024.

Arwen Bradley and Preetum Nakkiran. Classifier-free guidance is a predictor-corrector. *Transactions on Machine Learning Research*, 2025.

Jeffrey A. Chan Santiago, Praveen Tirupattur, Gaurav Kumar Nayak, Gaowen Liu, and Mubarak Shah. MGD^3: Mode-guided dataset distillation using diffusion models. In *ICML*, 2025.

Jinho Chang, Changsun Lee, Hyungjin Chung, and Jong Chul Ye. Contrastivecfg: Guiding diffusion sampling by contrasting positive and negative concepts. In *ICML*, 2026.

Chubin Chen, Jiashu Zhu, Xiaokun Feng, Nisha Huang, Chen Zhu, Meiqi Wu, Fangyuan Mao, Jiahong Wu, Xiangxiang Chu, and Xiu Li. s^2-guidance: Stochastic self-guidance for training-free enhancement of diffusion models. In *ICLR*, 2026a.

Huayu Chen, Kai Jiang, Kaiwen Zheng, Jianfei Chen, Hang Su, and Jun Zhu. Visual generation without guidance. In *ICML*, 2025.

Zehua Chen, Yucheng Yang, Binjie Yuan, Kaiwen Zheng, Jun S Liu, and Jun Zhu. Guidedbridge: Training-freely improving bridge models with prior guidance. In *ICML*, 2026b.

Hyungjin Chung, Jeongsol Kim, Geon Yeong Park, Hyelin Nam, and Jong Chul Ye. Cfg++: Manifold-constrained classifier free guidance for diffusion models. In *ICLR*, 2025.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

Alexandre Galashov, Ashwini Pokle, Arnaud Doucet, Arthur Gretton, Mauricio Delbracio, and Valentin De Bortoli. Learn to guide your diffusion model. In *ICLR*, 2026.

Jiayang Gao, Tianyi Zheng, Jiayang Zou, Fengxiang Yang, Shice Liu, Luyao Fan, Zheyu Zhang, Hao Zhang, Jinwei Chen, Peng-Tao Jiang, Bo Li, and Jia Wang. C^2fg: Control classifier-free guidance via score discrepancy analysis. In *CVPR*, 2026.

Zheng Gao, Jifei Song, Zhensong Zhang, Jiankang Deng, and Ioannis Patras. Frequency-guided diffusion for training-free text-driven image translation. In *ICCV*, 2025.

Sang gil Lee, Zhifeng Kong, Arushi Goel, Sungwon Kim, Rafael Valle, and Bryan Catanzaro. Etta: Elucidating the design space of text-to-audio models. In *ICML*, 2025.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1125–1134, 2017.

Yuxuan Jiang, Zehua Chen, Zeqian Ju, Yusheng Dai, Weibei Dou, and Jun Zhu. Controlaudio: Tackling text-guided, timing-indicated and intelligible audio generation via progressive diffusion modeling. In *ACL*, 2026.

Tero Karras, Miika Aittala, Tuomas Kynkäänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine. Guiding a diffusion model with a bad version of itself. In *NeurIPS*, 2024a.

Tero Karras, Miika Aittala, Jaakko Lehtinen, Janne Hellsten, Timo Aila, and Samuli Laine. Analyzing and improving the training dynamics of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024b.

Yeongmin Kim, Donghyeok Shin, Byeonghu Na, Minsang Park, Richard Lee Kim, and Il-Chul Moon. Lookahead sample reward guidance for test-time scaling of diffusion models. In *ICML*, 2026.

Felix Koulischer, Johannes Deleu, Gabriel Raya, Thomas Demeester, and Luca Ambrogioni. Dynamic negative guidance of diffusion models. In *ICLR*, 2025.

Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. In *NeurIPS*, 2024.

Tiancheng Li, Weijian Luo, Zhiyang Chen, Liyuan Ma, and Guo-Jun Qi. Self-guidance: Official code repository. `https://github.com/maple-research-lab/Self-Guidance`, 2025. GitHub repository.

Tiancheng Li, Weijian Luo, Zhiyang Chen, Liyuan Ma, and Guo-Jun Qi. Self-guidance: Boosting flow and diffusion generation on their own. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

Runlong Liao, Jian Yu, Baiyu Su, Chi Zhang, Lizhang Chen, and Qiang Liu. Momentum guidance: Plug-and-play guidance for flow models. *CoRR*, abs/2602.20360, 2026.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and Lawrence C. Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pp. 740–755, 2014.

Guan-Horng Liu, Arash Vahdat, De-An Huang, Evangelos A Theodorou, Weili Nie, and Anima Anandkumar. I^2SB: Image-to-image Schrödinger bridge. In *ICML*, 2023.

NVIDIA Research. Edm2 and autoguidance: Official pytorch implementation. `https://github.com/NVlabs/edm2`, 2024. GitHub repository.

Sunghyun Park, Seokeon Choi, Hyoungwoo Park, and Sungrack Yun. Steering guidance for personalized text-to-image diffusion models. In *ICCV*, 2025.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *ICLR*, 2024.

Hannes Risken. *The Fokker–Planck Equation: Methods of Solution and Applications*. Springer, 1996.

Seyedmorteza Sadat, Otmar Hilliges, and Romann M. Weber. Eliminating oversaturation and artifacts of high guidance scales in diffusion models. In *ICLR*, 2025a.

Seyedmorteza Sadat, Tobias Vontobel, Farnood Salehi, and Romann M. Weber. Guidance in the frequency domain enables high-fidelity sampling at low cfg scales, 2025b.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *ICLR*, 2022.

Jascha Sohl-Dickstein, Eric A Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021.

Shikun Sun, Longhui Wei, Zhicai Wang, Zixuan Wang, Junliang Xing, Jia Jia, and Qi Tian. Inner classifier-free guidance and its taylor expansion for diffusion models. In *ICLR*, 2024.

Igor Vasiljevic, Nick Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z. Dai, Andrea F. Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R. Walter, and Gregory Shakhnarovich. DIODE: A dense indoor and outdoor depth dataset. *arXiv preprint arXiv:1908.00463*, 2019.

Junyou Wang, Zehua Chen, Binjie Yuan, Kaiwen Zheng, Chang Li, Yuxuan Jiang, and Jun Zhu. Audiomog: Guiding audio generation with mixture-of-guidance. In *ICME*, 2026.

Yuji Wang, Zehua Chen, Xiaoyu Chen, Yixiang Wei, Jun Zhu, and Jianfei Chen. Framebridge: Improving image-to-video generation with bridge models. In *ICML*, 2025.

Xuehui Yu, Fucheng Cai, Meiyi Wang, Xiaopeng Fan, and Harold Soh. Conflict-aware additive guidance for flow models under compositional rewards. In *ICML*, 2026.

Huijie Zhang, Aliaksandr Siarohin, Willi Menapace, Michael Vasilkovsky, Sergey Tulyakov, Qing Qu, and Ivan Skorokhodov. Alphaflow: Understanding and improving meanflow models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL `https://openreview.net/forum?id=adacb4JTIv`.

Candi Zheng and Yuan Lan. Characteristic guidance: Non-linear correction for diffusion model at large guidance scale. In *ICML*, 2024.

Kaiwen Zheng, Guande He, Jianfei Chen, Fan Bao, and Jun Zhu. Diffusion bridge implicit models. In *ICLR*, 2025.

Linqi Zhou, Aaron Lou, Samar Khanna, and Stefano Ermon. Denoising diffusion bridge models. In *ICLR*, 2024.

A GUIDANCE METHODS

A.1 BASELINE METHODS USED IN OUR EXPERIMENTS

Auto-guidance (Karras et al., 2024a) mentions that CFG enhances condition alignment but may sacrifice sampling diversity, thereby suffering from mode collapse. Therefore, it aims to enhance score accuracy. Specifically, it additionally develops a *weak* diffusion model $D_{\boldsymbol{\theta}_{\text{bad}}}(\boldsymbol{x}_t, t, \boldsymbol{c})$ for the same task and conditioning information, which can make similar but larger errors than the pretrained *good* diffusion model $D_{\boldsymbol{\theta}_{\text{good}}}(\boldsymbol{x}_t, t, \boldsymbol{c})$. Practically, the weak model can be constructed with limited model capacity or fewer training steps. By contrasting the low-quality and high-quality denoising results from the weak and good diffusion models, respectively, AG enhances score accuracy as follows.

$$D_{\text{AG}}(\boldsymbol{x}_t, t, \boldsymbol{c}) = D_{\boldsymbol{\theta}_{\text{good}}}(\boldsymbol{x}_t, t, \boldsymbol{c}) + w_{\text{AG}} \left(D_{\boldsymbol{\theta}_{\text{good}}}(\boldsymbol{x}_t, t, \boldsymbol{c}) - D_{\boldsymbol{\theta}_{\text{bad}}}(\boldsymbol{x}_t, t, \boldsymbol{c}) \right). \tag{15}$$

Prior guidance (Chen et al., 2026b) is a recently proposed guidance method for bridge models that encourages prior exploitation without additional training to further boost the performance of bridge models in data-to-data generation tasks. In detail, it perturbs the prior information at $t = T$ with manually designed degradation methods, such as adding noise or blurring, to hinder prior exploitation and thereby decrease the denoising quality of pretrained bridge models. By contrasting the denoising results from the normal prior seen during pretraining and a perturbed prior unseen during pretraining, PG further encourages prior exploitation as follows.

$$D_{\text{PG}}(\boldsymbol{x}_t, t, \boldsymbol{c}) = D_{\boldsymbol{\theta}_{\text{normal}}}(\boldsymbol{x}_t, t, \boldsymbol{c}) + w_{\text{PG}} \left(D_{\boldsymbol{\theta}_{\text{normal}}}(\boldsymbol{x}_t, t, \boldsymbol{c}) - D_{\boldsymbol{\theta}_{\text{perturbed}}}(\mathcal{H}(\boldsymbol{x}_t), t, \boldsymbol{c}) \right). \tag{16}$$

Self-guidance (Li et al., 2026) is a training-free guidance method that exploits the prediction difference of the same pretrained diffusion model across different noise levels. Specifically, given the current state $\boldsymbol{x}_t$, SG additionally evaluates the same model at a shifted and noisier timestep $t + \delta(t)$ while keeping the model input and condition information unchanged. The prediction at the shifted timestep provides a less-refined reference associated with a higher noise level. By contrasting the predictions at the current and shifted timesteps, SG suppresses low-quality generation and artifacts without introducing an additional model. It can be formulated as

$$D_{\text{SG}}(\boldsymbol{x}_t, t, \boldsymbol{c}) = D_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t, \boldsymbol{c}) + w_{\text{SG}} \left(D_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t, \boldsymbol{c}) - D_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t + \delta(t), \boldsymbol{c}) \right). \tag{17}$$

Classifier-free guidance (Ho & Salimans, 2021) extrapolates two denoising terms learned by the same diffusion model: a *conditional* one $D_{\boldsymbol{\theta}_{\text{diff}}}(\boldsymbol{x}_t, t, \boldsymbol{c})$ and an *unconditional* one $D_{\boldsymbol{\theta}_{\text{diff}}}(\boldsymbol{x}_t, t)$. At the training stage, the conditional one costs more training resources and is easier to learn, while the unconditional one costs a small training ratio and is more difficult to model, resulting in the high-quality and low-quality denoising result, respectively. By contrasting these two denoising terms with a scale factor w_{CFG}, CFG enhances condition alignment as follows.

$$D_{\text{CFG}}(\boldsymbol{x}_t, t, \boldsymbol{c}) = D_{\boldsymbol{\theta}_{\text{c}}}(\boldsymbol{x}_t, t, \boldsymbol{c}) + w_{\text{CFG}} \left(D_{\boldsymbol{\theta}_{\text{c}}}(\boldsymbol{x}_t, t, \boldsymbol{c}) - D_{\boldsymbol{\theta}_{\text{uc}}}(\boldsymbol{x}_t, t, \phi) \right). \tag{18}$$

Here w_{CFG} follows the conditional-prediction-anchored convention used throughout this paper. If CFG is instead written as $D_{\text{uncond}} + s_{\text{CFG}}(D_{\text{cond}} - D_{\text{uncond}})$, then $s_{\text{CFG}} = 1 + w_{\text{CFG}}$.

A.2 OTHER GUIDANCE METHODS

S^2-Guidance (**S^{2}G**, Chen et al., 2026a) constructs degraded reference predictions through stochastic block dropping within the pretrained model. At each sampling timestep, it randomly drops a subset of network blocks to instantiate a stochastic sub-network while retaining important components of the original architecture.

Characteristic Guidance (**CHG**, Zheng & Lan, 2024) approaches guidance from a different perspective by correcting the dynamical inconsistency of linear classifier-free guidance at large guidance scales. Under its characteristic-based approximation, it constructs a nonlinear correction such that the resulting guided score is consistent with the score Fokker–Planck equation (Risken, 1996) of the underlying diffusion process.

Inner Classifier-Free Guidance (**ICFG**, Sun et al., 2024) interprets CFG through a structured continuous condition space, under which standard CFG corresponds to a first-order Taylor approximation.

It further derives a second-order extension through higher-order variations in the condition representation, rather than through differences across multiple reference-quality levels as considered in our framework.

Momentum Guidance (**MG**, Liao et al., 2026) exploits temporal information along the sampling trajectory. It maintains an exponential moving average of past model velocities and extrapolates the current velocity away from this historical reference, providing guidance without additional model evaluations.

AudioMoG (**MoG**, Wang et al., 2026) combines heterogeneous guidance signals by nesting CFG with AG. It contrasts guided predictions from stronger and weaker models, thereby jointly exploiting conditional alignment and model-quality differences.

Personalization Guidance (Park et al., 2025) balances subject fidelity and text alignment in personalized diffusion models. It constructs a weak personalized reference through interpolation between the pre-trained and fine-tuned models, and steers the fine-tuned model away from this reference during sampling.

Lookahead Sample Reward Guidance (**LiDAR**, Kim et al., 2026) introduces reward guidance through short lookahead trajectories. It evaluates several denoised lookahead samples with a reward function and uses their reward-reweighted distribution to construct a closed-form guidance direction without neural backpropagation.

Predictor-Corrector Guidance (**PCG**, Bradley & Nakkiran, 2025) provides a theoretical interpretation of CFG as alternating denoising and sharpening dynamics. In the SDE limit, it combines a DDIM predictor for the conditional distribution with a Langevin corrector targeting a sharpened γ-powered distribution.

Perturbed-attention guidance (**PAG**, Ahn et al., 2024, Bradley & Nakkiran, 2025) constructs a degraded prediction by perturbing the internal self-attention modules of the pretrained diffusion model. Specifically, PAG replaces selected self-attention maps with identity matrices, thereby suppressing the structural information captured by these modules and producing a prediction with degraded sample structure.

Conflict-Aware Additive Guidance (**CAR**, Yu et al., 2026) addresses off-manifold drift when multiple reward gradients conflict. It dynamically detects gradient misalignment and interpolates between naive additive guidance and a learned terminal-value guidance direction to reduce compositional conflicts.

Dynamic Negative Guidance (**DNG**, Koulischer et al., 2025) improves negative prompting by adapting the guidance strength to the current sample and timestep. It estimates the posterior probability of the negative concept during sampling and weakens or deactivates the repulsive guidance when that concept is unlikely to be present.

Contrastive CFG (**CCFG**, Chang et al., 2026) reformulates positive and negative prompting through a contrastive objective. Its negative-guidance strength decays as the sample moves away from the undesired concept, reducing the excessive repulsion induced by fixed negative CFG.

Mode Guidance (**MGD**^{3}, Chan Santiago et al., 2025) steers diffusion samples toward representative modes discovered from the training data. At each guided step, it adds a direction from the current clean estimate toward a selected mode, and later switches back to unguided denoising to reduce artifacts.

Frequency-Guided Diffusion (**FGD**, Gao et al., 2025) performs training-free text-driven image translation by decomposing the diffusion representation into low- and high-frequency components. The low-frequency part mainly preserves global appearance and style, whereas the high-frequency part captures edges and local structure. It then recombines source and sampling components differently for style-guided and structure-guided translation.

Adaptive Projected Guidance (**APG**, Sadat et al., 2025a) mitigates oversaturation and artifacts at large CFG scales by decomposing the CFG update into components parallel and orthogonal to the conditional prediction. It suppresses the parallel component while retaining the orthogonal component for semantic steering.

CFG++ (**CFG++**, Chung et al., 2025) reformulates text guidance as a manifold-constrained inverse problem to mitigate the off-manifold behavior of standard CFG. It enables effective guidance at smaller scales and improves sampling stability and invertibility.

Control Classifier-Free Guidance (**C^{2}FG**, Gao et al., 2026) analyzes the score discrepancy between conditional and unconditional distributions across timesteps and uses this result to motivate time-dependent guidance. It adapts the CFG strength through an exponential-decay control function that follows the diffusion dynamics.

A.3 Terminal-Marginal Interpretation

Let $p_{\boldsymbol{\theta}}(\boldsymbol{x}_{0:T-1} \mid \boldsymbol{x}_T)$ denote the reverse-path distribution. Its terminal marginal is

$$q_{\boldsymbol{\theta}}(\boldsymbol{x}_0) := \int p_T(\boldsymbol{x}_T) \left[\int p_{\boldsymbol{\theta}}(\boldsymbol{x}_{0:T-1} \mid \boldsymbol{x}_T)\, \mathrm{d}\boldsymbol{x}_{1:T-1} \right] \mathrm{d}\boldsymbol{x}_T. \tag{19}$$

Replacing the reverse transitions by their guided counterparts analogously induces $q_{\boldsymbol{\theta}}^{\mathrm{guide}}(\boldsymbol{x}_0)$. Both $q_{\boldsymbol{\theta}}^{\mathrm{guide}}$ and p_{target} are therefore distributions on the terminal variable $\boldsymbol{x}_0$. In the CFG discussion, the tilted density in the main text specifies the desired terminal distribution. Any fixed guided sampler induces a terminal marginal, but this interpretation does not require the same-state linear combination to be the exact score evolution of that target density at every intermediate time.

B Higher-Order Guidance

Higher-order refinement. The same refinement principle extends naturally beyond second order. If the second-order refinement is still insufficient, one can further refine how the second-order direction changes across degradation levels, and continue this process recursively. We separate the first-order direction inherited from the underlying guidance method from the recursively constructed higher-order terms. Let $\boldsymbol{g}_1$ denote this inherited direction: $\boldsymbol{g}_1 = D_0 - D_1$ for the adjacent-reference construction, whereas the evaluated CFG instance uses $\boldsymbol{g}_1 = D_0 - D_2$ to preserve standard CFG. Given a progressive sequence $D_0, D_1, \ldots, D_K$, the k-th order reference difference is

$$\boldsymbol{u}_k = \sum_{j=0}^{k} (-1)^j \binom{k}{j} D_j, \qquad k = 2, \ldots, K. \tag{20}$$

The closed form in eq. (20) follows directly from the recursive refinement construction; its derivation is given in Section B.1. The corresponding raw K-th order guidance is

$$D_{\mathrm{HOG}}^{(K)} = D_0 + w_1 \boldsymbol{g}_1 + \sum_{k=2}^{K} w_k \boldsymbol{u}_k. \tag{21}$$

For $K = 2$, $\boldsymbol{u}_2 = D_0 - 2D_1 + D_2$, so eq. (21) exactly recovers the main-text formulation, including its CFG-specific first-order direction. Each additional order can therefore be viewed as refining the previous reference difference using one more degradation level. In this work, we focus on $K = 2$, since it is the simplest extension beyond standard first-order guidance and requires only one additional prediction.

Directional complementarity. The first- and second-order directions are not generally collinear in the evaluated CFG, AG, and PG settings, as shown in Figure 5.

B.1 Recursive Construction and Closed-Form Expression

We derive the closed-form expression in eq. (20) for the higher-order terms. This recursion does not redefine the method-specific first-order direction $\boldsymbol{g}_1$. For a sequence of predictions $D_0, D_1, \ldots$, define the adjacent first differences

$$\boldsymbol{u}_1^{(j)} = D_j - D_{j+1}, \tag{22}$$

and recursively define each higher-order refinement by

$$\boldsymbol{u}_{k+1}^{(j)} = \boldsymbol{u}_k^{(j)} - \boldsymbol{u}_k^{(j+1)}. \tag{23}$$

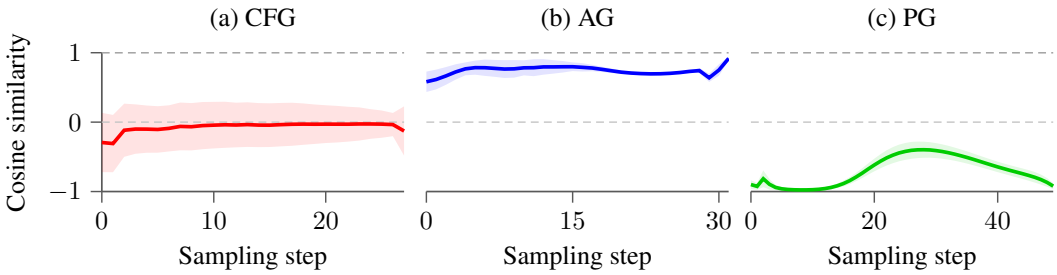


Figure 5: Cosine similarity between first- and second-order guidance directions across sampling steps. Solid lines show the mean over 128 samples at each step; shading denotes one standard deviation. CFG follows the official Self-Guidance codebase, while AG and PG follow their respective official codebases, as cited in the experimental section of the main text.

For each higher order $k \geq 2$, the direction above is $\boldsymbol{u}_k = \boldsymbol{u}_k^{(0)}$.

Lemma 1. *For every* $k \geq 1$,

$$\boldsymbol{u}_k^{(j)} = \sum_{r=0}^{k} (-1)^r \binom{k}{r} D_{j+r}. \tag{24}$$

Proof. The claim holds for $k = 1$, since

$$\boldsymbol{u}_1^{(j)} = D_j - D_{j+1} = \sum_{r=0}^{1} (-1)^r \binom{1}{r} D_{j+r}.$$

Assume that it holds for order k. Using eq. (23),

$$\boldsymbol{u}_{k+1}^{(j)} = \boldsymbol{u}_k^{(j)} - \boldsymbol{u}_k^{(j+1)} \tag{25}$$

$$= \sum_{r=0}^{k} (-1)^r \binom{k}{r} D_{j+r} - \sum_{r=0}^{k} (-1)^r \binom{k}{r} D_{j+r+1}. \tag{26}$$

Reindexing the second sum and applying Pascal's identity

$$\binom{k}{r} + \binom{k}{r-1} = \binom{k+1}{r},$$

gives

$$\boldsymbol{u}_{k+1}^{(j)} = \sum_{r=0}^{k+1} (-1)^r \binom{k+1}{r} D_{j+r}.$$

Therefore, the result follows by induction. Setting $j = 0$ yields, for $k \geq 2$,

$$\boldsymbol{u}_k = \sum_{j=0}^{k} (-1)^j \binom{k}{j} D_j,$$

which is exactly eq. (20). In particular, $\boldsymbol{u}_2 = D_0 - 2D_1 + D_2$ without replacing the CFG baseline $\boldsymbol{g}_1 = D_0 - D_2$. □

C Projection Theory for Sample-wise ProjC^{2}G

C.1 Sample-wise Projection Coefficient

To match the practical ProjC^{2}G construction, the analysis uses the sample-wise projection coefficient. The implementation computes the squared norm in FP32 and uses $\max\{\|\boldsymbol{u}_{1,t}\|_2^2, 10^{-12}\}$ in

the denominator. For the analysis, we remove this numerical safeguard and assume $\|\boldsymbol{u}_{1,t}\|_2 > 0$ almost surely. For each timestep and sample, define

$$\kappa_t := \frac{\boldsymbol{u}_{2,t}^\top \boldsymbol{u}_{1,t}}{\|\boldsymbol{u}_{1,t}\|_2^2}, \qquad \boldsymbol{v}_{2,t} := \boldsymbol{u}_{2,t} - \kappa_t \boldsymbol{u}_{1,t}. \tag{27}$$

The residual is orthogonal to the first-order direction for every sample:

$$\boldsymbol{v}_{2,t}^\top \boldsymbol{u}_{1,t} = \boldsymbol{u}_{2,t}^\top \boldsymbol{u}_{1,t} - \kappa_t \|\boldsymbol{u}_{1,t}\|_2^2 = 0. \tag{28}$$

All expectations below are therefore taken jointly over the sampling timestep and sample, rather than replacing the sample-wise coefficient by a ratio of population expectations.

C.2 Minimum Shared-Coefficient Excess Risk

Proof of Theorem 1. Consider the target correction

$$\boldsymbol{r}_t = a\boldsymbol{u}_{1,t} + b\boldsymbol{v}_{2,t}, \tag{29}$$

where a and b are fixed over the trajectory and across samples. For a projection ratio $c \in [0, 1]$, the second basis direction is

$$\widetilde{\boldsymbol{u}}_{2,t}(c) = \boldsymbol{u}_{2,t} - c\kappa_t \boldsymbol{u}_{1,t} = (1-c)\kappa_t \boldsymbol{u}_{1,t} + \boldsymbol{v}_{2,t}. \tag{30}$$

The correction error produced by a shared pair (w_1, w_2) is

$$\begin{aligned} \boldsymbol{r}_t - w_1\boldsymbol{u}_{1,t} - w_2\widetilde{\boldsymbol{u}}_{2,t}(c) = {} & \left[a - w_1 - (1-c)\kappa_t w_2\right]\boldsymbol{u}_{1,t} && (31)\\ & + (b - w_2)\boldsymbol{v}_{2,t}. && (32) \end{aligned}$$

Let $R_t := \|\boldsymbol{u}_{1,t}\|_2^2$ and $\bar{V} := \mathbb{E}[\|\boldsymbol{v}_{2,t}\|_2^2]$. Pointwise orthogonality removes the cross term before taking expectations, giving

$$\mathcal{E}_c(w_1, w_2) = \mathbb{E}\Big[R_t\big(w_1 + (1-c)\kappa_t w_2 - a\big)^2\Big] + \bar{V}(w_2 - b)^2. \tag{33}$$

Define

$$\bar{\kappa}_R := \frac{\mathbb{E}[R_t\kappa_t]}{\mathbb{E}[R_t]}, \qquad S := \mathbb{E}\big[R_t(\kappa_t - \bar{\kappa}_R)^2\big]. \tag{34}$$

The weighted bias–variance decomposition gives

$$\mathcal{E}_c(w_1, w_2) = \mathbb{E}[R_t]\big[w_1 + (1-c)\bar{\kappa}_R w_2 - a\big]^2 + (1-c)^2 S w_2^2 + \bar{V}(w_2 - b)^2. \tag{35}$$

For fixed w_2, the first term is minimized by

$$w_1^*(c) = a - (1-c)\bar{\kappa}_R w_2. \tag{36}$$

Substitution leaves the one-variable quadratic

$$\mathcal{E}_c(w_1^*(c), w_2) = (1-c)^2 S w_2^2 + \bar{V}(w_2 - b)^2. \tag{37}$$

Setting its derivative with respect to w_2 to zero gives

$$w_2^*(c) = \frac{b\bar{V}}{(1-c)^2 S + \bar{V}}, \qquad w_1^*(c) = a - (1-c)\bar{\kappa}_R \frac{b\bar{V}}{(1-c)^2 S + \bar{V}}. \tag{38}$$

Substituting these coefficients back yields the minimum excess risk

$$\mathcal{E}_c^* = \frac{b^2(1-c)^2 S\bar{V}}{(1-c)^2 S + \bar{V}}. \tag{39}$$

This is an excess risk because coefficients adapted to each sample and timestep can represent the target exactly: $w_2(t, \boldsymbol{x}_t) = b$ and $w_1(t, \boldsymbol{x}_t) = a - (1-c)b\kappa_t$. At $c = 0$, the formulas reduce to the raw row of Table 6. At $c = 1$, they give $(w_1^*, w_2^*) = (a, b)$ and $\mathcal{E}_1^* = 0$, which is the fully projected row. The middle row follows directly for $0 < c < 1$. □

Table 6: Optimal globally shared coefficients and excess risk under sample-wise projection.

Parameterization	w_1^*	w_2^*	c	Excess risk
Raw $\{\boldsymbol{u}_1, \boldsymbol{u}_2\}$	$a - \bar{\kappa}_R \frac{b\bar{V}}{S+\bar{V}}$	$\frac{b\bar{V}}{S+\bar{V}}$	0	$\frac{b^2 S\bar{V}}{S+\bar{V}}$
Partially projected $\{\boldsymbol{u}_1, \widetilde{\boldsymbol{u}}_2(c)\}$	$a - (1-c)\bar{\kappa}_R \frac{b\bar{V}}{(1-c)^2 S+\bar{V}}$	$\frac{b\bar{V}}{(1-c)^2 S+\bar{V}}$	$0 < c < 1$	$\frac{b^2(1-c)^2 S\bar{V}}{(1-c)^2 S+\bar{V}}$
Projected $\{\boldsymbol{u}_1, \boldsymbol{v}_2\}$	a	b	1	0

C.3 Population Geometry of the Anisotropic Gaussian Illustration

For the closed-form diagnostic used in the controlled experiment, let

$$\boldsymbol{x}_t \sim \mathcal{N}(\boldsymbol{0}, s\boldsymbol{A}_t), \qquad \boldsymbol{A}_t \succ 0, \qquad \operatorname{tr}(\boldsymbol{A}_t) = d, \tag{40}$$

with $\boldsymbol{u}_{1,t} = \boldsymbol{x}_t$ and $\boldsymbol{u}_{2,t} = (\boldsymbol{I} - \boldsymbol{A}_t)\boldsymbol{x}_t$. Since $\mathbb{E}[\boldsymbol{x}_t \boldsymbol{x}_t^\top] = s\boldsymbol{A}_t$,

$$\mathbb{E}[\boldsymbol{u}_{2,t}^\top \boldsymbol{u}_{1,t}] = s\left[d - \operatorname{tr}(\boldsymbol{A}_t^2)\right], \tag{41}$$

$$\mathbb{E}[\|\boldsymbol{u}_{1,t}\|_2^2] = sd. \tag{42}$$

Writing $\mu_t := \operatorname{tr}(\boldsymbol{A}_t^2)/d$, the corresponding population coefficient and residual are

$$\kappa_t^{\mathrm{pop}} := \frac{\mathbb{E}[\boldsymbol{u}_{2,t}^\top \boldsymbol{u}_{1,t}]}{\mathbb{E}[\|\boldsymbol{u}_{1,t}\|_2^2]} = 1 - \mu_t, \qquad \boldsymbol{v}_{2,t}^{\mathrm{pop}} := \boldsymbol{u}_{2,t} - \kappa_t^{\mathrm{pop}} \boldsymbol{u}_{1,t} = -(\boldsymbol{A}_t - \mu_t \boldsymbol{I})\boldsymbol{x}_t. \tag{43}$$

This population residual satisfies $\mathbb{E}[(\boldsymbol{v}_{2,t}^{\mathrm{pop}})^\top \boldsymbol{u}_{1,t}] = 0$. It is used only for the analytically tractable controlled diagnostic below; the theorem above instead uses the sample-wise coefficient of practical $\mathrm{ProjC}^2\mathrm{G}$. In high-dimensional settings where the relevant quadratic forms concentrate, the sample-wise coefficient concentrates around this population counterpart.

C.4 Time-varying Projection Ratio

The preceding result assumes a constant target amplitude a. More generally, let the locally required first-order amplitude be a_t, while w_1 and w_2 remain fixed and c_t is shared across samples at timestep t. With $R_t = \|\boldsymbol{u}_{1,t}\|_2^2$, define the conditional moments

$$M_{1,t} := \mathbb{E}_{\boldsymbol{x}_t}[R_t \kappa_t], \qquad M_{2,t} := \mathbb{E}_{\boldsymbol{x}_t}[R_t \kappa_t^2]. \tag{44}$$

Minimizing the expected first-order mismatch at timestep t gives, when $w_2 M_{2,t} \neq 0$,

$$c_{t,\mathrm{ideal}}^* = 1 - \frac{(a_t - w_1) M_{1,t}}{w_2 M_{2,t}}. \tag{45}$$

Because a projection ratio is restricted to $[0, 1]$, the feasible solution is

$$c_t^* = \mathrm{clip}_{[0,1]}\left(c_{t,\mathrm{ideal}}^*\right), \qquad \mathrm{clip}_{[0,1]}(z) := \min\{1, \max\{0, z\}\}. \tag{46}$$

If κ_t is deterministic across samples, this expression reduces to $1 - (a_t - w_1)/(w_2 \kappa_t)$. In general, one static c can attain every timestep-wise optimum only if $(a_t - w_1) M_{1,t}/M_{2,t}$ is constant over the trajectory. Otherwise, the locally preferred projection ratio varies with t.

D Controlled Population-Geometry Experiment

Two-band family. We use the covariance family

$$\boldsymbol{A}_\delta = \mathrm{diag}\big((1+\delta)\boldsymbol{I}_{d/2}, (1-\delta)\boldsymbol{I}_{d/2}\big), \qquad 0 \leq \delta < 1, \tag{47}$$

for which $\operatorname{tr}(\boldsymbol{A}_\delta) = d$, $\mu_\delta = 1 + \delta^2$, and

$$\kappa_\delta^{\mathrm{pop}} = -\delta^2, \qquad V_\delta^{\mathrm{pop}} := \mathbb{E}[\|\boldsymbol{v}_2^{\mathrm{pop}}\|_2^2] = sd\,\delta^2(1-\delta^2). \tag{48}$$

We set $d = 64$, $s = 1$, $a = 2$, $b = 0.5$, and evaluate $\delta \in \{0, 0.1, \ldots, 0.7\}$.

Learned predictors. For each sample $\boldsymbol{x} \sim \mathcal{N}(\mathbf{0}, s\boldsymbol{A}_\delta)$, let $D^*(\boldsymbol{x}, \delta)$ denote the Bayes-optimal Gaussian predictor and define

$$D_0(\boldsymbol{x}) = D^*(\boldsymbol{x}, \delta) - \boldsymbol{\xi}(\boldsymbol{x}) - \boldsymbol{F}\boldsymbol{x}, \tag{49}$$

$$D_1(\boldsymbol{x}) = D^*(\boldsymbol{x}, \delta) - \boldsymbol{\xi}(\boldsymbol{x}) - (\boldsymbol{F} + \boldsymbol{I})\boldsymbol{x}, \tag{50}$$

$$D_2(\boldsymbol{x}) = D^*(\boldsymbol{x}, \delta) - \boldsymbol{\xi}(\boldsymbol{x}) - (\boldsymbol{F} + \boldsymbol{I} + \boldsymbol{A}_\delta)\boldsymbol{x}, \tag{51}$$

where $\boldsymbol{F} = a\boldsymbol{I} - b(\boldsymbol{A}_\delta - \mu_\delta \boldsymbol{I})$. The irreducible residual $\boldsymbol{\xi}$ is population-orthogonal to $\boldsymbol{u}_1$ and $\boldsymbol{v}_2^{\mathrm{pop}}$ and is normalized to $\mathbb{E}[\|\boldsymbol{\xi}\|_2^2] = \tau^2$ with $\tau = 0.1$. We independently train three MLPs $\widehat{D}_0, \widehat{D}_1, \widehat{D}_2$ for 60,000 optimization steps and construct all guidance directions only from their predictions:

$$\widehat{\boldsymbol{u}}_1 = \widehat{D}_0 - \widehat{D}_1, \qquad \widehat{\boldsymbol{u}}_2 = \widehat{D}_0 - 2\widehat{D}_1 + \widehat{D}_2. \tag{52}$$

Implementation and evaluation details. Training samples are generated online with $\delta \sim \mathcal{U}(0, 0.7)$ and $\boldsymbol{x} \sim \mathcal{N}(\mathbf{0}, s\boldsymbol{A}_\delta)$, where $d = 64$ and $s = 1$. We interpret $\boldsymbol{x} = \boldsymbol{X}_t$ under $\boldsymbol{X}_t = \alpha \boldsymbol{X}_0 + \sigma\boldsymbol{\varepsilon}$, with $\alpha = 0.8$, $\sigma = 0.25$, and $\boldsymbol{X}_0 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Lambda}_\delta)$, where $\boldsymbol{\Lambda}_\delta = (s\boldsymbol{A}_\delta - \sigma^2\boldsymbol{I})/\alpha^2$. Thus, the Bayes-optimal predictor is $D^*(\boldsymbol{x}, \delta) = \alpha\boldsymbol{\Lambda}_\delta(s\boldsymbol{A}_\delta)^{-1}\boldsymbol{x}$. The residual is defined coordinate-wise as $\xi_i = c(\delta)(x_i^2 - \mathrm{Var}(x_i))$, with $c(\delta) = \tau/[2\sum_i \mathrm{Var}(x_i)^2]^{1/2}$ and $\tau = 0.1$; consequently, $\mathbb{E}\|\boldsymbol{\xi}\|_2^2 = \tau^2$ and $\boldsymbol{\xi}$ is population-orthogonal to the linear guidance directions.

Each predictor is an independent conditional MLP taking $[\boldsymbol{x}; \delta] \in \mathbb{R}^{65}$ as input, with four width-256 SiLU hidden layers, a 64-dimensional output, and a linear residual skip from $\boldsymbol{x}$. We train with AdamW for 60,000 steps using batch size 512, learning rate 2×10^{-4}, weight decay 10^{-5}, and gradient-norm clipping at 5, without a learning-rate scheduler; the loss is the mean of the three predictors' element-wise MSE losses. Results use five independent training seeds. At each evaluation value $\delta \in \{0, 0.1, \ldots, 0.7\}$, and at $\delta = 0.35$ for transfer, we use 4,096 calibration samples to estimate the population projection coefficient, 2,048 validation samples to select coefficients, and 2,048 disjoint test samples to report held-out risk. First-order guidance searches 81 values in $w \in [0, 4]$; raw and projected second-order guidance use the full Cartesian product of these values and 81 values in $\beta \in [-0.5, 1.5]$. The evaluation samples are shared across training seeds. The reported risk is $\mathbb{E}\|D^* - \widehat{D}\|_2^2$, i.e., squared error per sample rather than per-coordinate MSE. Aggregation is over the five training seeds, with sample standard deviations computed using $\mathrm{ddof} = 1$.

For this controlled population-geometry diagnostic, on held-out calibration samples we estimate

$$\widehat{\kappa}_\delta^{\mathrm{pop}} = \frac{\mathbb{E}[\widehat{\boldsymbol{u}}_2^\top \widehat{\boldsymbol{u}}_1]}{\mathbb{E}[\|\widehat{\boldsymbol{u}}_1\|_2^2]}, \qquad \widehat{\boldsymbol{v}}_2^{\mathrm{pop}} = \widehat{\boldsymbol{u}}_2 - \widehat{\kappa}_\delta^{\mathrm{pop}}\widehat{\boldsymbol{u}}_1. \tag{53}$$

Recovery of the predicted geometry. The learned projection coefficient and curvature energy closely track their population predictions after convergence:

Table 7: Learned versus theoretical population geometry of the two-band Gaussian family.

δ	$\widehat{\kappa}_\delta^{\mathrm{pop}}$	$-\delta^2$	$\mathbb{E}\|\widehat{\boldsymbol{v}}_2^{\mathrm{pop}}\|_2^2$	V_δ^{pop}
0.2	−0.0450	−0.0400	2.587	2.458
0.3	−0.0869	−0.0900	5.098	5.242
0.4	−0.1527	−0.1600	8.543	8.602
0.5	−0.2437	−0.2500	12.012	12.000
0.6	−0.3607	−0.3600	14.959	14.746
0.7	−0.5010	−0.4900	16.524	15.994

Per-anisotropy optimization. The target correction is $D^* - D_0 = a\boldsymbol{u}_1 + b\boldsymbol{v}_2^{\mathrm{pop}} + \boldsymbol{\xi}$. A scalar first-order correction cannot represent the $b\boldsymbol{v}_2^{\mathrm{pop}}$ component, whereas either second-order basis can. We optimize the coefficients separately at each δ using a grid search and report held-out squared prediction risk:

Table 8: Held-out prediction risk under per-anisotropy optimization and cross-anisotropy transfer. For per-anisotropy optimization, coefficients are optimized independently at each δ. For cross-anisotropy transfer, coefficients are tuned only at $\delta_{\text{train}} = 0.35$ and frozen across test geometries.

	Per-anisotropy optimization			Cross-anisotropy transfer		
δ	First order	Raw second order	Projected second order	First order	Raw second order	Projected second order
0.0	–	–	–	0.7236	0.4050	0.9008
0.1	–	–	–	0.4955	0.3824	0.6198
0.2	0.7641	0.3153	0.3130	0.8927	0.4146	0.4302
0.3	1.6226	0.3810	0.4016	1.6310	0.4412	0.4017
0.4	2.3901	0.4121	0.4419	2.3709	0.4112	0.4435
0.5	3.3058	0.4788	0.4841	3.4078	0.7300	0.5888
0.6	4.4186	0.6146	0.6227	4.6710	1.9167	0.8764
0.7	4.6310	0.8234	0.8483	4.9466	3.9338	1.1620

The two second-order parameterizations have nearly the same optimum for each fixed problem; their small differences arise from grid-search resolution and predictor approximation error. This confirms that orthogonalization should not be interpreted as enlarging the fixed-problem correction span.

Cross-anisotropy transfer. We next tune at $\delta_{\text{train}} = 0.35$, obtaining $(w_{\text{raw}}, \beta_{\text{raw}}) = (2.05, 0.50)$ and $(w_{\text{proj}}, \beta_{\text{proj}}) = (1.95, 0.475)$, and freeze these coefficients for all test geometries. The raw first-order optimum predicted by the model is $a + b\delta^2_{\text{train}} = 2.06125$, close to the learned value. The results are shown in the right side of Table 8.

The raw and projected forms are comparable near the tuning geometry, but raw risk grows rapidly in the high-anisotropy regime. This agrees with the predicted raw transfer excess risk $b^2 sd(\delta^2_{\text{test}} - \delta^2_{\text{train}})^2$.

E Experimental Details

E.1 Time-varying projection ablation settings

For w_2, we use the same search resolution as for w_1 in the corresponding guidance method; for example, if w_1 is tuned with a resolution of 0.01, we use the same resolution for w_2. We typically search w_2 over approximately $[-w_1, w_1]$, with method-specific adjustments when necessary, and report all selected values in the corresponding tables. For static projection, we search $c \in \{0, 0.25, 0.5, 0.75, 1\}$.

For the time-varying ablation in Table 5, let $\tau = i/(N-1) \in [0, 1]$ denote normalized sampling progress, where $\tau = 0$ is the high-noise endpoint and $\tau = 1$ is the low-noise endpoint. We use

$$c_{\text{up}}(\tau) = c_{\min} + 2(c_{\text{mid}} - c_{\min})\tau, \tag{54}$$

$$c_{\text{down}}(\tau) = c_{\min} + 2(c_{\text{mid}} - c_{\min})(1 - \tau), \tag{55}$$

$$c_{\text{quad}}(\tau) = c_{\min} + 4(c_{\text{mid}} - c_{\min})\tau(1 - \tau). \tag{56}$$

Here c_{up} and c_{down} are monotone increasing and decreasing, respectively, while c_{quad} is low at both endpoints and peaks at the midpoint. We search $c_{\min}, c_{\text{mid}} \in \{0, 0.25, 0.5, 0.75, 1\}$ subject to $0 \le c_{\min} \le c_{\text{mid}} \le 1$. For the two monotone families, we additionally require $2c_{\text{mid}} - c_{\min} \le 1$, ensuring that every schedule remains in $[0, 1]$. Each three-point notation below lists the start, midpoint, and end values. The second-order weights w_2 for the SD3 settings refer to the indicated second-order branch (2-CFG or 2-SG).

Table 9: SD3 time-varying projection schedules ($N = 28$). FID is shown as static $c \to$ time-varying c_i.

Setting	w_2	c_i (start $\to$ midpoint $\to$ end)	FID↓
2-CFG	0.5	$i/27 \quad (0 \to 0.5 \to 1)$	$24.73 \to 24.65$
CFG+2-SG	-0.5	$0.5 + 0.5i/27 \quad (0.5 \to 0.75 \to 1)$	$23.93 \to 23.93$
CFG+PAG+2-SG	1	$1 - 0.5i/27 \quad (1 \to 0.75 \to 0.5)$	$24.49 \to 24.47$

Table 10: Bridge-model time-varying projection schedules. FID is shown as static $c \to$ time-varying c_i.

Setting	Steps	w_2	c_i (start $\to$ midpoint $\to$ end)	FID↓
E2H PG	10	-15	$0.5i/9 \quad (0 \to 0.25 \to 0.5)$	$1.350 \to 1.311$
E2H PG	50	-9	$0.5(1 - i/49) \quad (0.5 \to 0.25 \to 0)$	$0.782 \to 0.730$
DIODE PG	5	-12	$i/4 \quad [0, 0.25, 0.5, 0.75, 1]$	$6.05 \to 5.796$
DIODE PG	50	0.05	$4(i/49)(1 - i/49) \quad (0 \to 1 \to 0)$	$2.062 \to 2.020$

E.2 PG AND FMPG PARAMETER SETTINGS

Our PG and FMPG implementations are based on the official GuidedBridge source code (Chen et al., 2026b), available at `https://openreview.net/forum?id=wFLc5L23QL`. For FMPG, the released implementation does not provide the exact U-shaped guidance schedule used in the original experiments, preventing an exact reproduction of that scheduling strategy. We therefore adopt a static guidance schedule, which already yields a competitive FMPG baseline. Under the same static schedule, $\mathrm{ProjC}^2\mathrm{G}$ further improves upon FMPG.

Table 11 gives the complete inference parameters for the E2H and DIODE results in Table 3. Within each comparison pair, the first-order baseline and its $\mathrm{ProjC}^2\mathrm{G}$ extension use the same perturbation type, corruption strength, guidance start ratio, sampling budget, and first-order coefficient w_1. The baseline sets $w_2 = 0$, whereas the $\mathrm{ProjC}^2\mathrm{G}$ extension uses the listed w_2 and c. We evaluate all 138,567 E2H samples and all 16,502 samples in the DIODE outdoor validation split.

Table 11: Inference parameters for the PG and FMPG experiments in Table 3.

Benchmark	Guidance	Steps	Perturbation	Strength	Start	w_1	w_2	c
E2H	PG	10	noise	0.28	–	17	-15	0.25
E2H	PG	50	noise	0.18	–	11	-9	0
E2H	PG	3	noise	0.28	–	20	-30	0
E2H	FMPG	10	noise	0.28	–	$20/8^*$	-17.5	0.25
DIODE	PG	10	noise	0.30	0.3	20	-15	0
DIODE	PG	50	blur	0.40	0.5	1.20	0.05	1
DIODE	PG	5	noise	0.31	0.5	38	-12	0.75
DIODE	FMPG	10	noise	0.30	–	$22/19^*$	-16.5	0

* For FMPG, the pair denotes $w_{\mathrm{LF}}/w_{\mathrm{HF}}$; the DIODE frequency cutoff is 0.9. A dash indicates that no separate guidance start ratio is specified.

E.3 FRAMEBRIDGE VBENCH-I2V RESULTS AND PARAMETERS

Table 12 reports the complete VBench-I2V component scores underlying the summary in Table 4. The DynamiCrafter-256 and FrameBridge-VideoCrafter reference scores use the baseline values reported in the main comparison. The separate FrameBridge (Baseline) row records the reproduced unguided setting. Following the official protocol, our reproduced FrameBridge, PG, and $\mathrm{ProjC}^2\mathrm{G}$ scores are averaged over five random seeds. Bold marks the best reported value in each column across all rows.

The component abbreviations follow FrameBridge (Wang et al., 2025): CM denotes Camera Motion; I2V-SC and I2V-BC denote I2V-Subject Consistency and I2V-Background Consistency; SC and BC denote Subject Consistency and Background Consistency; MS and DD denote Motion Smoothness and Dynamic Degree; and AQ and IQ denote Aesthetic Quality and Imaging Quality, respectively.

Table 12: Complete VBench-I2V results under official normalization. Our reproduced FrameBridge and guidance results are averaged over five random seeds. All component scores are higher-is-better; bold marks the best reported value in each column across all rows, and dashes indicate unreported components.

Model / Setting	Total	CM	I2V-SC	I2V-BC	SC	BC	MS	DD	AQ	IQ	I2V	Quality
SEINE-512×320	83.49	23.36	93.97	91.90	93.21	96.29	89.47	34.31	58.42	70.97	89.62	77.37
DynamiCrafter-256	84.35	22.18	94.61	94.88	93.68	97.70	93.38	**38.69**	59.40	62.29	91.29	77.42
FrameBridge-VideoCrafter	85.37	30.72	95.60	96.28	93.71	98.54	95.75	35.77	59.38	63.28	92.83	77.92
FrameBridge (Baseline)	85.23	**30.73**	95.60	96.39	93.69	98.49	95.72	33.74	58.47	63.40	92.89	77.57
PG (blur)	85.71	27.44	96.34	96.63	94.78	99.11	97.11	31.46	59.45	64.02	93.20	78.22
PG-$\mathrm{ProjC}^2\mathrm{G}$ (blur)	**85.83**	27.57	96.34	96.68	94.76	99.13	97.12	33.90	59.40	64.00	93.23	**78.43**
PG (pooling)	85.74	26.97	96.41	96.69	94.96	99.14	97.12	30.16	**59.67**	**64.35**	93.23	78.24
PG-$\mathrm{ProjC}^2\mathrm{G}$ (pooling)	85.78	28.22	**96.45**	**96.71**	**95.01**	**99.15**	**97.16**	30.56	59.60	64.08	**93.32**	78.23

Both PG variants use the same first-order CFG coefficient and cutoff interval. Table 13 lists the perturbation strength and the $\mathrm{ProjC}^2\mathrm{G}$-specific second-order coefficient β and projection ratio c. The $+0.12$ Total-score gain for blur refers to $85.83 - 85.71$ relative to its first-order PG counterpart.

Table 13: FrameBridge inference parameters for the pooling and blur comparisons.

Variant	CFG coefficient	First-order perturbation	Cutoff	β	c
Pooling	7.5	`kernel=3, pool_scale=0.30`	0.3–0.8	-1.0	0.50
Blur	7.5	`kernel=3`, $\sigma = 0.5$	0.3–0.8	$+0.40$	0.50

E.4 EDM2 CHECKPOINT SELECTION

All EDM2 experiments use 50,000 generated samples and 32 Heun sampling steps. For official first-order AutoGuidance (AG), D_0 is the target model and D_1 is the official weaker guiding model. Our second-order AG additionally uses D_2, an early weak checkpoint. The suffix in each checkpoint name denotes the post-hoc EMA outstd, while the numerical identifier denotes the training kimg.

Table 14 lists the exact checkpoints used in each preset. The old and new protocols use the same checkpoint for each row in the table; they differ only in class sampling, as described below.

Table 14: EDM2 checkpoints used in our ImageNet experiments. First-order AG uses (D_0, D_1), while second-order AG uses (D_0, D_1, D_2). The same checkpoint set is used for the old and new protocols. The †-marked rows use the inconsistent second-order branch ablated in Table 2.

Metric	Setting	D_0	D_1	D_2 (second-order)
FID	S+XS	`edm2-img512-s-2147483-0.070.pkl`	`edm2-img512-xs-0134217-0.125.pkl`	`edm2-img512-xxs-0067108-0.235.pkl`
FID	S+XS†	`edm2-img512-s-2147483-0.070.pkl`	`edm2-img512-xs-0134217-0.125.pkl`	`edm2-img512-xs-0033554-0.150.pkl`
DINO	S+XS	`edm2-img512-s-2147483-0.120.pkl`	`edm2-img512-xs-0134217-0.165.pkl`	`edm2-img512-xxs-0033554-0.210.pkl`
DINO	S+XS†	`edm2-img512-s-2147483-0.120.pkl`	`edm2-img512-xs-0134217-0.165.pkl`	`edm2-img512-xs-0033554-0.150.pkl`
FID	XXL+M	`edm2-img512-xxl-0939524-0.075.pkl`	`edm2-img512-m-0268435-0.155.pkl`	`edm2-img512-xxs-0067108-0.235.pkl`
DINO	XXL+M	`edm2-img512-xxl-0939524-0.130.pkl`	`edm2-img512-m-0268435-0.205.pkl`	`edm2-img512-xxs-0033554-0.280.pkl`

Protocol details. The old protocol samples the class label independently for each generated image. The new protocol uses balanced sampling, generating exactly 50 images for each of the 1,000 ImageNet classes. For each configuration, we evaluate three complementary, mutually disjoint seed sets and report the minimum metric value across the three sets; each set contains 50,000 generated images. Both protocols use the same model checkpoints and 32-step Heun sampler. For S+XS, the FID and DINO settings use the formal XXS checkpoints at outstd 0.235 and 0.210, respectively.

Following the official EDM2 evaluation protocol, the same three seed sets are used across methods within each comparison. To additionally characterize run-to-run evaluation variability, the observed sample standard deviations across the three evaluations are approximately $0.014/0.007$ for S+XS FID under the old/new protocols, $0.20/0.12$ for S+XS FD-DINOv2, $0.005/0.004$ for XXL+M FID, and $0.12/0.15$ for XXL+M FD-DINOv2, respectively.

Guidance construction. For first-order AG, only D_0 and D_1 are evaluated. For second-order AG, D_2 is additionally evaluated to construct the second-order guidance direction, while all other sampling and evaluation settings are kept fixed. The S+XS and XXL+M rows therefore differ in both the target/guiding model pair and the additional weak checkpoint, rather than only in model size labels.

E.5 SD3 AUXILIARY-CAPTION PREPROCESSING

We generate the auxiliary captions offline using Qwen2.5-7B-Instruct with Hugging Face Transformers (`AutoModelForCausalLM` and `AutoTokenizer`). The model is run in FP16 with greedy decoding (`do_sample=False`) and a maximum of 32 newly generated tokens. Each original caption is formatted as `Input: {caption}\nOutput:` and provided together with the following system instruction:

> Make the image caption deliberately vaguer and less informative. You may keep the general main information, or keep only one meaningful part of the original. Decide for yourself what to retain and how to make it vague. You may omit any details, but do not replace a stated fact with a different fact and do not add new information. The output must be English. Return only one caption with no explanation.

The original caption conditions the standard CFG branch, whereas the resulting weakened caption conditions the additional guidance branch. Caption generation is performed once before image sampling, and the resulting prompt pairs are reused for all methods.

E.6 PSEUDOCODE

Algorithm 1 summarizes the inference procedure of $\mathrm{C}^2\mathrm{G}$ and $\mathrm{ProjC}^2\mathrm{G}$. At each sampling step, we construct three predictions D_0, D_1, D_2 along the method-specific reference degradation trajectory.

For AG, SG, and PG, the first-order direction is $\boldsymbol{u}_1 = D_0 - D_1$, while CFG retains its original guidance direction $\boldsymbol{u}_1 = D_0 - D_2$. The second-order contrast is shared across all methods: $\boldsymbol{u}_2 = D_0 - 2D_1 + D_2$. We then remove a fraction c_t of the component of $\boldsymbol{u}_2$ parallel to $\boldsymbol{u}_1$ and combine the resulting direction with the original first-order guidance. Setting $c_t = 0$ recovers $\mathrm{C}^2\mathrm{G}$, while $c_t > 0$ gives $\mathrm{ProjC}^2\mathrm{G}$; setting $w_2 = 0$ exactly recovers the original first-order guidance method.

Algorithm 1 Sampling with $\mathrm{C}^2\mathrm{G}$ and $\mathrm{ProjC}^2\mathrm{G}$

Require: Initial state $\boldsymbol{x}_T$, sampling timesteps $\{t_T, \dots, t_1\}$, condition c, first-order weight w_1, second-order weight w_2, projection schedule $\{c_t\}$, numerical constant ϵ

for $t \in \{t_T, \dots, t_1\}$ **do**
  Evaluate the predictions $D_0(\boldsymbol{x}_t, t, c)$, $D_1(\boldsymbol{x}_t, t, c)$, $D_2(\boldsymbol{x}_t, t, c)$
  if the underlying guidance method is CFG **then**
    $\boldsymbol{u}_1 \leftarrow D_0(\boldsymbol{x}_t, t, c) - D_2(\boldsymbol{x}_t, t, \phi)$
  else
    $\boldsymbol{u}_1 \leftarrow D_0(\boldsymbol{x}_t, t, c) - D_1(\boldsymbol{x}_t, t, c)$
  end if
  $\boldsymbol{u}_2 \leftarrow D_0(\boldsymbol{x}_t, t, c) - 2D_1(\boldsymbol{x}_t, t, c) + D_2(\boldsymbol{x}_t, t, c)$
  if Using $\mathrm{ProjC}^2\mathrm{G}$ **then**
    $\kappa_t \leftarrow \dfrac{\langle \boldsymbol{u}_2, \boldsymbol{u}_1 \rangle}{\|\boldsymbol{u}_1\|_2^2 + \epsilon}$
    $\widetilde{\boldsymbol{u}}_2 \leftarrow \boldsymbol{u}_2 - c_t \kappa_t \boldsymbol{u}_1$
    $D_{\mathrm{guide}}(\boldsymbol{x}_t, t, c) \leftarrow D_0(\boldsymbol{x}_t, t, c) + w_1 \boldsymbol{u}_1 + w_2 \widetilde{\boldsymbol{u}}_2$
  else
    $D_{\mathrm{guide}}(\boldsymbol{x}_t, t, c) \leftarrow D_0(\boldsymbol{x}_t, t, c) + w_1 \boldsymbol{u}_1 + w_2 \boldsymbol{u}_2$
  end if
  $\boldsymbol{x}_{t-1} \leftarrow \mathrm{SamplerStep}(\boldsymbol{x}_t, t, D_{\mathrm{guide}}(\boldsymbol{x}_t, t, c))$
end for
return $\boldsymbol{x}_0$

F Generated Samples

We provide qualitative examples of generated samples in the supplementary material to illustrate the improvements achieved by $\mathrm{ProjC}^2\mathrm{G}$ across different tasks and guidance methods. The six-prompt MS-COCO comparison in Figure 6 includes the three prompts selected for Figure 1.

Figures 7 and 8 provide additional qualitative examples of generated images from COCO2017 captions using CFG and CFG-$\mathrm{ProjC}^2\mathrm{G}$ with 28 sampling steps. We use the conditional-anchored coefficient $w_1 = 2.5$ for CFG (equivalent to the conventional scale $s_{\mathrm{CFG}} = 3.5$), and set the $\mathrm{ProjC}^2\mathrm{G}$ parameters to $w_2 = 0.5$ and $c = 1.0$.

F.1 Edges to Handbags, Prior Guidance (PG)

Figures 12 and 13 provide additional qualitative examples on Edges2Handbags using PG and $\mathrm{ProjC}^2\mathrm{G}$ with 10 sampling steps. We use $w_1 = 17$ for PG, and set the $\mathrm{ProjC}^2\mathrm{G}$ parameters to $w_2 = -15$ and $c = 0.25$.

F.2 DIODE, Prior Guidance (PG)

Figures 14 and 15 provide additional qualitative examples on DIODE using PG and $\mathrm{ProjC}^2\mathrm{G}$ with 10 sampling steps. We use $w_1 = 20$ for PG, and set the $\mathrm{ProjC}^2\mathrm{G}$ parameters to $w_2 = -15$ and $c = 0$.



Figure 6: Full six-prompt qualitative comparison on MS-COCO (Lin et al., 2014). Rows show unguided generation, CFG, and CFG-ProjC^{2}G (ours); prompts appear below the columns. All rows use the same codebase and evaluation protocol, with CFG evaluated at its best-performing configuration. Codebase: `https://github.com/maple-research-lab/Self-Guidance`.



Figure 7: (Part I) Qualitative comparison of generated images from COCO2017 captions using CFG and CFG-ProjC^{2}G with 28 sampling steps.

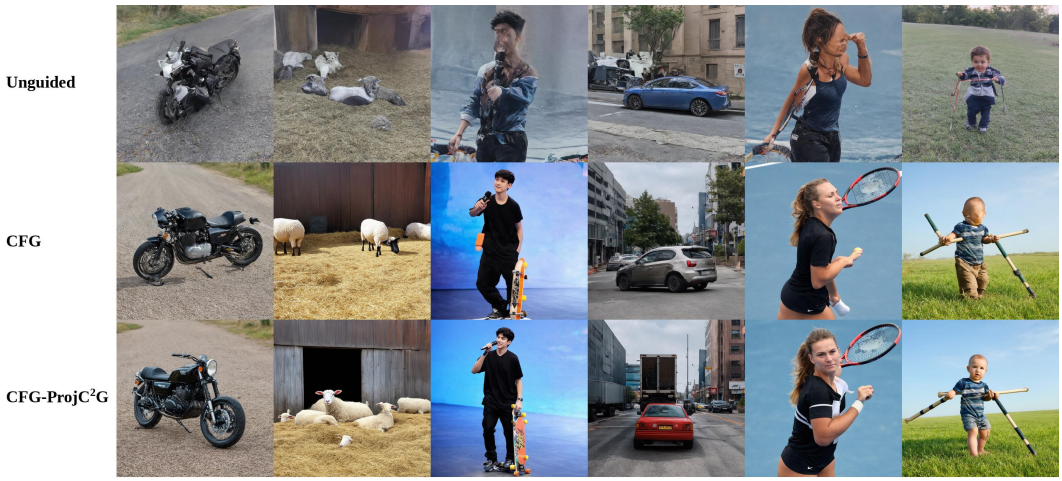


Unguided						
CFG						
CFG-ProjC^{2}G						
Prompt	A large airliner with a kangaroo on the tail wing.	Subway map art cow sculpture on the sidewalk	a profession baseball player holding a bat	A person is leaning low on their motorcycle on the tracks.	A man is talking on his phone next to a fallen bicycle.	A man hiding behind a concrete skate ramp taking pictures
Vague	A plane has an animal logo on its rear.	Artwork on the ground	An athlete with a bat	A figure is near a vehicle on some rails.	A person using a device near an object.	A person concealed by an object capturing images
Unguided						
CFG						
CFG-ProjC^{2}G						
Prompt	Motorcyclists are riding down the street with each other.	a black goat standing next to two white goats	there is a man wearing a plad shirt riding a skateboard	The ten speed bike is attached to the pole.	a giraffe standing in a pen next to a tree trunk.	A room with an old sofa, coffee table, lamp and t.v.
Vague	People on motorcycles are traveling together.	Some goats are together.	A person is moving on a skateboard.	A bike is secured to a post.	An animal near a wooden structure.	A room with some furniture and electronics.

Figure 8: (Part II) Qualitative comparison of generated images from COCO2017 captions using CFG and CFG-ProjC^{2}G with 28 sampling steps.



Prompt	People with elephants doing various activities in a river.	Two sheep in a caged area with bushes	A couple of horses by a fence on a field.	cars are stopped at a traffic light on a highway	Two birds flying in the sky over a lake	A woman walking past a store front on the sidewalk.
Vague	People and large animals near water.	Two animals in a fenced space near greenery	Animals near a barrier in an open space.	vehicles are paused at an intersection	Birds in flight near water	A person on a walk near a building.

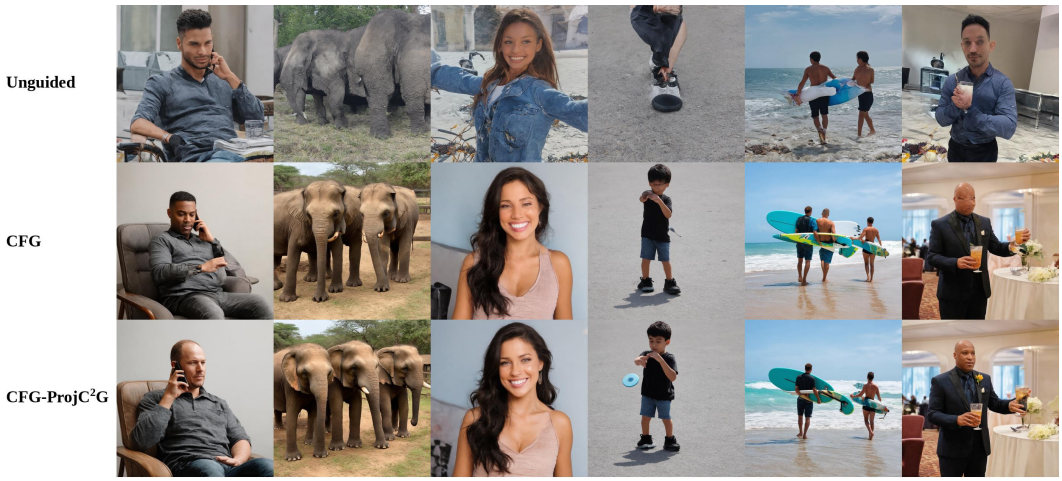


Prompt	A motorcycle is parked on a gravel road.	Sheep stand and lay in hay strewn around a barn.	A young man holding a microphone and a skate board.	A car and a large truck on a city street.	A woman holding a tennis racket over her shoulder.	A baby is holding some poles in the middle of a grass field.
Vague	A vehicle is stopped on a path.	Animals in a hay-filled area.	A person holding an object and another item.	Two vehicles on a road.	A person carrying a sports equipment.	A child is grasping objects outdoors.

Figure 9: (Part III) Qualitative comparison of generated images from COCO2017 captions using CFG and CFG-$\mathrm{ProjC}^2\mathrm{G}$ with 28 sampling steps.

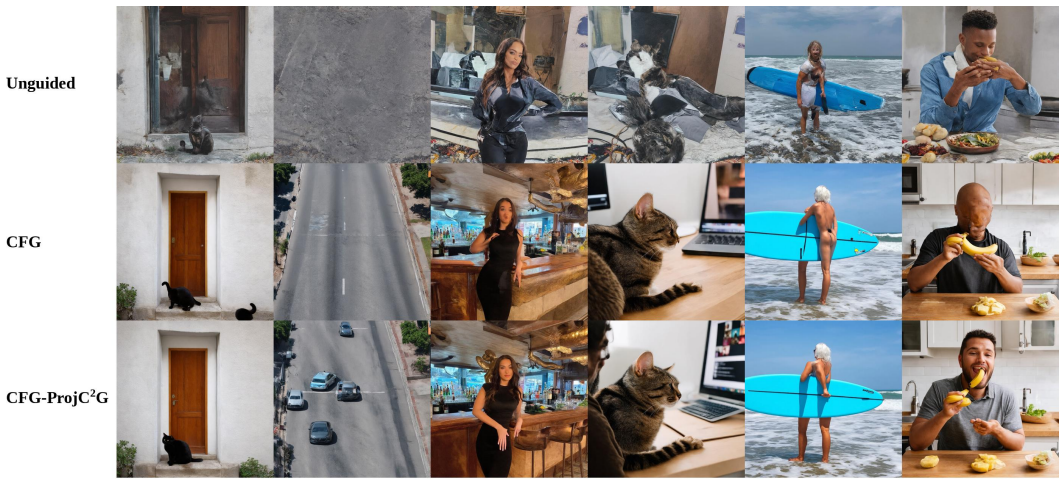


Prompt	Four children are playing a video game near each other.	A hand is reaching out to the top if a toilet.	a couple of thigs are sitting on a wood table	The boy is looking back at a wave in the ocean.	The woman has her arms outstretched with the frisbee in front of her.	A woman is holding a baby with another young boy beside her.
Vague	Kids are gaming together.	A hand is extending near an object.	Some items are on a table.	A child gazes at something in the distance near water.	A person is holding something in front.	A group of people standing together.



Prompt	A man in grey shirt sitting in chair on cellphone.	Three elephants standing in their pen are looking all around.	A woman smiling and posing for a picture.	the boy has on black sneakers and his trying to grab the frisbee	two people walking on the beach carrying surf boards	A man in a reception hall holds a drink
Vague	A person using a phone while seated.	Three animals in a enclosure are observing their surroundings.	A person smiling.	A boy reaches for something airborne while wearing dark shoes.	two figures on a sandy shore	A person with a beverage in a room

Figure 10: (Part IV) Qualitative comparison of generated images from COCO2017 captions using CFG and CFG-ProjC^{2}G with 28 sampling steps.



Prompt	A wooden door in front of an entrance next to a black cat.	A street that has four lanes of traffic.	A woman posing in front of a sculpture at a bar.	A cat trying next to his owner who is using the computer.	a white woman with a blue surf board	a person eating a banana in a kitchen
Vague	A door near an entrance with a cat nearby.	A road with multiple lanes.	A person standing near an artwork.	A pet near its human neighbor.	A woman with a board	Someone is consuming something in a room.



Prompt	The exterior of a building with a clock.	Two toy vehicles and a police box are set up in a street scene.	A little girl holding a baseball bat on grass.	A man who is looking at a baseball bat.	A man in brown shirt on black sofa with phone.	A boat can be seen at the edge of the water and a building is in the background.
Vague	A structure with a timepiece on its facade.	Two objects and a structure are arranged in a setting.	A child with a bat outdoors.	A person examining an object.	A person using a device on a piece of furniture.	A structure is visible behind a vessel near the water's edge.

Figure 11: (Part V) Qualitative comparison of generated images from COCO2017 captions using CFG and CFG-ProjC^{2}G with 28 sampling steps.



Figure 12: (Part I) Qualitative comparison on Edges2Handbags using PG and $\mathrm{ProjC}^2\mathrm{G}$ with 10 sampling steps.



Figure 13: (Part II) Qualitative comparison on Edges2Handbags using PG and $\mathrm{ProjC}^2\mathrm{G}$ with 10 sampling steps.



Figure 14: (Part I) Qualitative comparison on DIODE using PG and $\mathrm{ProjC}^2\mathrm{G}$ with 10 sampling steps.

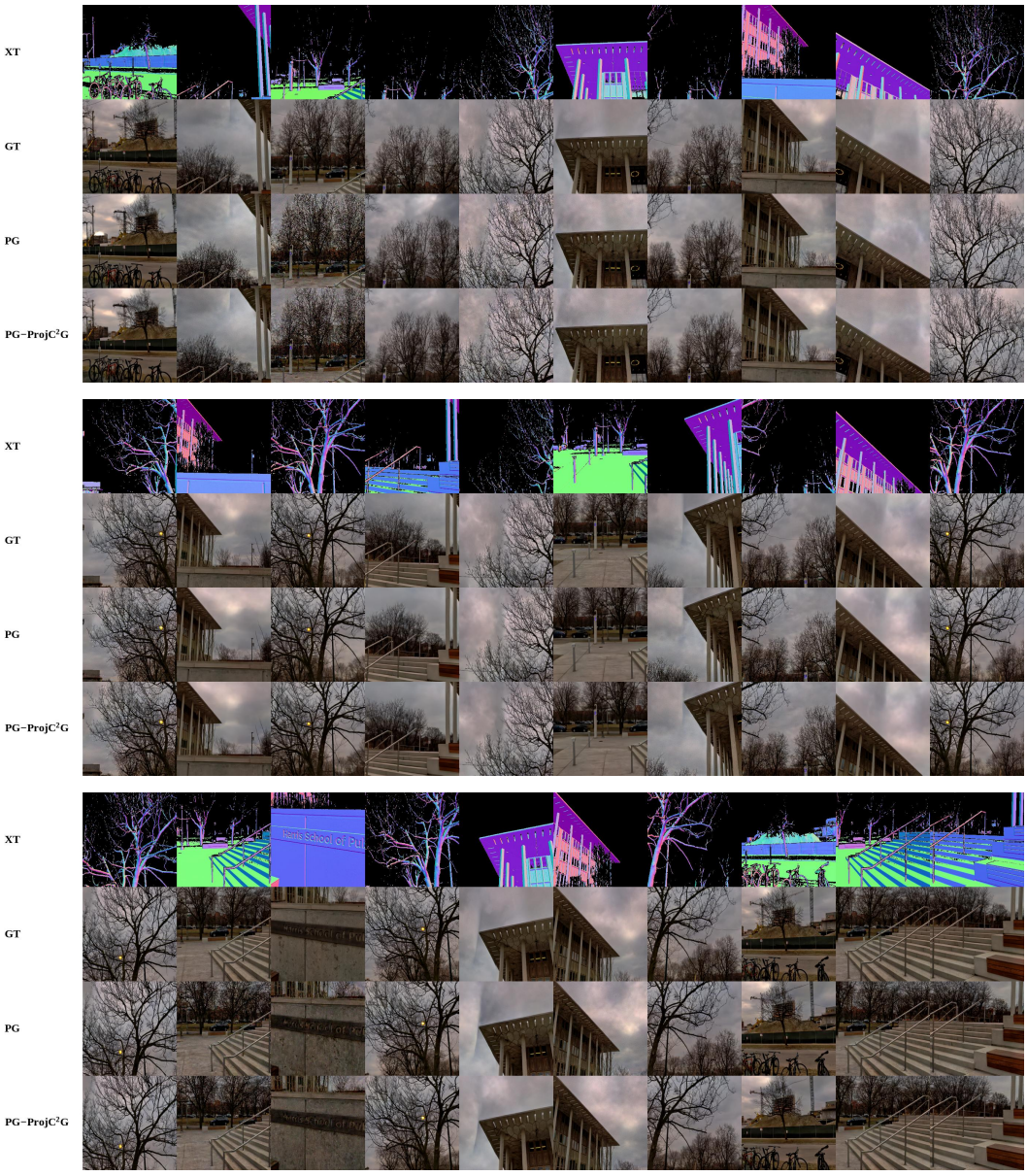


Figure 15: (Part II) Qualitative comparison on DIODE using PG and $\mathrm{ProjC}^2\mathrm{G}$ with 10 sampling steps.