ALIGN BEFORE DISTILLING: TEACHER-ALIGNED REPAIR FOR COMPACT ONE-STEP DIFFUSION MODELS

Anonymous authors

ABSTRACT

Pruning and step distillation are typically applied separately: pruning reduces the network size of diffusion models, while step distillation reduces their sampling cost. When combined, however, pruning can leave the student poorly initialized for distillation, which often causes step distillation to fail. In our EDM2-XS experiment, direct SiDA training from a pruned checkpoint produces unusable samples. We introduce Align Before Distilling (ABD), a teacher-aligned repair stage before step distillation: the compact model matches teacher outputs on noisy real-image latents, then uses the repaired weights to initialize one-step training. We evaluate this procedure with SD1.5/SiD-LSG, EDM2-XS/SiDA, and CIFAR-10 EDM/Diff-Instruct. On ImageNet-512, the original EDM2-XS teacher uses 124.713M parameters and 63 network evaluations at FID 3.53. Our 20% target reduction (20.7571% actual) model reaches FID 3.12 with 98.826M parameters and one evaluation; the 30% target reduction (29.4146% actual) model reaches 4.26 with 88.029M parameters. These compact models trade quality for size relative to dense one-step SiDA at FID 2.228. The SD1.5 model reduces parameters by 23.9448% and reaches FID 18.7, while the 36.807M-parameter CIFAR-10 student reaches FID 5.32 with Diff-Instruct.

1 INTRODUCTION

Diffusion models generate high-quality images for unconditional generation, class-conditional generation, text-to-image synthesis, editing, and super-resolution Ho et al. (2020); Dhariwal & Nichol (2021); Rombach et al. (2022); Karras et al. (2024), but inference is expensive: the denoising network is large and sampling requires many evaluations. On ImageNet-512, EDM2 uses 63 evaluations per image.

Pruning removes weights, channels, blocks, or heads to reduce network size Fang et al. (2023); Castells et al. (2024); Ramesh & Zhao (2024). Step distillation reduces sampling to one or a few evaluations Salimans & Ho (2022); Song et al. (2023); Yin et al. (2024); Zhou et al. (2024b). The usual pipeline recovers a pruned many-step sampler and then distills it, paying for both stages. We ask whether a compact model can enter distillation after only a short initialization stage.

Our answer is Align Before Distilling (ABD), a repair stage between pruning and distillation. ABD matches the compact model to teacher outputs on noisy real-image latents; the repaired weights then initialize one-step distillation. The final generator has fewer parameters and needs fewer sampling steps.

In pruned EDM2-XS, direct SiDA training Zhou et al. (2024b) reaches FID 112.56 with the 88.029M-parameter student (Table 1). Pruning changes the student's predictions at the noisy inputs used by the teacher-based loss. The change is damaging because SiDA and Diff-Instruct update the generator through auxiliary fake-score or adversarial terms trained jointly with it; a badly misaligned generator changes the initial sample distribution and can make the coupled gradients ill-conditioned. ABD lowers the mismatch before this coupled optimization starts. This motivates repair before distillation, although direct distillation need not fail for every pruning setting.

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.

Table 1: Replacing post-pruning retraining with step distillation is not a direct plug-in. The pruned checkpoint fails under direct one-step distillation, while ABD makes the same recovery path usable.

Model	Repair	NFE (↓)	#Params	FID (↓)
EDM2-XS teacher Karras et al. (2024)	–	63	124.713M	3.53
Pruned + direct SiDA	No	1	88.029M	112.56 (failed)
ABD (ours, 20.7571% pruned)	Yes	1	98.826M	**3.12**
ABD (ours, 29.4146% pruned)	Yes	1	88.029M	4.26

The repair stage is narrow: it does not recover a full many-step sampler. It aligns the compact model with the teacher at noisy real-image states (Figure 1). Output mismatch is an initialization diagnostic, not a direct measure of distillation-gradient quality.

On EDM2-XS/ImageNet-512, the 20% target (20.7571% actual) generator improves FID from 3.53 to 3.12 while reducing NFE from 63 to 1. The 30% target (29.4146% actual) generator keeps one-step sampling with 88.029M parameters and FID 4.26. On SD1.5, the generator falls from 859.521M to 653.710M parameters, a 23.9448% reduction, and reaches FID 18.7 (Table 3).

Our main contributions are:

- We identify the initialization failure of direct one-step distillation after structured pruning and introduce ABD as a repair stage for this transition.
- We test ABD on multiple step-distillation methods, including SD1.5/SiD-LSG, EDM2-XS/SiDA, and CIFAR-10 EDM/Diff-Instruct.
- We obtain compact one-step generators: on ImageNet-512, FID 3.12 with 98.826M parameters and FID 4.26 with 88.029M parameters.

2 Related Work

2.1 Pruning diffusion models

Pruning removes parameters and retrains the remainder Han et al. (2015). Diffusion pruning must also preserve denoising across noise levels; good performance at one level does not guarantee stable multi-step sampling.

Diff-Pruning uses a timestep-wise Taylor score for lightweight diffusion recovery Fang et al. (2023). LD-Pruner measures pruning damage in latent space Castells et al. (2024). Stable Diffusion results show abrupt quality drops at high sparsity Ramesh & Zhao (2024), and learnable masks reduce recovery cost Zhang et al. (2024). These methods still produce iterative samplers.

We instead use the compact architecture as a one-step student. Repair initializes it without first restoring full many-step performance.

SD1.5 replaces operators ranked by paired teacher–student x_0 error; EDM2 removes branches ranked by teacher-output sensitivity. Section 3 gives the shared procedure, and Appendix A lists the settings.

2.2 Diffusion distillation

Progressive distillation repeatedly halves teacher steps Salimans & Ho (2022). Consistency models learn maps that generate in one or few steps Song et al. (2023). DMD matches distributions with a score-difference loss Yin et al. (2024). Diff-Instruct minimizes an integral KL between teacher and student denoising distributions through a learned fake score Luo et al. (2023b). SiD/SiDA use score identity losses, and SiDA adds real images and adversarial training Zhou et al. (2024b).

All these objectives assume that the student can be optimized from its initialization. Pruning changes both capacity and predictions, so we test whether teacher-output alignment makes the compact student easier to train.

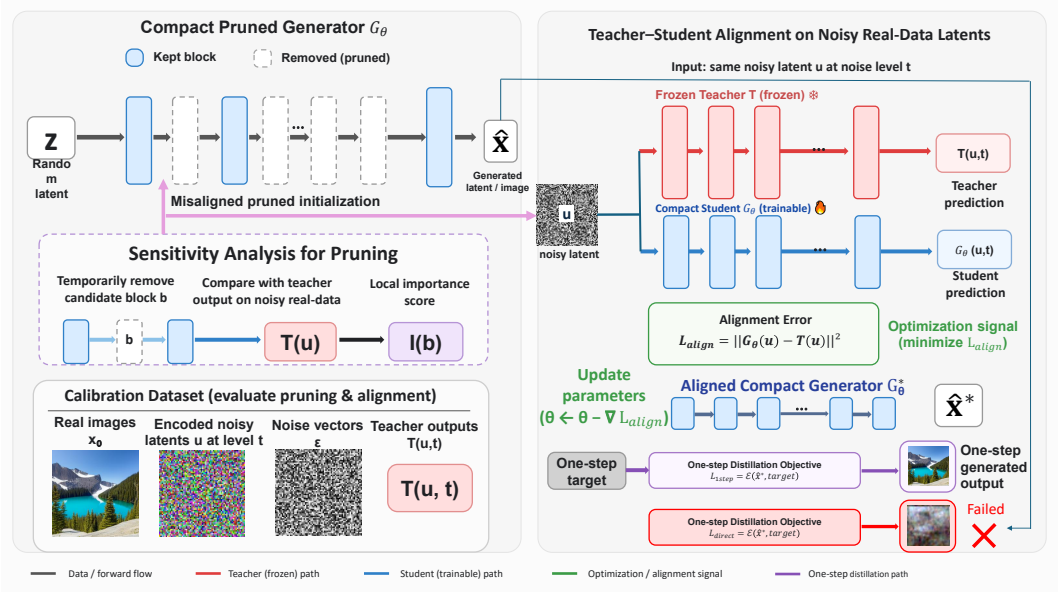

Figure 1: Overview of ABD. The repaired checkpoint $G_{\theta_{\text{align}}}$ initializes downstream distillation; the deployed generator is $G_{\theta^\star}$.

2.3 Joint compression and one-step generation

SlimFlow trains smaller one-step rectified-flow students with Annealing Reflow and Flow-Guided Distillation Zhu et al. (2024). BK-SDM removes Stable Diffusion blocks and recovers a compact multi-step sampler with feature distillation Kim et al. (2024). SnapFusion and MobileDiffusion co-design architectures, decoders, and step reduction for mobile generation Li et al. (2023); Zhao et al. (2024).

ABD covers only the transition from a pruned checkpoint to an existing one-step objective. It introduces no rectified-flow procedure, mobile architecture, or distillation loss. The chosen downstream method and its training settings therefore affect the final quality-capacity tradeoff. Direct-distillation and conventional-recovery controls test the training order under the same objective.

The closest distinction is where compression enters. SlimFlow changes reflow and distillation for smaller rectified-flow students. BK-SDM, SnapFusion, and MobileDiffusion design compact architectures or deployment pipelines jointly. ABD assumes a pruned checkpoint and an existing one-step objective, and isolates the initialization problem that arises when the two are combined. This makes ABD complementary to dedicated compact-student pipelines, although our experiments do not combine ABD with those systems.

3 Method

ABD uses a fixed order: structured pruning, teacher-aligned repair, and one-step distillation. Let G_θ be the compact generator, T the frozen teacher, and $\mathcal{D}(G_\theta; T, \mathcal{A})$ a teacher-based objective. The evaluated implementations share this order; only the pruning unit, output parameterization, and loss change.

Figure 1 shows the procedure. The pruning operator $\mathcal{P}$ constructs

$$G_{\mathcal{S}} = \mathcal{P}(T; \rho, \mathcal{B}), \tag{1}$$

where ρ is the target reduction and $\mathcal{B}$ is the candidate set. Candidates are ranked by their effect on teacher-output error. Repair then reduces the mismatch between $G_{\mathcal{S}}$ and T on noisy real-data latents, and the repaired checkpoint initializes distillation.

3.1 COMPACT GENERATOR AND STRUCTURAL PRUNING

The compact generator keeps the teacher's input/output interface and conditioning. Only internal widths or computational branches are reduced. Removed tensors are not instantiated, so pruning lowers parameters and inference computation rather than applying a mask.

For a removable block b, branch removal changes the block computation from

$$h_{b+1} = \mathrm{MPsum}\left(h_b, R_b(h_b, e)\right) + A_b(h_b) \tag{2}$$

to the surviving skip/resample path. Here R_b is a residual branch, A_b attention, and e the conditioning embedding. For EDM2, MPsum is magnitude-preserving addition Karras et al. (2024): with $t \in [0,1]$, $\mathrm{MPsum}(a,b) = ((1-t)a + tb)/\sqrt{(1-t)^2 + t^2}$, and $t = 1/2$. Other backbones keep their standard skip, concatenation, and slicing operations. Width-sliced decoder inputs are sliced separately before concatenation, preserving the U-Net topology.

3.2 TEACHER-OUTPUT SENSITIVITY

Teacher-output sensitivity decides which computations to remove with limited local damage.

We draw a fixed calibration set of real images, encode them to latents, sample noise at a chosen level $\sigma^\star$, and compute teacher targets:

$$z_i = E(x_i), \qquad \tilde{z}_i = z_i + \sigma^\star \epsilon_i, \qquad t_i = T(\tilde{z}_i, \sigma^\star, y_i), \tag{3}$$

where E is the data encoder. The calibration noise is chosen in the region relevant to the downstream one-step objective.

The calibration set defines $\hat{\mu}_{\sigma^\star}$ over noisy states. For student G, the empirical alignment error is

$$\hat{\varepsilon}_{\sigma^\star}(G) = \frac{1}{N}\sum_{i=1}^{N} \left\| G(\tilde{z}_i, \sigma^\star, y_i) - t_i \right\|_2^2 . \tag{4}$$

With candidate b temporarily removed, its importance is the loss increase:

$$I(b) = \hat{\varepsilon}_{\sigma^\star}(G^{-b}) - \hat{\varepsilon}_{\sigma^\star}(G). \tag{5}$$

Smaller $I(b)$ indicates a safer removal. We rank candidates by this increase and select under the parameter budget; Appendix A gives the search rule.

3.3 NOISY-REAL TEACHER ALIGNMENT

After pruning, $G_{\mathcal{S}}$ has the required size but may disagree with the teacher on noisy inputs. ABD reduces this mismatch before the downstream loss is applied.

Let q_r denote the repair-noise distribution. For a real-data latent $z = E(x)$ and a repair level $\sigma \sim q_r$,

$$\tilde{z}_\sigma = z + \sigma\epsilon. \tag{6}$$

The repair loss is

$$\begin{aligned} \mathcal{L}_{\mathrm{repair}}(G_{\mathcal{S}}) = \mathbb{E}_{x,y,\epsilon,\sigma\sim q_r}\left[w_{\mathrm{noisy}} \left\| G_{\mathcal{S}}(\tilde{z}_\sigma, \sigma, y) - T(\tilde{z}_\sigma, \sigma, y) \right\|_2^2 \right] \\ + \; w_{\mathrm{init}} \mathbb{E}_{y,\epsilon}\left[\left\| G_{\mathcal{S}}(\sigma_{\mathrm{init}}\epsilon, \sigma_{\mathrm{init}}, y) - T(\sigma_{\mathrm{init}}\epsilon, \sigma_{\mathrm{init}}, y) \right\|_2^2 \right]. \end{aligned} \tag{7}$$

All reported runs use $w_{\mathrm{init}} = 0$ and $w_{\mathrm{noisy}} = 1$. Student and teacher share the output parameterization required downstream, so the repaired output is directly usable. The loss is local to the distribution induced by real latents and repair noise.

A short alignment run is written as

$$\theta_{\mathrm{align}} = \theta_{\mathcal{S}} - \tau \nabla_\theta \mathcal{L}_{\mathrm{repair}}(G_\theta)\big|_{\theta=\theta_{\mathcal{S}}}. \tag{8}$$

The expression gives the update direction; in practice we run Adam for the finite budgets in Appendix A.

Algorithm 1 Teacher-aligned pruning-to-distillation

Require: teacher T; candidates $\mathcal{B}$; budget P; calibration set $\mathcal{C}$; noise $\sigma^\star$
Require: repair selector q_r; weights $w_{\text{init}}, w_{\text{noisy}}$; updates M_{repair}
Require: optimizer (Opt, τ); initialization noise σ_{init}; distillation loss $\mathcal{D}(\cdot; T, \mathcal{A})$
Ensure: compact one-step generator $G_{\theta^\star}$

```
Form (z_i, z̃_i, t_i, y_i) with Equation 3.
for all b ∈ B do estimate I(b) with Equation 5.
end for
Select low-importance blocks under P and build G_{θ_S} with copied or sliced weights.
for m = 1, ..., M_repair do
    Sample σ_m ~ q_r, (x, y), and Gaussian noise.
    Compute the weighted init-state and noisy-real loss in Equation 7.
    Update θ ← Opt(θ, ∇_θ L_m; τ).
end for
Initialize downstream distillation with θ_0 = θ_align.
Optimize (θ*, φ*_A) with D(G_θ; T, A_{φ_A}).
return G_{θ*}
```

q_r may be fixed or selected from candidates $\Sigma_r = \{\sigma_1, \ldots, \sigma_K\}$. A selector s_r gives

$$q_r = \delta_{s_r(\Sigma_r)}, \qquad \Sigma_r = \{\sigma_1, \ldots, \sigma_K\}. \tag{9}$$

ImageNet uses $q_r = \delta_{2.5}$. CIFAR-10 evaluates 16 log-spaced levels from 1.0 to 0.09 and uses the maximum, giving $q_r = \delta_{1.0}$. Diagnostic bins are separate and are not sampled by the repair update; the reported stage is not full multi-noise retraining.

Repair uses noisy real latents as inputs and teacher predictions at those inputs as targets. Alignment there does not guarantee alignment over the full generated-state distribution.

3.4 One-step distillation after alignment

The aligned compact generator $G_{\theta_{\text{align}}}$ initializes a teacher-based one-step distillation method:

$$\theta_0 \leftarrow \theta_{\text{align}}, \qquad \theta^\star = \arg \min_{\theta, \phi_{\mathcal{A}}: \theta_0 = \theta_{\text{align}}} \mathcal{D}(G_\theta; T, \mathcal{A}_{\phi_{\mathcal{A}}}). \tag{10}$$

Here θ parameterizes the compact generator and $\phi_{\mathcal{A}}$ an auxiliary score, fake-score, or critic. Only G_θ is pruned; $\mathcal{A}_{\phi_{\mathcal{A}}}$ remains dense, is initialized from the teacher, and is discarded after training. Parameter counts and NFE therefore describe the deployed generator, not training-time memory.

The training-time distinction is concrete. The CIFAR-10 generator is 36.807M, while its Diff-Instruct score generator remains 55.734M. The ImageNet SiDA run initializes a dense fake-score network from the 124.713M teacher. Thus ABD reduces the network used for sampling, but the reported speedups should not be read as reductions in training memory or in the cost of the auxiliary objective.

Algorithm 1 selects low-importance blocks, instantiates the compact network, repairs it under q_r, and initializes downstream distillation. Pruning units and noise rules are implementation-specific; Appendix A lists them.

4 Experiments

We test whether repair makes a pruned model trainable, whether ABD transfers across backbones and losses, and how much recovery cost it removes. ImageNet-512 and SD1.5 are the main tracks; CIFAR-10 is a transfer test. Where available, controls compare direct distillation, teacher-aligned initialization, and conventional recovery.

4.1 Datasets, protocols, and model tracks

The main tracks are class-conditional ImageNet-512 with EDM2/SiDA and text-to-image SD1.5 with SiD-LSG; CIFAR-10 uses Diff-Instruct. Calibration and repair use real training data. Evaluation

Table 2: ImageNet-512 comparison. Additional literature baselines follow the SiDA comparison Zhou et al. (2024b), retaining their reported evaluation-count conventions; their times were not measured in our setup.

Method	NFE (↓)	#Params	FID (↓)	Time/img
ADM-G Dhariwal & Nichol (2021)	250	559M	23.24	–
U-DiT-B Tian et al. (2024)	250	204M	15.39	–
DiT-XL/2 Peebles & Xie (2023)	250	675M	12.03	–
MaskDiT Zheng et al. (2023)	79	736M	10.79	–
ADM-U Dhariwal & Nichol (2021)	250	>559M	9.96	–
LEGO-XL-PR Zheng et al. (2024)	250	681M	9.01	–
BigGAN-deep Brock et al. (2018)	1	160M	8.43	–
DiMR-XL/3R Liu et al. (2024)	250	525M	7.93	–
MaskGIT Chang et al. (2022)	12	227M	7.32	–
MAGVIT-v2 Yu et al. (2024)	12	307M	4.61	–
RIN Jabri et al. (2022)	1000	320M	3.95	–
MAGVIT-v2 Yu et al. (2024)	64	307M	3.07	–
VDM++ Kingma & Gao (2023)	512	2B	2.99	–
MAR Li et al. (2024)	100	481M	2.74	–
StyleGAN-XL Sauer et al. (2022)	2	168M	2.41	–
SiDA-EDM2-XS Zhou et al. (2024b)	1	125M	2.228	–
EDM2-XS Karras et al. (2024)	63	124.713M	3.53	3390.8 ms
ABD (ours, 20.7571% pruned)	1	98.826M	3.12	62.7 ms
ABD (ours, 29.4146% pruned)	1	88.029M	4.26	51.6 ms

follows each public implementation's preprocessing, sample count, feature extractor, and conditioning protocol.

We report FID, deployed-generator parameters, and NFE. Full-capability auxiliary score or critic networks used only during distillation are excluded. One sampling step is one generator evaluation for our models; baseline conventions follow their solvers.

4.2 Implementation setup

All variants prune only the generator. SD1.5 replaces operators ranked by paired teacher–student denoised-output error, then applies noisy-real repair. ImageNet removes branches ranked by teacher-output sensitivity, then applies the same repair.

SD1.5 then uses SiD-LSG and ImageNet uses SiDA, with the teacher frozen during recovery. Code is available at `https://anonymous.4open.science/r/BRIDGING-C5EC`.

4.3 Main results

Tables 2 and 3 mix our measurements with literature results. We retain each baseline's reported evaluation convention; only the EDM2-XS and ABD times are measured in our setup. The strongest controlled comparison is the ImageNet pair in Table 1, which changes only initialization. CIFAR-10 controls are useful operating points, but their logged budgets differ. We therefore claim trainability and observed tradeoffs, not universal Pareto dominance.

Direct distillation from the 29.4146%-pruned checkpoint fails, whereas alignment makes both compact models trainable (Tables 1 and 2). The 20.7571%-pruned model improves FID over EDM2-XS while reducing NFE from 63 to 1; the 29.4146%-pruned model gives a smaller one-step generator.

Dense SiDA reaches FID 2.228 with about 125M parameters. Our 98.826M and 88.029M students reach 3.12 and 4.26, respectively. Dense SiDA latency is unavailable, so pruning's separate latency gain cannot be isolated.

ABD also transfers to SD1.5 (Table 3): 653.7M parameters, a 23.9448% reduction, and FID 18.7 at one evaluation. Other 512-pixel generators provide context because their backbones and budgets differ.

Table 3: Generation results at 512×512. Reference entries are retained from the original SiD-LSG comparison Zhou et al. (2024a), with their reported step or evaluation counts and timing conventions. Other backbones are grouped separately; their results are contextual references, not matched-budget controls. Dashes denote unreported values. For StyleGAN-XL, NFE is reported as two forward passes.

Method	NFE (↓)	#Params	FID (↓)	Time
Other 512-pixel generators (reference results)				
StyleGAN-T Sauer et al. (2023a)	1	1B	13.90	0.10s
GigaGAN Kang et al. (2023)	1	1B	9.09	0.13s
ADD (SD-Turbo) Sauer et al. (2023b)	1	–	16.25	–
ADD (SDXL-Turbo) Sauer et al. (2023b)	1	–	19.08	–
SD1.5 and its accelerated or distilled variants				
SD1.5 (CFG=3) Rombach et al. (2022)	50	0.9B	8.78	2.59s
SD1.5 (CFG=8) Rombach et al. (2022)	50	0.9B	13.45	2.59s
DPM++ (4 step) Lu et al. (2022)	4	0.9B	22.44	0.26s
UniPC (4 step) Zhao et al. (2023)	4	0.9B	22.30	0.26s
LCM-LoRA (4 step) Luo et al. (2023a)	4	0.9B	23.62	0.19s
LCM-LoRA (1 step) Luo et al. (2023a)	1	0.9B	77.90	0.07s
InstaFlow-0.9B Liu et al. (2023)	1	0.9B	13.10	0.09s
InstaFlow-1.7B Liu et al. (2023)	1	1.7B	11.83	0.12s
UFOGen Xu et al. (2023)	1	0.9B	12.78	0.09s
DMD (CFG=3) Yin et al. (2024)	1	0.9B	11.49	0.09s
DMD (CFG=8) Yin et al. (2024)	1	0.9B	14.93	0.09s
BOOT Gu et al. (2023)	1	0.9B	48.20	0.09s
Guided Distillation Meng et al. (2022)	1	0.9B	37.3	–
LSG Zhou et al. (2024a)	1	0.9B	13.5	0.09s
ABD (ours, 24% pruned)	1	**653.7M**	**18.7**	0.07s

Table 4: ImageNet-512 ablation study. The reference uses 40,000 repair updates, while the no-repair control uses zero. The pruning-score and repair-noise variants also use 40,000 repair updates. The no-repair comparison retains the reference compact architecture. The pruning-score variant uses attention-only sensitivity: each candidate is ranked by the increase in teacher-output MSE when only its attention branch is disabled, whereas the reference removes the full residual–attention branch.

Component (changed part)	Variant (setting)	Repair steps	FID (↓)
Reference (none)	Teacher-aware pruning + high-noise repair	40,000	**4.26**
Repair stage	No repair; direct distillation from pruned checkpoint	0	112.56
Pruning score	Attention-only sensitivity ranking	40,000	8.16
Repair noise	Matched multi-noise alignment	40,000	8.96

4.4 Ablation and transfer evidence

With architecture and downstream budget fixed, removing repair raises FID from 4.26 to 112.56 (Table 4). Attention-only sensitivity ranking gives 8.16, and matched multi-noise alignment gives 8.96. Both the structural score and repair distribution matter.

On CIFAR-10, ABD plus Diff-Instruct reaches FID 5.32 with 36.807M parameters (Table 5); dense Diff-Instruct reaches 4.53 with 55.73M. Appendix A reports alignment diagnostics, and Table 6 reports recovery costs. Logged image budgets are not exactly matched.

Table 5: Compact CIFAR-10 transfer results. The proposed route uses pruning, noisy-real repair, and one-step Diff-Instruct.

Method	NFE (↓)	#Params	FID (↓)	Time
EDM Karras et al. (2022)	35	55.734M	1.97	36.953s
Diff-Instruct Luo et al. (2023b)	1	55.73M	4.53	–
Conventional recovery → Diff-Instruct	1	36.807M	6.26	–
ABD (ours): repair → Diff-Instruct	1	**36.807M**	**5.32**	0.921s

Table 6: Measured CIFAR-10 recovery cost. Image budgets differ slightly between logged runs; times and FIDs are therefore reported as measured operating points rather than an exact speedup.

Pipeline	Pre-distillation (h)	Distillation (h)	FID (↓)
Direct pruned Diff-Instruct	–	2.11	264.64
Repair-only	0.48	–	181.04
Conventional recovery → Diff-Instruct	9.21	17.57	6.26
ABD (ours): repair → Diff-Instruct	0.48	17.11	**5.32**

4.5 Analysis of ABD

The ImageNet control compares two initializations of the same 88.029M architecture under the same budget. Repair lowers FID from 112.56 to 4.26 (Table 4), so the copied and repaired weights lead to very different distillation outcomes.

CIFAR-10 repair reduces generator relative MSE from 11.26197 to 0.07303 and lowers mean and maximum score error (Appendix A). Yet repair alone reaches FID 181.04, while repair followed by Diff-Instruct reaches 5.32. Alignment measures initialization; the downstream objective still trains the generator.

Repair takes 0.48 h versus 9.21 h for conventional recovery. Including distillation, totals are 17.59 h at FID 5.32 and 26.78 h at FID 6.26. Most repair-route time is distillation. Because budgets and accounting differ, these are observed operating points, and the short direct control shows failure only within its training window.

5 Limitations

Our evidence covers three backbone–objective combinations. It supports reuse of ABD but does not show independence from the downstream objective. Pruning units, output parameterizations, and structural scores are implementation-specific, and other architectures or multi-step distillation may need different calibration and noise schedules.

Repair needs representative data, teacher access, and extra optimization. Its effectiveness depends on calibration coverage, pruning ratio, repair noise, and update budget. Matching sampled noisy states does not guarantee alignment during generation, and we give no data-free variant or cross-backbone budget rule.

ABD is not a new step-distillation method. It uses SiD-LSG, SiDA, or Diff-Instruct, so the objective and its training settings affect the final tradeoff. The pruning ratio, calibration set, repair noise, and update budget also remain empirical choices. The reported schedules are fixed before distillation rather than learned jointly with the downstream loss. Finally, ABD does not remove the search for a compact architecture or the choice of a distillation method; it shortens the transition between them.

6 Conclusion

ABD connects structured pruning to one-step distillation by aligning the compact initialization with the teacher on noisy real-data latents before distillation. We evaluated it with SD1.5/SiD-LSG, EDM2-XS/SiDA, and CIFAR-10 EDM/Diff-Instruct.

On ImageNet-512, the 20.7571%-pruned model reaches FID 3.12 with 98.826M parameters and one evaluation. Compared with dense one-step distillation, the compact models trade quality for size. Recovery savings beyond these runs require matched budgets.

In practice, ABD is most useful when a pruning rule has already produced a compact architecture and a one-step objective has already been selected. The recipe is to measure teacher-output mismatch on high-noise real states, run a short repair, and then start the original distillation from the repaired checkpoint. The mismatch is an initialization check, while the final quality must still be measured after downstream training.

Three extensions would sharpen the evidence. Matched-budget conventional-recovery runs would separate repair savings from training duration. Learning the repair schedule jointly with the downstream loss would test whether the fixed high-noise choice is conservative. Combining ABD with compact architecture designers such as SlimFlow or MobileDiffusion would show whether initialization repair remains useful when the student and objective are designed together.

REFERENCES

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

Thibault Castells, Hyoung-Kyu Song, Bo-Kyeong Kim, and Shinkook Choi. Ld-pruner: Efficient pruning of latent diffusion models using task-agnostic insights. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 821–830. IEEE, 2024.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11305–11315. IEEE, 2022.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Gongfan Fang, Xinyin Ma, and Xinchao Wang. Structural pruning for diffusion models, 2023. URL https://arxiv.org/abs/2305.10924.

Jiatao Gu, Shuangfei Zhai, Yizhe Zhang, Lingjie Liu, and Josh Susskind. BOOT: Data-free Distillation of Denoising Diffusion Models with Bootstrapping. *arXiv preprint arXiv:2306.05544*, 2023. URL https://arxiv.org/abs/2306.05544.

Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28, 2015.

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

Allan Jabri, David Fleet, and Ting Chen. Scalable adaptive computation for iterative generation. *arXiv preprint arXiv:2212.11972*, 2022.

Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up GANs for Text-to-Image Synthesis. *arXiv preprint arXiv:2303.05511*, 2023. URL https://arxiv.org/abs/2303.05511.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

Tero Karras, Miika Aittala, Jaakko Lehtinen, Janne Hellsten, Timo Aila, and Samuli Laine. Analyzing and improving the training dynamics of diffusion models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24174–24184. IEEE, 2024.

Bo-Kyeong Kim, Hyoung-Kyu Song, Thibault Castells, and Shinkook Choi. Bk-sdm: A lightweight, fast, and cheap version of stable diffusion. In *European Conference on Computer Vision*, pp. 381–399. Springer, 2024.

Diederik Kingma and Ruiqi Gao. Understanding diffusion objectives as the elbo with simple data augmentation. *Advances in Neural Information Processing Systems*, 36:65484–65516, 2023.

Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. *Advances in Neural Information Processing Systems*, 37: 56424–56445, 2024.

Yanyu Li, Huan Wang, Qing Jin, Ju Hu, Pavlo Chemerys, Yun Fu, Yanzhi Wang, Sergey Tulyakov, and Jian Ren. Snapfusion: Text-to-image diffusion model on mobile devices within two seconds. *Advances in Neural Information Processing Systems*, 36:20662–20678, 2023.

Qihao Liu, Zhanpeng Zeng, Ju He, Qihang Yu, Xiaohui Shen, and Liang-Chieh Chen. Alleviating distortion in image generation via multi-resolution diffusion models and time-dependent layer normalization. *Advances in Neural Information Processing Systems*, 37:133879–133907, 2024.

Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, and Qiang Liu. InstaFlow: One Step is Enough for High-Quality Diffusion-Based Text-to-Image Generation. *arXiv preprint arXiv:2309.06380*, 2023. URL https://arxiv.org/abs/2309.06380.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver++: Fast Solver for Guided Sampling of Diffusion Probabilistic Models. *arXiv preprint arXiv:2211.01095*, 2022. URL https://arxiv.org/abs/2211.01095.

Simian Luo, Yiqin Tan, Suraj Patil, Daniel Gu, Patrick von Platen, Apolinário Passos, Longbo Huang, Jian Li, and Hang Zhao. LCM-LoRA: A Universal Stable-Diffusion Acceleration Module. *arXiv preprint arXiv:2311.05556*, 2023a. URL https://arxiv.org/abs/2311.05556.

Weijian Luo, Tianyang Hu, Shifeng Zhang, Jiacheng Sun, Zhenguo Li, and Zhihua Zhang. Diff-instruct: A universal approach for transferring knowledge from pre-trained diffusion models. *Advances in Neural Information Processing Systems*, 36:76525–76546, 2023b.

Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik P. Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On Distillation of Guided Diffusion Models. *arXiv preprint arXiv:2210.03142*, 2022. URL https://arxiv.org/abs/2210.03142.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4172–4182. IEEE, 2023.

Samarth N Ramesh and Zhixue Zhao. Efficient pruning of text-to-image models: Insights from pruning stable diffusion. *arXiv preprint arXiv:2411.15113*, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10674–10685. ieee, 2022.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

Axel Sauer, Katja Schwarz, and Andreas Geiger. Stylegan-xl: Scaling stylegan to large diverse datasets. In *ACM SIGGRAPH 2022 conference proceedings*, pp. 1–10, 2022.

Axel Sauer, Tero Karras, Samuli Laine, Andreas Geiger, and Timo Aila. StyleGAN-T: Unlocking the Power of GANs for Fast Large-Scale Text-to-Image Synthesis. *arXiv preprint arXiv:2301.09515*, 2023a. URL https://arxiv.org/abs/2301.09515.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial Diffusion Distillation. *arXiv preprint arXiv:2311.17042*, 2023b. URL https://arxiv.org/abs/2311.17042.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models, 2023. URL https://arxiv.org/abs/2303.01469.

Yuchuan Tian, Zhijun Tu, Hanting Chen, Jie Hu, Chao Xu, and Yunhe Wang. U-dits: Downsample tokens in u-shaped diffusion transformers. *Advances in Neural Information Processing Systems*, 37:51994–52013, 2024.

Yanwu Xu, Yang Zhao, Zhisheng Xiao, and Tingbo Hou. UFOGen: You Forward Once Large Scale Text-to-Image Generation via Diffusion GANs. *arXiv preprint arXiv:2311.09257*, 2023. URL https://arxiv.org/abs/2311.09257.

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

Lijun Yu, José Lezama, Nitesh Bharadwaj Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Agrim Gupta, Xiuye Gu, Alexander G Hauptmann, et al. Language model beats diffusion-tokenizer is key to visual generation. In *International Conference on Learning Representations*, volume 2024, pp. 765–783, 2024.

Yang Zhang, Er Jin, Wenzhong Liang, Yanfei Dong, Ashkan Khakzar, Philip Torr, Johannes Stegmaier, and Kenji Kawaguchi. Learnable sparsity for vision generative models. *arXiv preprint arXiv:2412.02852*, 2024.

Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. Unipc: A unified predictor-corrector framework for fast sampling of diffusion models. *Advances in Neural Information Processing Systems*, 36:49842–49869, 2023.

Yang Zhao, Yanwu Xu, Zhisheng Xiao, Haolin Jia, and Tingbo Hou. Mobilediffusion: Instant text-to-image generation on mobile devices. In *European Conference on Computer Vision*, pp. 225–242, 2024.

Hongkai Zheng, Weili Nie, Arash Vahdat, and Anima Anandkumar. Fast training of diffusion models with masked transformers. *arXiv preprint arXiv:2306.09305*, 2023.

Huangjie Zheng, Zhendong Wang, Jianbo Yuan, Guanghan Ning, Pengcheng He, Quanzeng You, Hongxia Yang, and Mingyuan Zhou. Learning stackable and skippable lego bricks for efficient, reconfigurable, and variable-resolution diffusion modeling. In *International conference on learning representations*, volume 2024, pp. 3030–3055, 2024.

Mingyuan Zhou, Zhendong Wang, Huangjie Zheng, and Hai Huang. Guided score identity distillation for data-free one-step text-to-image generation. *arXiv preprint arXiv:2406.01561*, 2024a.

Mingyuan Zhou, Huangjie Zheng, Yi Gu, Zhendong Wang, and Hai Huang. Adversarial score identity distillation: Rapidly surpassing the teacher in one step. *arXiv preprint arXiv:2410.14919*, 2024b.

Yuanzhi Zhu, Xingchao Liu, and Qiang Liu. Slimflow: Training smaller one-step diffusion models with rectified flow. In *European Conference on Computer Vision*, pp. 342–359, 2024.

APPENDIX

A IMPLEMENTATION DETAILS

This appendix records the pruning configurations and training settings for the three evaluated backbone–objective combinations.

A.1 IMAGENET-512 SETTINGS

For ImageNet-512, the teacher is EDM2-XS with 124.713M parameters. The reported compact generators are built directly from the teacher with model channels 128, channel list $(128, 256, 384, 512)$, and 64 channels per attention head. Block sensitivity uses $\sigma^\star = 2.5$, 8 calibration batches of 8 samples, and seed 1234. Candidates are sorted by teacher-output MSE increase and selected with the low-delta-closest-prefix rule. The teacher-alignment repair stage uses the single operational level $q_r = \delta_{2.5}$, while $\{1.0, 0.3, 0.09\}$ are diagnostic bins only and are not jointly sampled by the reported repair update. It uses 40,000 updates, learning rate 2×10^{-4}, and effective batch size 128. On four GPUs, the 40,000-update repair run takes 8.15 wall-clock hours (32.6 GPU-hours). After repair, SiDA is run with the frozen EDM2 teacher as the true score model and a dense 124.713M fake-score network initialized from the teacher. This fake-score network is not pruned; it is used only during distillation and is excluded from deployed-generator parameter counts.

Table 7: SiDA/EDM2 pruning and repair settings for the ImageNet-512 models. Actual reductions differ slightly from the targets because complete residual-attention branches are removed as discrete units.

Item	20% target model	30% target model
Selection rule	low-delta-closest-prefix; $\sigma^\star = 2.5$; 8 batches $\times$ 8; seed 1234	
Disabled blocks	`enc.8x8_block2`, `enc.8x8_block1`, `dec.8x8_block3`, `enc.8x8_block0`	`enc.8x8_block2`, `enc.8x8_block1`, `dec.8x8_block3`, `enc.8x8_block0`, `dec.8x8_in0`, `dec.16x16_block1`
Parameters after pruning	98.826401M	88.029343M
Actual reduction vs. teacher	20.7571%	29.4146%
Repair	40,000 steps; batch 32/GPU $\times$ 4; Adam, lr $= 2 \times 10^{-4}$; $q_r = \delta_{2.5}$; diagnostic bins $\{1.0, 0.3, 0.09\}$; 8.15 wall-clock h (32.6 GPU-h)	

Table 8 lists the disabled encoder and decoder branches of the 88.029M ImageNet model using their module names in the EDM2 implementation.

Table 8: Disabled branches for the more compressed ImageNet-512 compact generator.

U-Net side	Disabled branches
Encoder	`enc.8x8_block0`, `enc.8x8_block1`, `enc.8x8_block2`
Decoder	`dec.8x8_in0`, `dec.8x8_block3`, `dec.16x16_block1`

A.2 STABLE DIFFUSION 1.5 SID-LSG SETTINGS

Table 9 records the operator-pruning and SiD-LSG training settings for the SD1.5 results in Table 3.

Table 9: Pruning and distillation settings for the SD1.5 SiD-LSG main track.

Item	Setting
Base model and resolution	Stable Diffusion 1.5; 512×512
Pruning unit	LD-Pruner operator replacement with identity-like residual/transformer/downsampling replacements
Actual parameter reduction	23.9448%
Parameters before / after	859.520964M / 653.710084M
Calibration data	LAION-Aesthetics 6.5+ real images and text prompts
Ranking score	`repair_x0_mse`: paired teacher–student x_0 MSE on noisy real-image latents
Score timestep	625
Score conditions / samples	1,250 conditions; 8 samples per condition; 10,000 total
Score batch / seed	Batch 8; seed 1234
Candidate / selected operators	44 candidates / 8 selected operators
Repair	30,000 steps; batch 1; learning rate 1×10^{-5}; teacher on `cuda:1`
Pruning precision	FP16; xFormers disabled
Downstream SiD-LSG	Init timestep 625; batch 128 (batch/GPU 1); lr $=$ glr $= 1 \times 10^{-6}$; CFG 2; duration 10 epochs

A.3 CIFAR-10 DIFF-INSTRUCT SETTINGS

For the CIFAR-10 experiment in Table 5, we start from the public EDM CIFAR-10 unconditional VP checkpoint and use Diff-Instruct as the downstream one-step distillation objective. The dataset is CIFAR-10 at 32×32 resolution, with 50k training images, no labels, and no horizontal flipping. The reported compact variants prune only the Diff-Instruct generator; the score generator remains dense at the full 55.734M teacher capacity and is initialized from the teacher checkpoint. The teacher architecture is EDMPrecond SongUNet with 128 base channels, channel multipliers $(2, 2, 2)$, dropout 0.13, and $\sigma_{\text{init}} = 1.0$.

The conventional-recovery control trains the 36.807M-parameter pruned generator for 5,800 kimg with teacher-denoising regression, batch size 128, and learning rate 2×10^{-4}. Its final paired checkpoint retains the full 55.734M-parameter score network. The subsequent Diff-Instruct run restores this pair and uses the common settings in Table 10 for another 5,800 kimg; no pruning or repair update is repeated during this second stage. Training takes 9.2081 h for recovery and 17.5730 h for the subsequent distillation.

Table 10: Common CIFAR-10 distillation settings used with Diff-Instruct.

Item	Setting
Teacher checkpoint	EDM CIFAR-10 unconditional VP checkpoint
Distillation loss	Diff-Instruct `DI_EDMLoss`
Generator optimizer	Adam, learning rate 1×10^{-5}, betas $(0, 0.999)$, $\epsilon = 10^{-8}$
Score-generator optimizer	Adam, learning rate 1×10^{-5}, betas $(0, 0.999)$, $\epsilon = 10^{-8}$
Training batch size	128
EMA half-life	500 kimg
Evaluation metric	FID-50k full

Table 11: CIFAR-10 pruning and repair settings for the compact Diff-Instruct students.

Item	18% compact student	34% compact student
Pruning scope and type	Generator-only width pruning	Generator-only width pruning
Channel selection	Prefix EDM parameter Taylor score	Prefix EDM parameter Taylor score
Target pruning ratio	0.20	0.32
Selected base channels	116	104
Parameter count	45.78M from a 55.73M teacher	36.81M from a 55.73M teacher
Actual parameter reduction	17.86%	33.96%
Protected channel fraction	0.9	0.9
Calibration	8 pruning batches; score batch 8; 2 Diff-Taylor batches	Same
Diff-Taylor score	16 steps with threshold 0.05	Same
Diagnostic bins for pruning checks	$\{1.0, 0.3, 0.09\}$	Same
Repair schedule	12k steps, batch 16, learning rate 2×10^{-4}	Same
Repair-noise configuration	Candidate mode: lognormal; operational repair uses the maximum level over a 16-level log-spaced grid, giving $q_r = \delta_{1.0}$; other levels are diagnostic only	Same
Repair-loss weights	Init-state weight 0; noisy-real weight 1	Same
Validation score	Gen/mean/max weights $1/5/1$; early stopping disabled, so these weights do not enter the repair loss	Same

The CIFAR-10 launcher distinguishes the candidate noise mode, diagnostic bins, and operational repair-loss mode. It sets the candidate mode to lognormal and the repair-loss selector to the maximum level. Consequently, the reported repair update uses the single distribution $q_r = \delta_{1.0}$ rather than sampling or optimizing over all 16 candidate levels. The full 55.734M score generator remains dense and is used only as the auxiliary network during Diff-Instruct training.

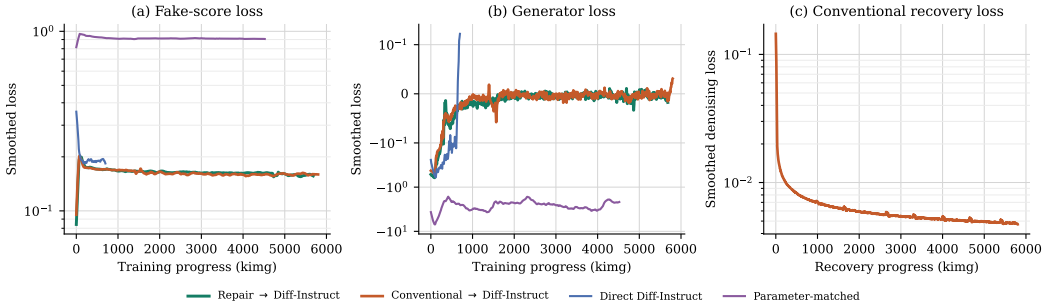

Figure 2: CIFAR-10 training curves. Panels (a) and (b) show the fake-score and generator losses for the Diff-Instruct controls, smoothed over 11 and 17 logged ticks, respectively. Panel (b) uses a symmetric-logarithmic y-axis; its tick labels are powers of ten on both sides of zero. Panel (c) shows the conventional teacher-regression loss smoothed over 401 updates. Loss magnitudes diagnose optimization within each objective; they are not used as cross-objective quality metrics.

Table 12: CIFAR-10 alignment diagnostics before and after repair-only optimization. All values are relative MSE measured with the same compact student and teacher. For outputs G and T, $\mathrm{relMSE}(G,T) = \|G(u) - T(u)\|_2^2 / \|T(u)\|_2^2$, averaged over the reported evaluation samples.

Repair state	Generator rel. MSE	Score rel. MSE mean	Score rel. MSE max
Before repair	11.26197	0.09777	0.20302
After repair	0.07303	0.01022	0.01700

B QUALITATIVE RESULTS



Figure 3: ImageNet-512 samples from our 20% pruned one-step generator. The generator uses 98.826M parameters, one network evaluation, and reaches an FID of 3.12.



Figure 4: Qualitative SD1.5 samples from our repaired compact one-step generator. The model uses 653.7M parameters (24% rounded pruning), one sampling step, and reaches an FID of 18.7.



Figure 5: CIFAR-10 samples from our 34% pruned one-step generator distilled with Diff-Instruct.