ICML 2026 Workshop
SPIGM

Variance Reduction for Expectations with Diffusion Teachers

Jesse Bettencourt[1,2,3] · Xindi Wu[1,4] · Matan Atzmon[1] · James Lucas[1] · Jonathan Lorraine[1]

[1]NVIDIA · [2]University of Toronto · [3]Vector Institute · [4]Princeton University

paper · code · video

1 · Teacher gradients are expensive Monte Carlo estimates

A **frozen diffusion teacher** supplies gradients to a differentiable generator with parameters $\boldsymbol{\theta}$, used in text-to-3D (SDS), one-step distillation (DMD), and data attribution. Each gradient is an expectation over a noise level t and Gaussian noise $\boldsymbol{\epsilon}$, estimated by Monte Carlo and applied through the chain rule:

$$\mathbf{u}_{\mathrm{SDS}}(\boldsymbol{\theta}) \;=\; \mathbb{E}_{\mathbf{q},\,t,\,\boldsymbol{\epsilon}}\left[\mathbf{f}(\mathbf{x},t,\boldsymbol{\epsilon})\,\frac{\partial \mathbf{x}}{\partial \boldsymbol{\theta}}\right] \xrightarrow{\text{Monte Carlo}} \hat{\mathbf{u}}_{\boldsymbol{\theta}}^{\text{naive}} = \frac{1}{R}\sum_{r=1}^{R} \mathbf{f}\big(\mathbf{x}^{(r)},t^{(r)},\boldsymbol{\epsilon}^{(r)}\big)\,\frac{\partial \mathbf{x}^{(r)}}{\partial \boldsymbol{\theta}}$$

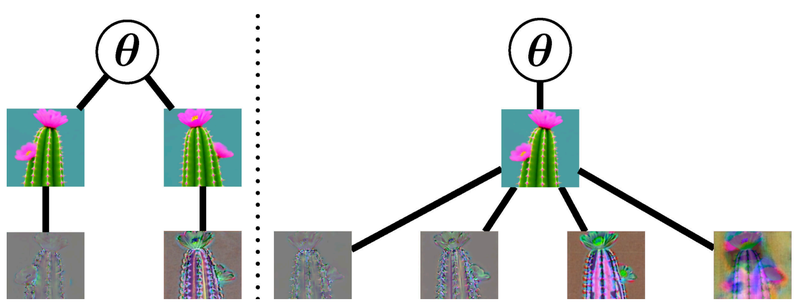
- $\mathbf{x}$ **encoded render**, $\mathbf{x} = \mathrm{Encode}\big(g(\boldsymbol{\theta},\mathbf{q})\big)$: a differentiable render g from camera $\mathbf{q}$, encoded into the teacher's latent space.
- $\mathbf{f}(\mathbf{x},t,\boldsymbol{\epsilon})$ **teacher direction**. For SDS, $\mathbf{f} = w_{\mathrm{SDS}}(t)\,\mathbf{r}$ with residual $\mathbf{r} = \hat{\boldsymbol{\epsilon}}_{\boldsymbol{\phi}}(\mathbf{z}_t,t,\mathbf{c}) - \boldsymbol{\epsilon}$, $\mathbf{z}_t = \alpha_t\mathbf{x} + \sigma_t\boldsymbol{\epsilon}$.
- $\partial\mathbf{x}/\partial\boldsymbol{\theta}$ **renderer / encoder Jacobian**: backprop through render + encode. This is the **dominant cost**.

Naive variance reduction is wasteful: each Monte Carlo estimate of the gradient averages R samples, and every sample pays the full render/encode cost ($c_{\text{denoise}} \ll c_{\text{render+encode}}$), so reducing variance by naively growing the batch size R scales this dominant cost. Downstream pipelines inherit the teacher's schedule and average ad hoc, often biasing the objective, without accounting for which randomness dominates or how it trades against compute.

CARV is a compute-aware variance-accounting framework: it measures gradient variance against wall-clock to expose which randomness actually dominates, then reduces it with **three unbiased drop-ins** that leave the objective unchanged.

2 · Method: three unbiased drop-ins

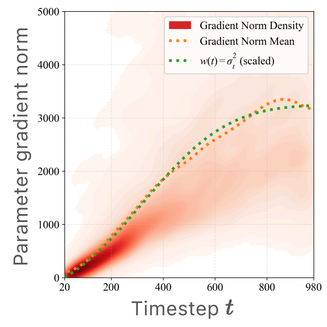
1 Amortized compute reuse



$$\hat{\mathbf{u}}_{\boldsymbol{\theta}}^{\text{reuse}} = \frac{1}{R}\sum_{r=1}^{R}\Big(\frac{1}{K}\sum_{k=1}^{K}\mathbf{f}\big(\mathbf{x}^{(r)},t^{(r,k)},\boldsymbol{\epsilon}^{(r,k)}\big)\Big)\frac{\partial \mathbf{x}^{(r)}}{\partial \boldsymbol{\theta}}$$

The naive estimator renders once per teacher evaluation ($K = 1$). Instead, reuse render $\mathbf{x}$ and average K teacher evaluations with fresh $(t,\boldsymbol{\epsilon})$: the shared vector–Jacobian product through $\partial\mathbf{x}/\partial\boldsymbol{\theta}$ is computed once, so within-render variance falls at little added cost.

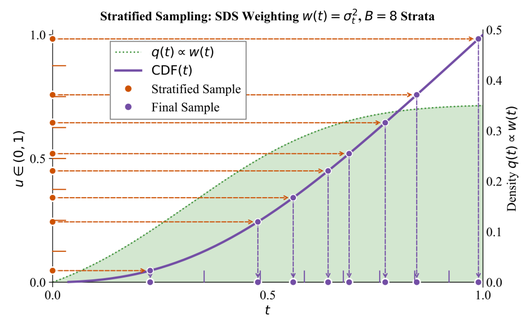
2 Timestep importance weighting (IW)



$$q^{\star}(t) \propto p(t)\sqrt{\mathbb{E}\big[\|\mathbf{f}(t,\boldsymbol{\xi})\|_2^2 \mid t\big]} \;\rightarrow\; q(t) \propto p(t)\,w_{\mathrm{SDS}}(t)$$

The variance-optimal proposal is intractable. After backprop, the parameter-gradient norm tracks the **teacher weight** $w_{\mathrm{SDS}}(t) = \sigma_t^2$ (green), a free, near-oracle surrogate.

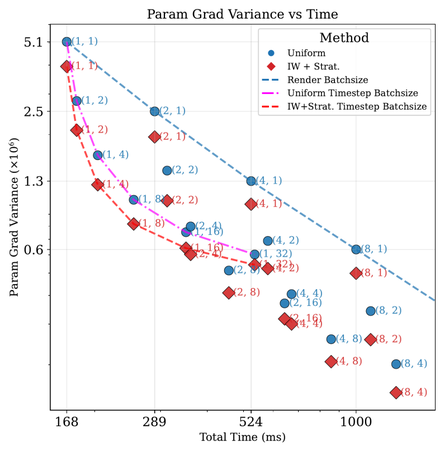
3 Stratified inverse-CDF



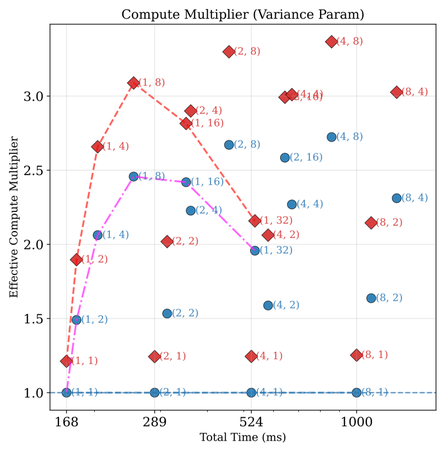
$$u_b^{(r)} = \frac{b - 1 + \boldsymbol{\xi}_b^{(r)}}{B}, \quad t_b^{(r)} = \mathrm{CDF}_q^{-1}\big(u_b^{(r)}\big)$$

IID draws of t clump and leave gaps. Forcing **one draw per equal-mass bin** and mapping through the proposal's inverse-CDF combines stratification with importance weighting at **no extra compute**.

3 · Results: 2–3× effective compute on text-to-3D



Gradient variance vs wall-clock. Each point is an (R,K) pair; at every budget, IW + stratification (red) sits below the uniform baseline (blue).

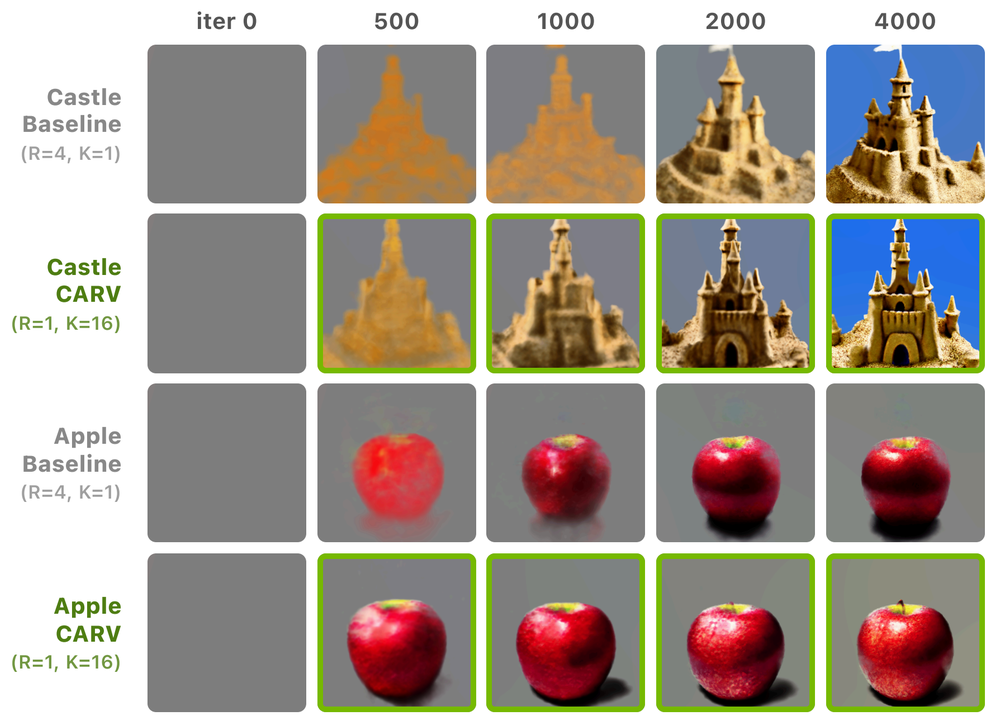


Effective compute multiplier $\big(\mathrm{cost}_{\mathrm{baseline}}/\mathrm{cost}_{\mathrm{method}}\big|_{\text{equal Var}}\big)$: baseline compute needed to match CARV's variance. Reuse (blue) drives most of the lift; IW + stratification (red) peaks at **3.3×** at $(1,8)$.

K	Uniform	IW	Strat.	IW+Strat.
1	1.00	1.24	1.00	**1.24**
2	1.57	1.93	1.66	**2.05**
4	2.24	2.68	2.47	**2.94**
8	2.63	3.01	2.96	**3.29**
16	2.52	2.75	2.75	**2.94**
32	1.98	2.09	2.08	**2.18**

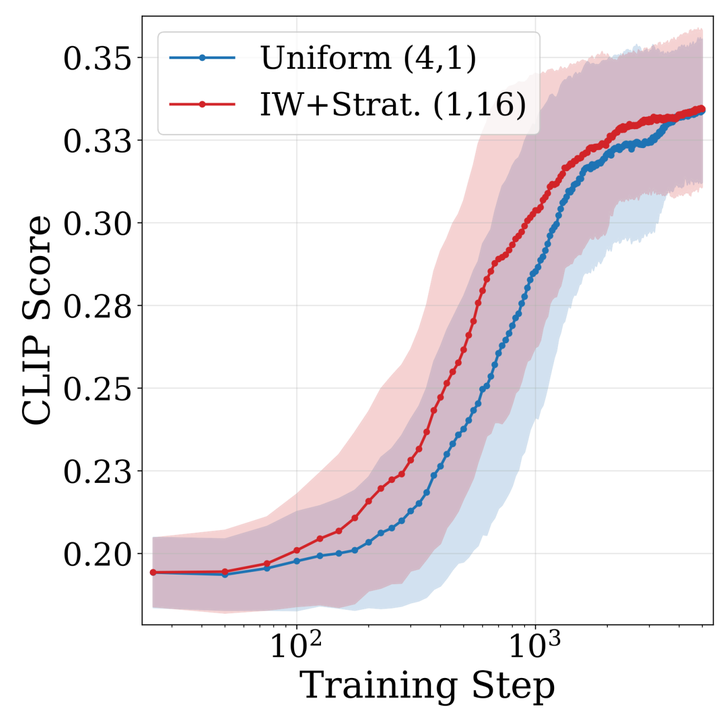
ECM by re-noising K (avg. over renders R, 5 runs). Ranking **IW+Strat > IW ≈ Strat > Uniform** is stable; the three drop-ins are complementary.

Equal-cost training trajectory · *"a castle-shaped sandcastle"* and *"a red apple"*



At matched per-iteration cost, CARV resolves coherent geometry and texture several thousand iterations earlier than the baseline.

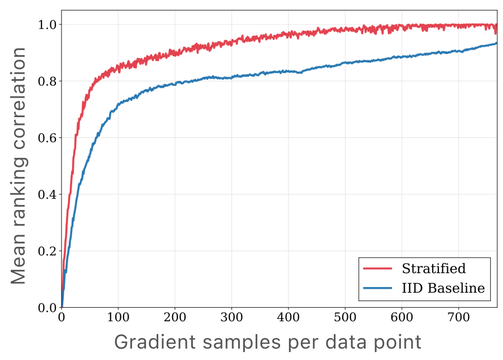
Lower variance → faster convergence



At equal cost, CARV (red, $(1,16)$) sits above the uniform baseline (blue, $(4,1)$), reaching the converged score in **~half the iterations** (30 prompts × 3 seeds).

4 · Applications beyond text-to-3D

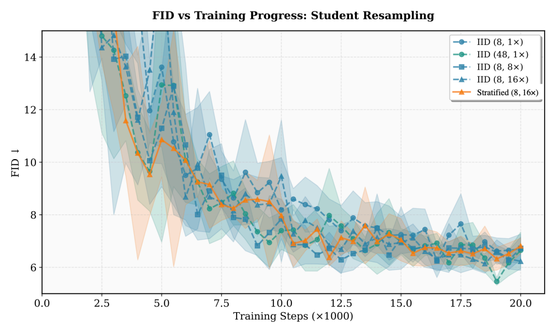
Video data attribution: the same idea drops in



$$I(\mathrm{query},n) = \frac{1}{|\mathcal{T}|}\sum_{(t,\boldsymbol{\epsilon})\in\mathcal{T}} \frac{\mathbf{g}_{\mathrm{query}}(t,\boldsymbol{\epsilon})}{\|\mathbf{g}_{\mathrm{query}}(t,\boldsymbol{\epsilon})\|_2}^{\top} \frac{\mathbf{g}_n(t,\boldsymbol{\epsilon})}{\|\mathbf{g}_n(t,\boldsymbol{\epsilon})\|_2}$$

MOTIVE-style influence on **Wan2.1** video generation (Outstanding Paper Honorable Mention). Stratified sampling (red) matches ground-truth influence rankings with far fewer per-clip gradients than IID (blue): a **2–3×** multiplier at practical budgets, with global stratification most effective.

Single-step distillation: when variance isn't the bottleneck



$$\nabla_{\boldsymbol{\theta}} D_{\mathrm{KL}} \simeq \mathbb{E}_{\boldsymbol{\epsilon},\,t,\,\boldsymbol{\epsilon}'}\left[w(t)\,\alpha_t\big(\mathbf{s}_{\mathrm{fake}}(\mathbf{z}_t,t) - \mathbf{s}_{\mathrm{real}}(\mathbf{z}_t,t)\big)\,\frac{\partial G_{\boldsymbol{\theta}}(\boldsymbol{\epsilon})}{\partial \boldsymbol{\theta}}\right]$$

The same drop-ins cut DMD gradient variance by an **order of magnitude**, yet FID does not move. DMD marks where MC variance **stops being the bottleneck**; auxiliary stabilizers and input diversity bind instead. CARV stays unbiased, and its variance-per-compute accounting exposes the regime where MC variance is the bottleneck, and, as in DMD, **where it isn't**.

When CARV helps: the MC teacher gradient dominates the update · rendering / encoding costs more than denoising ($c_{\text{render+encode}} > c_{\text{denoise}}$) · variance limits convergence.