PHYSLENSDIFF: VALID-BY-CONSTRUCTION LENSING EMULATION, AND WHY PHYSICS PENALTIES FAIL

Anonymous authors
Paper under double-blind review

ABSTRACT

Simulators obey their equations but are slow; the generative models that emulate them are fast but usually carry no guarantee that their samples obey those equations, especially far from the training data. We study the two standard fixes on strong gravitational lensing, where validity can be checked exactly. The *soft* fix adds the lens equation to the training loss as a penalty. It lowers the violation, but by darkening the images: we prove that the penalty always rewards darkening, and observe that the samples lose their content and stop following the requested lens, in lensing and in divergence-free flows. Reweighting or normalising the penalty does not help. The *hard* fix avoids this. `PhysLensDiff` generates the source, the lens and the dark-matter substructure, then ray-traces them through a differentiable lens layer, so every sample is exactly valid for any requested lens, at $\approx 0.1\%$ of generation time. Unconstrained image generators, even when given the full lens, remain far from valid out of distribution. Our with/without-substructure pairs train dark-matter classifiers that are more accurate on held-out simulated systems than the best alternative from image generators.

1 INTRODUCTION

Generative models are increasingly used as *emulators*: fast surrogates for scientific simulators, trained on their outputs and queried where running the simulator is too slow (Cranmer et al., 2020). An emulator that reproduces the *look* of a simulator's output need not reproduce the equations that produced it, and the gap is invisible exactly where emulators are most useful: away from the training data. Two families of remedies exist. *Soft* constraints add the governing equation to the loss as a penalty, as in physics-informed networks (Raissi et al., 2019) and physics-informed diffusion (Bastek et al., 2025); *hard* constraints restrict the model's output to the feasible set, by projection, reflection or reparameterisation (Christopher et al., 2024; Lou & Ermon, 2023; Liu et al., 2023; Négiar et al., 2023). Soft constraints are cheaper and architecture-agnostic, and are usually assumed to trade a little exactness for that convenience. We show that this assumption can fail in a specific, diagnosable way.

We study the question on strong gravitational lensing, where validity has an exact, checkable meaning. Light obeys the lens equation $\boldsymbol{\beta} = \boldsymbol{\theta} - \boldsymbol{\alpha}(\boldsymbol{\theta})$, so an image is *valid* for a lens if some unlensed source maps to it; the rendering operator is linear in the source, so validity can be tested by a convex program. The application also matters in itself: faint perturbations of lensed arcs by dark-matter subhalos discriminate dark-matter models (Vegetti & Koopmans, 2009; Alexander et al., 2020), ray-tracing simulators (Birrer & Amara, 2018; Toomey, 2022) are slow, and current emulators (Sayed et al., 2025) synthesise the image directly, so the lens equation holds only where training data lies. Yet survey forecasts and inference-network training sets need exactly the lenses a finite simulation covers worst: the tails of the prior.

Contributions. (i) `PhysLensDiff`, an emulator that generates the unlensed source, the lens potential and the substructure density and ray-traces them through a differentiable lens layer, so every sample is valid by construction for *any* requested lens, at $\approx 0.1\%$ of generation time. (ii) A calibrated validity protocol: an exactly solved residual with a convergence certificate, anchored on real data between a matched-strength wrong lens and an image that ignores the requested lens. Under it, dif-

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.

fusion and flow-matching image generators out of distribution reproduce the requested lens's coarse morphology yet, on the raw residual, are as inconsistent with it as images that ignore it (the scale-free residual places them between the real floor and that anchor), and giving the generator the full lens, as the scoring operator's own ray map, does not close the gap. (iii) An analysis of the soft alternative. A lens-equation penalty drives the raw residual as low as ours, but by emptying the image; we prove this is structural (the valid set is convex and contains the empty image, so the penalty's gradient always has a component that darkens its input) and show that the scale-free penalty that removes this drive falls into the other degenerate optimum the geometry predicts: thin, bright, circular rings that no longer respond to the requested lens. Post-hoc projection onto the valid set, the other hard constraint, keeps the content, confirming that the failure is the penalty's and not the constraint's. (iv) Because the model is decomposed, it emits with/without-substructure *counterfactuals*, which image generators can only approximate; classifiers trained on them read the dark-matter labels of held-out simulated systems more accurately than the strongest image-space proxy we tested.

2 Related work

Soft physical constraints. Penalising the residual of a governing equation is the physics-informed recipe (Raissi et al., 2019); its optimisation pathologies are documented for regression (Krishnapriyan et al., 2021; Wang et al., 2021). For generative models, the residual has been added to the diffusion training loss (Bastek et al., 2025) or used at sampling time (Huang et al., 2024; Chung et al., 2023; Jacobsen et al., 2025). These works evaluate the residual itself; we show that for a generator, the residual alone can be satisfied by degenerating the sample, and that a scale-free residual and content statistics are needed to see it. **Hard constraints.** Projected (Christopher et al., 2024), reflected (Lou & Ermon, 2023) and mirror (Liu et al., 2023) diffusion keep samples in a constraint set during sampling; Négiar et al. (2023) enforce PDE constraints exactly through a differentiable solver layer, Chalapathi et al. (2024) scale this approach, and Hansen et al. (2023) project predictions onto a conservation constraint. Richter-Powell et al. (2022) build divergence-free fields by construction, the reparameterisation route; our constraint is of that kind: we do not project the image onto the valid set but generate the latent causes (source and lens) and apply the physical operator, so the valid set is reached by construction and the network never has to learn it. **Learning with inner optimisation.** Differentiating through an inner solve (Amos & Kolter, 2017; Blondel et al., 2022) is costly in general; for our penalty the envelope theorem (Milgrom & Segal, 2002) makes the gradient exact at the inner optimum without unrolling. **Machine learning for strong lensing.** Networks estimate lens parameters (Hezaveh et al., 2017), classify dark-matter substructure (Alexander et al., 2020), and embed the lens equation for *inversion* (Velôso et al., 2023; Ojha et al., 2024); score-based priors reconstruct sources through a known lens (Adam et al., 2022), and differentiable lens-modelling codes fit a source and a lens through a differentiable ray-tracer (Chianese et al., 2020; Galan et al., 2022; Karchev et al., 2022); we use the same kind of operator generatively. On the generative side, FlowLensing (Sayed et al., 2025) emulates the simulator in image space and is our strongest baseline; diffusion models also super-resolve lensing images (Reddy et al., 2024).

3 Validity, and two ways to impose it

Setup. An image is a vector $x \in \mathbb{R}^M$ on a pixel grid, a source is $s \in [0,1]^N$, and a lens ℓ (potential and external shear) defines the ray-tracing operator $\mathcal{R}_\ell : \mathbb{R}^N \to \mathbb{R}^M$, $(\mathcal{R}_\ell s)(\boldsymbol{\theta}) = s(\boldsymbol{\theta} - \boldsymbol{\alpha}_\ell(\boldsymbol{\theta}))$ with bilinear interpolation. Since interpolation is a fixed weighted sum, $\mathcal{R}_\ell$ is *linear* in s. The set of images valid for ℓ is $K_\ell = \mathcal{R}_\ell([0,1]^N)$, and the *validity residual* is the squared distance to it,

$$V_\ell(x) = \min_{s \in [0,1]^N} \|\mathcal{R}_\ell s - x\|^2 = \mathrm{dist}^2(x, K_\ell), \qquad \bar{V}_\ell(x) = V_\ell(x)/\|x\|^2. \tag{1}$$

V_ℓ is a convex box-constrained least-squares problem; we solve it with accelerated projected gradient (Beck & Teboulle, 2009) (step from a power-iteration Lipschitz bound with a $3\times$ margin, 400 iterations) and report the projected-gradient norm as a convergence certificate, so no optimiser budget hides in the numbers. $\bar{V}_\ell$ is its scale-free version; the reason to report both is Proposition 1.

Hard constraint. A generator that outputs a source s and a lens ℓ and returns $x = \mathcal{R}_\ell s$ satisfies $V_\ell(x) = 0$ for every ℓ, including lenses never seen in training: extrapolation in the lens is carried entirely by the operator, which is exact at any lens. For such a model $\bar{V}_\ell$ is a *control* that verifies the

implementation, not a measurement of generalisation; the measurement is how far models without this property fall from zero.

Soft constraint. The cheaper alternative trains an image generator with the residual as a penalty, $\mathcal{L} = \mathcal{L}_{\text{diff}} + \lambda \mathbb{E}\, V_\ell(\hat{x}_0)$, where $\hat{x}_0$ is the model's prediction of the clean image. The gradient needs no differentiation through the inner solver: V_ℓ is the squared distance to a closed convex set, so $\nabla V_\ell(x) = 2(x - P_{K_\ell}(x))$, where the projection $P_{K_\ell}(x) = \mathcal{R}_\ell s^\star$ is unique even when the minimiser $s^\star$ is not (the envelope theorem, Milgrom & Segal, 2002). A few projected-gradient steps without gradient tracking thus give the exact gradient up to inner-solve accuracy, at about 5% extra training cost. In training we truncate the inner solve at 12 steps, so the penalty is an upper bound of V_ℓ evaluated at an approximate projection; every arm is nevertheless scored with the exact 400-iteration fitter. What this penalty rewards is fixed by the geometry of K_ℓ:

Proposition 1 (the penalty is a dimmer)**.** *K_ℓ is compact, convex and contains 0. Consequently, for every image x and every $c \in [0,1]$, $V_\ell(cx) \le c^2\, V_\ell(x)$, and $\langle \nabla V_\ell(x),\, x\rangle \;\ge\; 2\, V_\ell(x)$.*

Proof. K_ℓ is the image of the box $[0,1]^N$ under the linear map $\mathcal{R}_\ell$, hence compact and convex, and $0 = \mathcal{R}_\ell 0 \in K_\ell$. By definition $V_\ell(x) = \min_{y \in K_\ell} \|x - y\|^2$; let $P = P_{K_\ell}(x)$ be the point of K_ℓ closest to x (it exists and is unique because K_ℓ is closed and convex), so that $V_\ell(x) = \|x - P\|^2$.

Scaling. For $c \in [0,1]$, the point $cP = cP + (1-c)\,0$ is a convex combination of P and 0, both in K_ℓ, so it lies in K_ℓ. Since $V_\ell(cx)$ is a minimum over K_ℓ, it is at most the value at this point:

$$V_\ell(cx) \;\le\; \|cx - cP\|^2 \;=\; c^2\, \|x - P\|^2 \;=\; c^2\, V_\ell(x).$$

Gradient. The squared distance to a closed convex set is differentiable, with $\nabla V_\ell(x) = 2(x - P)$. The projection satisfies $\langle x - P,\, y - P\rangle \le 0$ for every $y \in K_\ell$; taking $y = 0$ gives $\langle x - P, P\rangle \ge 0$. Writing $x = (x - P) + P$,

$$\langle \nabla V_\ell(x),\, x\rangle \;=\; 2\|x - P\|^2 + 2\langle x - P,\, P\rangle \;\ge\; 2\, V_\ell(x). \qquad \square$$

The second statement says that the penalty's gradient always has a positive component along the image itself: any descent step on the penalty reduces the image's energy, $\frac{d}{d\eta}\|x - \eta \nabla V_\ell\|^2\big|_{\eta=0} \le -4V_\ell(x) < 0$, unless the image is already valid. On data the bound is tight: on 256 real test images and 512 baseline samples, $V_\ell(x/2)/V_\ell(x) = 0.250$ and $\langle \nabla V_\ell(x), x\rangle / 2V_\ell(x) = 1.000$ (1.008 on one DDPM subset, Appendix D; medians; minimum 0.992, the fitter's tolerance), so the box's upper bound is inactive and V_ℓ is exactly 2-homogeneous. The cosine between the penalty's gradient and the image then equals $\sqrt{\bar{V}_\ell(x)}$: for a baseline's out-of-distribution sample ($\bar{V}_\ell \approx 0.13$) over a third of the gradient's norm points toward dimming. The cheapest way to lower the penalty is therefore always partly to *dim*, and the empty image is a global minimiser for every lens. The raw residual of a soft-constrained model is thus not evidence of validity on its own, and we report $\bar{V}_\ell$ and content statistics beside it. Dividing the penalty by $\|\hat{x}_0\|^2$ removes the dimming drive (for c close to 1, $\bar{V}_\ell(cx) = \bar{V}_\ell(x)$ whenever the box's upper bound is inactive), but its minimisers remain all of $K_\ell \setminus \{0\}$, which contains images that carry no information about the source: a compact source placed on the optical axis is imaged into a thin ring along the lens's tangential critical curve, whatever its internal structure. Both statements concern the geometry of the penalty, not the full training dynamics, in which the denoising loss competes with it; Section 5 tests what they imply.

4 PHYSLENSDIFF

Pipeline. Generation factors into learned components and fixed physical operators (Figure 1). A diffusion model samples an unlensed source s. Three numbers give the macro potential ψ_{nss}: it lies on a linear manifold recovered to 0.08% RMS, so a PCA basis with a smoothed-bootstrap draw replaces a network. A class-conditional diffusion model samples the substructure *density* κ_{sub}, spectrum-calibrated per class on training data (when a host lens is given, as in the utility test, a variant also conditioned on its seven parameters is used; the fully generative pipeline of the validity and realism experiments uses the class-only model); its potential follows by solving $\nabla^2 \psi_{\text{sub}} = 2\kappa_{\text{sub}}$ with an FFT Green's function (exact in Model A: the shipped pair satisfies the equation at correlation 0.9997 in our development study). A differentiable lens layer $\mathcal{R}$ (an exact deflection-field renderer $\boldsymbol{\alpha} = s_\alpha \nabla\psi + \Gamma\boldsymbol{\theta}$, with s_α a fixed deflection scale and Γ the external shear: one bilinear resample, adjustable in strength and shear, reproducing the simulator's own images at median correlation

0.94, Appendix B) ray-traces the source through the summed potential. Since $\mathcal{R}$ implements the lens equation explicitly, the output is valid by construction for any s and ψ; only realism depends on their quality, and the guarantee costs $\approx 0.1\%$ of generation time (Appendix H). Data: Model A, an unpublished Roman-like `lenstronomy` dataset (F062, 128^2) shipping per system the true source, the substructure-free image, potentials, densities and a dark-matter label; training uses the official 30,000-system split, every evaluation the disjoint 6,000-system test split, at 64^2 (Appendix A). All images are simulated; "real" denotes a Model A image as opposed to a generated one.

Calibration of the residual. The residual of Section 3 is calibrated on real test images with deliberately wrong lenses (Section 5), using the *same* free-source metric as the results. Calibrating with the true source held fixed instead would overstate the metric's range by more than an order of magnitude ($269\times$ for an unlensed image), because a free source can absorb most of a wrong lens.

Baselines. Three unconstrained generators synthesise the image directly. A lens-conditioned DDPM (Ho et al., 2020) and our reimplementation of FlowLensing (Sayed et al., 2025) (flow matching (Lipman et al., 2023) on a diffusion transformer (Peebles & Xie, 2023) in the published architecture, 30.2M parameters, classifier-free guidance (Ho & Salimans, 2022) weight 2, 100 Euler steps, the released code's peak learning rate) are conditioned on the seven lens parameters of each system, as in the literature. Because the scoring operator reads the full potential, a reviewer may object that these baselines fail for lack of information; the third baseline removes the objection. The *ray-map DDPM* receives, concatenated to its noisy input, the two-channel map $\boldsymbol{\theta} - \boldsymbol{\alpha}_\ell(\boldsymbol{\theta})$ that the scoring operator itself uses, so it is told exactly where each pixel's light must come from, at the requested lens, in and out of distribution. Our own source generator is swappable: `PhysLensDiff`-S uses a 3.9M U-Net DDPM and `PhysLensDiff`-L the *same* 30.2M DiT as FlowLensing, so the strongest baseline is compared at matched capacity. Every model receives a matched update budget (about 14–15k updates, 7.5k for the 128^2 substructure model) on the same 30,000 systems; the published 70k-update schedule exceeds our hardware. Finally, the soft-constraint baselines are the conditional DDPM trained with the penalty of Section 3, with the raw or the scale-free residual (Section 5).

5 Results

Validity out of distribution, against a calibrated anchor. Every model is scored under the *same* operator (the deflection field of the system's own potential) and the same exact fitter. Two anchors on real test images give the numbers meaning: real images against their own lens set the floor (0.00092), and against the *OOD-perturbed* lens they measure what ignoring the perturbation scores ($12.8\times$, $[11.4, 14.3]$). In distribution the DDPM sits *at* the real floor and FlowLensing near it (0.00080 and 0.00125): the story is about extrapolation. Out of distribution ($\theta_E \times 1.6$ plus shear $+0.28$, identical for all), the DDPM degrades by $12.1\times$ $[10.9, 13.4]$ and FlowLensing by $11.4\times$ $[10.3, 12.6]$, both overlapping with the anchor: *on the raw residual, their OOD images are as inconsistent with the requested lens as images made for the unperturbed one*. The raw ratio saturates there: a real image of an *unrelated* system, scored under the requested lens, gives $12.4\times$ $[11.1, 13.8]$, so on this measure the baselines are simply not a lensing through that lens. The scale-free residual does not saturate and places them between the real floor and the anchor (0.126 and 0.114 against 0.035 and 0.210; Tables 1 and 2, Figure 2a). They are not unresponsive: their ring radius and the ellipticity imprinted by the requested shear follow the request ($\Delta e_1 = +0.44$ for the DDPM against $+0.44$ for the true sources rendered through the requested lens, Table 2). The images look like the requested lens without being a lensing of anything through it. Missing information is not the explanation: the ray-

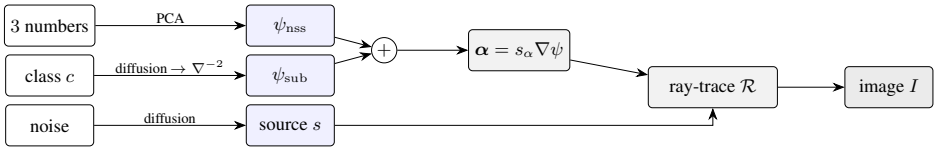


Figure 1: The pipeline. Blue boxes are generated, grey boxes are fixed operators; nothing is drawn from the dataset at inference. The lens equation is enforced by the operator rather than learned, so validity holds for *any* s and ψ; only realism depends on their quality.

Table 1: Fixed-lens validity residual (MSE), $n = 512$, one operator and one exact fitter for all models, bootstrap 95% intervals. The anchors are measured on real test images and bracket the scale.

Model	In-dist	OOD	ratio	95% CI
anchor: real images, own lens (floor)	0.00092	–	1	–
anchor: real images, OOD lens (ignores perturbation)	–	0.01176	12.8	[11.4, 14.3]
`PhysLensDiff-S` (ours)	$< 10^{-5}$	$< 10^{-5}$	*control (zero by construction)*	
`PhysLensDiff-L` (ours, FlowLensing's DiT)	$< 10^{-9}$	$< 10^{-9}$	*control (zero by construction)*	
Conditional DDPM	0.00080	0.00972	12.1	[10.9, 13.4]
FlowLensing (DiT + flow matching)	0.00125	0.01427	11.4	[10.3, 12.6]
Ray-map DDPM (full lens as input)	0.00074	0.00928	12.5	[11.3, 13.9]

Deflection operator of each system's own potential, exact FISTA fitter (median certificate $\leq 7 \times 10^{-3}$). The operator reads the full potential while baselines see seven parameters; this is identical at both regimes and cancels in the ratio. A scale-free variant gives 5.6, 3.6 and 6.8 against an anchor of 6.1: the DDPM at the anchor, FlowLensing below it, the ray-map DDPM above it. OOD $= \theta_E \times 1.6$, shear $+0.28$ (training range $[-0.15, 0.15]$). FlowLensing is our reimplementation at an equal update budget.

map DDPM, which receives at every pixel the source-plane position the scoring operator itself uses, degrades by $12.5\times$ [11.3, 13.9]. In distribution it is the most consistent generator we trained (scale-free residual 0.020, below the 0.035 of real images, which carry photon noise), so it does use the map; out of distribution it over-responds, with ring radius $\times 1.56$ and shear imprint $+0.59$ against $\times 1.37$ and $+0.44$ for the target. Conditioning on the lens, however complete, teaches the network the lens-to-image map only where training lenses lie. Sweeping the perturbation from $1.0\times$ to $2.0\times$, their degradation is monotone (Figure 2b) while our residual stays at zero to solver precision throughout. This is a *control*, not a measurement: our sources are rendered through the scoring operator, so a source reproducing them exists by construction; the measurement is the baselines' rows. This also rules out a capacity confound: the same DiT architecture that loses $11.4\times$ of validity when it generates the *image* keeps exact validity when it generates the *source* (`PhysLensDiff-L`, measured at $< 10^{-9}$ at both regimes). The guarantee is carried by the constraint, not the architecture; without guidance passes, the constrained variant is also twice as fast (Figure 2d). Figure 3 shows validity and appearance coming apart at the extreme lens.

The cheaper alternative: a soft constraint. Would an image generator trained with the lens equation as a *penalty* do? We give this baseline every advantage: the conditional DDPM above, identical in architecture, seed, budget and conditioning, plus the penalty $\lambda V_\ell(\hat{x}_0)$ of Section 3 computed with the system's *true* potential, which our own model never sees at inference. We sweep λ from 1 to 100 (Table 2, Figure 4). On the raw residual the penalty seems to work: the OOD degradation falls from $12.1\times$ to about $1\times$. The scale-free residual disagrees and climbs to the level of the ignore-the-lens anchor (0.22 against 0.21). The samples show why (Figure 5). From $\lambda = 30$ they are thin circular rings with no source structure, the easiest image for *some* source to explain under any lens. Out of distribution they keep a fifth of their energy at $\lambda = 100$, and their response to the requested shear falls from $+0.44$, the value of the physical target, to $+0.04$.

Variants of the penalty. Section 3 predicts that dividing the penalty by $\|\hat{x}_0\|^2$ removes the dimming but keeps uninformative minimisers, and that is what we observe: at $\lambda = 1$ the samples are bright thin rings that stay circular under the requested shear ($\Delta e_1 = +0.02$), and at $\lambda = 0.1$ the penalty barely acts. Weighting the penalty by $\bar{\alpha}_t$, so that it acts where $\hat{x}_0$ is sharp, gives the best trade-off we measured (scale-free OOD residual 0.104, second only to the nearly inactive $\lambda = 0.1$ arm) but only half the requested shear imprint. An explicit energy-conservation term blocks the dimming and produces compact blobs with no lensed structure, invalid even in distribution. None of these settings reaches validity without losing content or obedience. The difference is one of kind: the penalty moves the *image* toward whatever a fitted source can explain, whereas the hard constraint keeps the source and changes only the lens, so our OOD samples keep their energy, arc energy and peak to within 3%. Appendix C shows the same mechanism when the residual trains an inversion network.

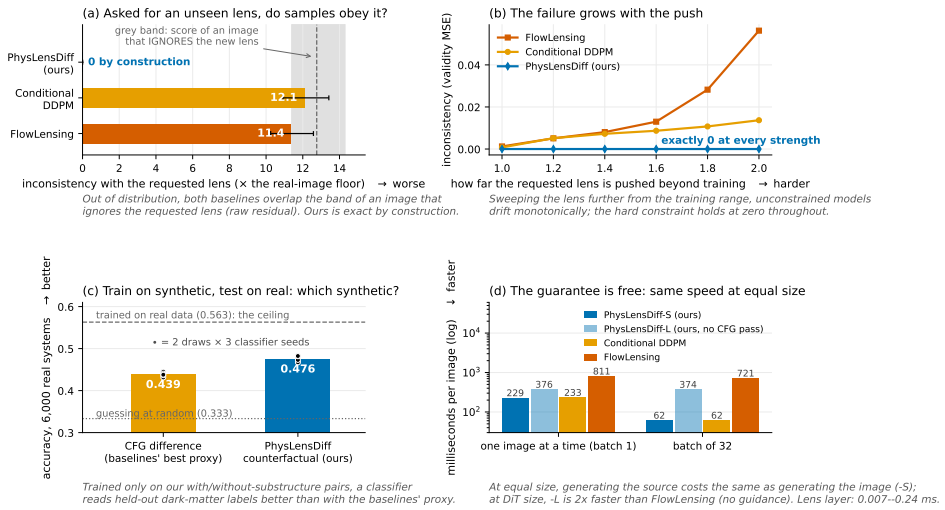


Figure 2: The results at a glance. **(a)** Asked for a lens outside the training range, how inconsistent with it are the samples? Both baselines overlap the grey band: the score of an image that ignores the request outright. **(b)** The same failure as a curve (a separate sampling run, hence values close to but not equal to Table 1): the further the requested lens is pushed, the worse their images fit it, while the hard constraint holds at zero. **(c)** Downstream utility: classifiers trained only on synthetic data, tested on 6,000 real systems; our counterfactual substrate beats the baselines' strongest proxy we tested on every seed of two paired draws. **(d)** The constraint is free: the lens layer costs 0.007–0.24 ms, and sampling the source costs the same as sampling the image with an equal-size network; `PhysLensDiff`-L, without guidance passes, is twice as fast as FlowLensing. The full pipeline, with the 128^2 substructure, is $5\times$ slower than the image DDPM (Appendix H).

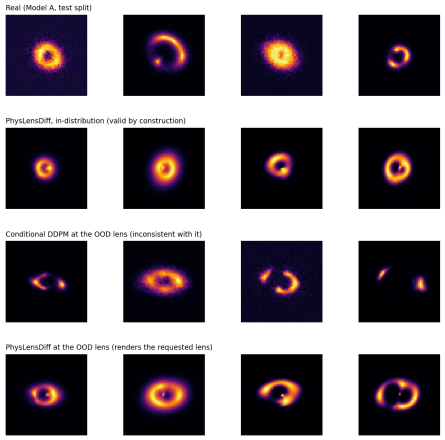


Figure 3: Validity is not appearance: at the OOD lens the baseline emits arcs that follow the requested lens's shape but are not a lensing of any source through it (row 3); ours renders the requested extreme lens and stays valid by construction (row 4).

The other hard constraint: post-hoc projection. Proposition 1's projection suggests a second hard constraint that needs no new model: keep the unconstrained generator and replace each sample by $P_{K_\ell}(x) = \mathcal{R}_\ell s^\star$, which the exact fitter already computes. It is valid by construction, and it works. Projecting the DDPM's OOD samples removes 12.6% of their energy (2.3% in distribution) and keeps both their arc energy (1.05 of in-distribution) and the requested shear imprint ($+0.41$ against $+0.44$; Table 2). The two hard constraints thus agree on what the soft one gets wrong: holding the sample *in* the valid set with an operator preserves content, rewarding it for approaching the set does not. They differ in three respects. The projection solves a 400-iteration convex program

Table 2: `PhysLensDiff` (shaded) is exactly valid and reproduces the physical target's response to the request out of distribution: it keeps its arc energy (1.02 against 1.04) and matches the shear imprint (+0.438 against +0.438). Post-hoc projection is also valid and comes close; every soft-constraint setting trades validity against content and obedience. Same protocol as Table 1 ($n = 512$, one operator, one exact fitter). The physical target renders the *true* test sources through the *true* lens, nominal and perturbed.

	validity residual, OOD			image content		
Model	raw	ratio [CI]	scale-free	arc en. in / real	arc en. OOD / in	shear imprint Δe_1
physical target (true source, requested lens)	–	–	–	$1.77^{\dagger}$	1.04	+0.438
`PhysLensDiff` (ours, hard)	$< \mathbf{10^{-5}}$	*exact*	$< \mathbf{10^{-5}}$	$1.92^{\dagger}$	**1.02**	**+0.438**
DDPM + post-hoc projection (hard)	0	*exact*	0	0.91	1.05	+0.409
FlowLensing + post-hoc projection (hard)	0	*exact*	0	1.07	1.60	+0.484
DDPM, no constraint	0.00972	12.1 [10.9, 13.4]	0.126	0.96	1.13	+0.442
FlowLensing, no constraint	0.01427	11.4 [10.3, 12.6]	0.114	1.14	1.70	+0.506
ray-map DDPM, no constraint	0.00928	12.5 [11.3, 13.9]	0.135	0.99	0.97	+0.586
soft, $\lambda = 1$	0.00372	4.80 [4.37, 5.26]	0.125	0.98	0.94	+0.422
soft, $\lambda = 3$	0.00159	2.13 [1.99, 2.29]	0.121	0.83	0.74	+0.363
soft, $\lambda = 10$	0.00079	1.29 [1.18, 1.41]	0.157	0.59	0.46	+0.240
soft, $\lambda = 30$	0.00059	1.00 [0.89, 1.13]	0.223	0.54	0.26	+0.086
soft, $\lambda = 100$	0.00070	0.90 [0.77, 1.04]	0.224	0.68	0.20	+0.040
soft, scale-free, $\lambda = 0.1$	0.00462	6.57 [5.71, 7.53]	0.102	1.17	0.98	+0.280
soft, scale-free, $\lambda = 1$	0.00255	2.86 [2.62, 3.13]	0.259	0.77	0.65	+0.023
soft, $\bar{\alpha}_t$-weighted, $\lambda = 30$	0.00128	2.28 [2.08, 2.51]	0.104	0.80	0.66	+0.232
soft, $\bar{\alpha}_t$-weighted, $\lambda = 100$	0.00117	1.77 [1.59, 1.98]	0.185	0.71	0.43	+0.047
soft + energy term, $\lambda = 30$, $\mu = 10$	0.01112	2.76 [2.61, 2.92]	0.286	1.08	1.57	+0.009
anchor: real images, OOD lens	0.01176	12.8 [11.4, 14.3]	0.210	1	–	–

Scale-free = residual divided by the image's mean squared intensity, averaged over images; the real-image floor is 0.035. Arc energy is the mean squared intensity above the image median, so it ignores the background pedestal. Δe_1 is the change, from the nominal to the OOD request, of the light's second-moment ellipticity: the direct imprint of the requested external shear, bootstrap half-widths ≤ 0.014. †Noise-free renders carry no photon noise and no pedestal, so their energy above background exceeds the real images'; our model matches the noise-free target.

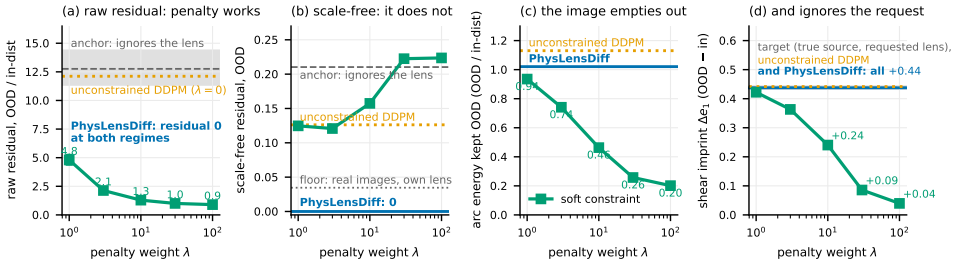


Figure 4: The soft constraint trades content for residual as the penalty weight λ grows. **(a)** Raw OOD/in-distribution validity ratio: the soft model reaches about 1, no degradation. **(b)** The same residual divided by the image's energy climbs to the ignore-the-lens anchor. **(c)** The energy above background kept out of distribution, relative to the same model in distribution: the images empty out. **(d)** A measure independent of the residual: the ellipticity change that the requested shear imprints on the image. The physical target, the unconstrained DDPM and ours all respond with +0.44; the soft model stops responding. Error bars and shaded bands are bootstrap 95% intervals.

per sample (149 ms per image at batch 1 and 5.4 ms at batch 32, about 65% and 9% of generation time), where our constraint is one resample (0.12 and 0.012 ms in the same benchmark). Its source $s^{\star}$ is whatever reproduces the image: on these noisy images it spreads light over 63% of the source plane where true sources light 5% (the same fitter on real images lights 80%, so this is the cost of fitting noise, not a defect of the generator), and it cannot be recombined with another lens or another substructure realisation. And it inherits the base generator's errors: projected FlowLensing samples keep its excess OOD arc energy (1.60) and over-elongation (+0.48). Generating the latent causes is the only one of the three routes whose source and lens can be recombined, which is what the counterfactual below requires.

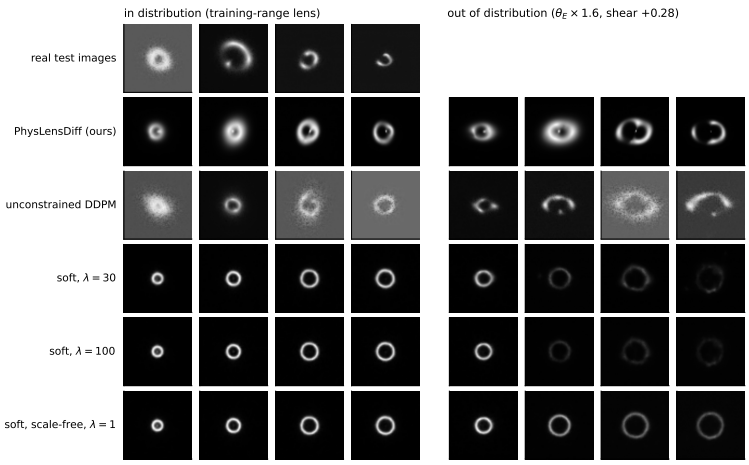


Figure 5: Same systems, same conditioning, four examples each. Left: training-range lens; right: the OOD lens. From $\lambda = 30$, or at $\lambda = 1$ with the scale-free penalty, the soft model collapses in distribution to thin circular rings without source structure, and out of distribution to rings that ignore the requested shear or to faint ghosts of them: a small residual obtained by generating a small image. Ours renders the requested extreme lens with the source held fixed.

Other interventions. The conclusions do not depend on the particular request (Appendix F). We repeat the protocol for shear alone (± 0.28), for $\theta_E \times 1.6$ alone, which stays within the training range for 96% of systems, and for $\theta_E \times 2.5$, beyond it for 54%; the original intervention is reproduced within sampling noise ($12.2\times$ for the DDPM). Our model is exact and matches the physical target's response in every case (shear imprint $+0.471$ against $+0.472$, ring radius $\times 1.73$ against $\times 1.72$). The unconstrained generators fail in proportion to how far the request leaves the training range: mildly for the in-range θ_E (2.8–$3.8\times$ against an anchor of $5.7\times$), at or above the anchor for shear, and in between for $\theta_E \times 2.5$, where FlowLensing alone exceeds the anchor. The soft penalty keeps its signature everywhere: the raw-penalty arm pays with energy (down to 0.16 of its in-distribution value, with a raw residual of $0.4\times$) and the scale-free arm with obedience (shear imprint $+0.02$ against $+0.47$).

What the diagnostic can and cannot see. We calibrated the metric before trusting it, on real test images with deliberately wrong lenses, including lenses from *other* simulators. On this scale, a lens from another system with the Einstein radius matched within 5% costs $1.04\times$ $[1.02, 1.07]$; a random lens $2.02\times$, almost entirely an Einstein-radius effect; ignoring our OOD perturbation $12.6\times$ on this 192-system subset ($12.8\times$ at $n = 512$). The metric reads lens *strength* and is nearly blind to *geometry* at matched strength. Three remedies (regularised, coherent and parametric sources) fail to restore that sensitivity, and an oracle test shows why: with the *true* source held fixed, a matched-strength lens from another system changes the discrepancy by only 3% $[1\%, 6\%]$. The estimators were already at the information ceiling; the near-degeneracy lives in the images themselves (Schneider & Sluse, 2014). The diagnostic is also itself a forward model: scoring each model under the operator it is conditioned on lets a misspecified analytic profile compress the baselines' degradation to 2.2–$2.4\times$ (Appendix B). Hence the final protocol: one operator for every model, anchors measured on real data under that operator, the fit solved exactly.

Generation quality. Our renders are noise-free, so a strict two-sample CNN critic tells them from real images at accuracy 0.99 (0.50 means indistinguishable). An observation layer learned on training pairs (clean render of the true source through the true lens, observed image) brings this to 0.56 and 0.57 on two draws, within about 0.02 of the image DDPM (0.54 and 0.56), and leaves the observed images as consistent with their lens as real ones (scale-free residual 0.027, against 0.032 for $1{,}024$ real test images; Appendix G). The validity results score the clean renders, which are exact. FlowLensing's in-distribution samples are clean and diverse, so its OOD failure points to extrapolation rather than undertraining, although our reimplementation has 15k of the published 70k updates.

The science signal is a counterfactual, which a decomposed model emits directly. The task that motivates this line of work is reading the dark-matter class from lensing images. On real data the raw image carries only a weak class signal (0.390 even with $30{,}000$ training systems; the substructure-free image, a negative control, is at chance (0.333), so no label leaks through the macro lens), while the substructure residual (not the validity residual of Section 3: the difference between the *same* system rendered with and without subhalos) carries four times more signal above chance (0.563, against 0.390 for the raw image; $n = 6{,}000$). That quantity is a *counterfactual*, not a function of the image: a generator that emits images can only approximate it (below), whereas our decomposition renders the pair by construction, for any perturbation and without retraining.

We measure whether this structural advantage survives contact with the strongest available proxy: we retrain the conditional DDPM with the class in its conditioning (with an explicit presence flag and dropout at the published rate of 0.1) and take its paired classifier-free-guidance difference: conditional minus unconditional sample from the *same* initial noise under deterministic sampling. Classifiers are then trained *only on generated* substrates and evaluated on the $6{,}000$ real test residuals, three classifier seeds each, on two generative draws that share hosts, classes and initial noise (the pairs are rendered on 900 training hosts with their true source and lens, so the test isolates the substructure model): ours reaches $\mathbf{0.470}$ and $\mathbf{0.482}$ against 0.440 and 0.438 for the guidance difference (disjoint seed ranges in both draws, Figure 2c), and 0.563 for the real-data reference. Conditioning the substructure on its host matters: the same pipeline with a class-only density model reaches 0.441 and 0.468, level with the guidance difference in the first draw. Generating the substructure as a *density* and solving the Poisson equation, rather than generating its potential, is what makes the counterfactual useful at all: in our development study ($5{,}400$ systems, $n = 600$), the same pipeline generating the potential directly reached 0.418 against 0.478 for the density route, because in potential space the k^{-4} Green's-function suppression hides small-scale power from the training loss. For the class-only model, what still separated us from the reference was spectral: the image-space substructure residual remains too steep ($-4.3/-3.7/-3.6$ against $-2.5/-2.1/-2.0$ for axion/CDM/WDM, measured for the class-only model), so the class *ordering* transfers while the small-scale statistics that physically separate the models are only partly recovered (Appendix I). Adding the generated substructure costs nothing in validity: $\psi = \psi_{\rm nss} + \psi_{\rm sub}$ is an exact decomposition and deflections are additive.

6 Discussion and limitations

Beyond lensing. Proposition 1 needs only a convex valid set containing zero, which every linear homogeneous constraint $Ax = 0$ provides, with equality. We test it on divergence-free 2D velocity fields conditioned on a wavenumber, scored with the exact (Leray) projection (Appendix E). The prediction holds: raising the penalty weight from 1 to 100 cuts the out-of-distribution residual about $145\times$, while the fields keep 15% of the data's energy and 0.56 of the requested wavenumber; in distribution nothing changes. The same experiment bounds our remedy: a stream-function generator is exactly valid but, asked for an unseen wavenumber, returns $15\times$ too much energy. In lensing the hard constraint keeps content because the operator carries the request (the source is fixed, only the lens changes); where the network must extrapolate the latent itself, exactness is all it buys.

Limitations. The method borrows nothing at inference (source, potential and density are all generated) but it requires simulator intermediates at *training* time: true sources to supervise the source model, and potentials and densities to fit the lens models on. Model A required an instrumented simulator, unlike the public one (Toomey, 2022; Alexander et al., 2023); this is a tool for accelerating a simulator one controls, not one trainable on observed catalogues. Several limitations remain. The classification input, the substructure residual, is a simulator counterfactual: it is available for simulated systems, not observable on the sky. All lensing results use one simulated dataset, and we do not test sampling-time guidance (Chung et al., 2023) as a soft alternative, nor per-step projection (Christopher et al., 2024) as a hard one. Bootstrap intervals cover sampling variance only; a second training seed reproduces the three decisive arms (Table 5). The validity diagnostic reads lens strength and cannot certify lens geometry at matched Einstein radius, an information ceiling of the images that bounds what "valid" can mean here. The class-only substructure model's image-space spectrum stays too steep: the power is present but not concentrated near the ring. Realism relies on a learned observation layer: the noise-free renders alone are easily told apart from real images, and the simulator's own observation step is not reproduced exactly.

AI USE STATEMENT

LLMs are not part of the method or the data. The authors conceived the method and the research questions, wrote part of the code and of the text, and made every design decision. LLM-based tools, used under their direction, helped run experiments, proposed parts of the evaluation protocol, and helped draft and edit the text. The authors checked every experiment, number and claim and take full responsibility for them.

ETHICS STATEMENT

This work uses simulated strong-lensing images only (Model A, an unpublished dataset generated with the public `lenstronomy` and `pyHalo` libraries through an instrumented pipeline); it involves no personal data, no human subjects and no observational data with access restrictions. We see no dual-use concern: the method emulates a physical forward model for a scientific task. The main risk we identify is scientific rather than societal, namely over-trusting an emulator whose samples are exactly valid for the potential they are given but only as realistic as that potential; the paper states this and reports both raw and scale-free validity so that a generator that satisfies the constraint by emptying the image cannot pass unnoticed.

REPRODUCIBILITY STATEMENT

All experiments use the official Model A splits (30 000 train, 6 000 test), fixed random seeds stated in the appendix, and the same exact convex fitter, the same deflection operator and the same out-of-distribution intervention for every model. The appendix specifies architectures, update budgets, preprocessing and evaluation protocols, and every confidence interval is a percentile bootstrap over the n images ($4{,}000$ resamples); the utility classifiers use seeds 0, 1 and 2. The training, generation, scoring and dissection scripts and the twenty-two checkpoints used for the tables (about 0.5 GB) will be released upon acceptance. Model A is not yet public; the data are not ours to release, and the splits will be shared once its creators publish it.

REFERENCES

Alexandre Adam, Adam Coogan, Nikolay Malkin, Ronan Legin, Laurence Perreault-Levasseur, Yashar Hezaveh, and Yoshua Bengio. Posterior samples of source galaxies in strong gravitational lenses with score-based priors. *arXiv preprint arXiv:2211.03812*, 2022. NeurIPS 2022 Workshop on Machine Learning and the Physical Sciences.

Stephon Alexander, Sergei Gleyzer, Evan McDonough, Michael W. Toomey, and Emanuele Usai. Deep learning the morphology of dark matter substructure. *The Astrophysical Journal*, 893(1): 15, 2020. doi: 10.3847/1538-4357/ab7925.

Stephon Alexander, Sergei Gleyzer, Hanna Parul, Pranath Reddy, Marcos Tidball, and Michael W. Toomey. Domain adaptation for simulation-based dark matter searches with strong gravitational lensing. *The Astrophysical Journal*, 954(1):28, 2023. doi: 10.3847/1538-4357/acdfc7.

Brandon Amos and J. Zico Kolter. OptNet: Differentiable optimization as a layer in neural networks. In *International Conference on Machine Learning (ICML)*, 2017.

Jan-Hendrik Bastek, WaiChing Sun, and Dennis M. Kochmann. Physics-informed diffusion models. In *International Conference on Learning Representations (ICLR)*, 2025.

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

Simon Birrer and Adam Amara. lenstronomy: Multi-purpose gravitational lens modelling software package. *Physics of the Dark Universe*, 22:189–201, 2018. doi: 10.1016/j.dark.2018.11.002.

Mathieu Blondel, Quentin Berthet, Marco Cuturi, Roy Frostig, Stephan Hoyer, Felipe Llinares-López, Fabian Pedregosa, and Jean-Philippe Vert. Efficient and modular implicit differentiation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Nithin Chalapathi, Yiheng Du, and Aditi S. Krishnapriyan. Scaling physics-informed hard constraints with mixture-of-experts. In *International Conference on Learning Representations (ICLR)*, 2024.

Marco Chianese, Adam Coogan, Paul Hofma, Sydney Otten, and Christoph Weniger. Differentiable strong lensing: uniting gravity and neural nets through differentiable probabilistic programming. *Monthly Notices of the Royal Astronomical Society*, 496(1):381–393, 2020.

Jacob K. Christopher, Stephen Baek, and Ferdinando Fioretto. Constrained synthesis with projected diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Hyungjin Chung, Jeongsol Kim, Michael T. McCann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations (ICLR)*, 2023.

Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020.

Aymeric Galan, Georgios Vernardos, Austin Peel, Frédéric Courbin, and Jean-Luc Starck. Using wavelets to capture deviations from smoothness in galaxy-scale strong lenses. *Astronomy & Astrophysics*, 668:A155, 2022.

Derek Hansen, Danielle C. Maddix, Shima Alizadeh, Gaurav Gupta, and Michael W. Mahoney. Learning physical models that can respect conservation laws. In *International Conference on Machine Learning (ICML)*, 2023.

Yashar D. Hezaveh, Laurence Perreault Levasseur, and Philip J. Marshall. Fast automated analysis of strong gravitational lenses with convolutional neural networks. *Nature*, 548:555–557, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 6840–6851, 2020.

Jiahe Huang, Guandao Yang, Zichen Wang, and Jeong Joon Park. DiffusionPDE: Generative PDE-solving under partial observation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Christian Jacobsen, Yilin Zhuang, and Karthik Duraisamy. CoCoGen: Physically-consistent and conditioned score-based generative models for forward and inverse problems. *SIAM Journal on Scientific Computing*, 47(2):C399–C425, 2025. doi: 10.1137/24M1636071.

Konstantin Karchev, Adam Coogan, and Christoph Weniger. Strong-lensing source reconstruction with variationally optimized Gaussian processes. *Monthly Notices of the Royal Astronomical Society*, 512(1):661–685, 2022.

Aditi S. Krishnapriyan, Amir Gholami, Shandian Zhe, Robert M. Kirby, and Michael W. Mahoney. Characterizing possible failure modes in physics-informed neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

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

Guan-Horng Liu, Tianrong Chen, Evangelos A. Theodorou, and Molei Tao. Mirror diffusion models for constrained and watermarked generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

David Lopez-Paz and Maxime Oquab. Revisiting classifier two-sample tests. In *International Conference on Learning Representations (ICLR)*, 2017.

Aaron Lou and Stefano Ermon. Reflected diffusion models. In *International Conference on Machine Learning (ICML)*, 2023.

Paul Milgrom and Ilya Segal. Envelope theorems for arbitrary choice sets. *Econometrica*, 70(2): 583–601, 2002.

Geoffrey Négiar, Michael W. Mahoney, and Aditi S. Krishnapriyan. Learning differentiable solvers for systems with hard constraints. In *International Conference on Learning Representations (ICLR)*, 2023.

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8162–8171, 2021.

Ashutosh Ojha, Sergei Gleyzer, Michael W. Toomey, and Pranath Reddy Kumbam. LensPINN: Physics informed neural network for learning dark matter morphology in lensing. In *NeurIPS Machine Learning and the Physical Sciences Workshop*, 2024.

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

Maziar Raissi, Paris Perdikaris, and George E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

Pranath Reddy, Michael W. Toomey, Hanna Parul, and Sergei Gleyzer. DiffLense: a conditional diffusion model for super-resolution of gravitational lensing data. *Machine Learning: Science and Technology*, 5(3):035076, 2024. doi: 10.1088/2632-2153/ad76f8.

Jack Richter-Powell, Yaron Lipman, and Ricky T. Q. Chen. Neural conservation laws: A divergence-free perspective. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations (ICLR)*, 2022.

Hamees Sayed, Pranath Reddy, Michael W. Toomey, and Sergei Gleyzer. FlowLensing: Simulating gravitational lensing with flow matching. In *NeurIPS Machine Learning and the Physical Sciences Workshop*, 2025. arXiv:2510.07878.

Peter Schneider and Dominique Sluse. Source-position transformation: an approximate invariance in strong gravitational lensing. *Astronomy & Astrophysics*, 564:A103, 2014.

Michael W. Toomey. DeepLenseSim: strong-lensing image simulation for the DeepLense project. `https://github.com/mwt5345/DeepLenseSim`, 2022. built on `lenstronomy` and `pyHalo`.

Simona Vegetti and Léon V. E. Koopmans. Bayesian strong gravitational-lens modelling on adaptive grids: objective detection of mass substructure in galaxies. *Monthly Notices of the Royal Astronomical Society*, 392(3):945–963, 2009.

Lucas J. Velôso, Michael W. Toomey, and Sergei Gleyzer. Lensformer: A physics-informed vision transformer for gravitational lensing. In *NeurIPS Machine Learning and the Physical Sciences Workshop*, 2023.

Sifan Wang, Yujun Teng, and Paris Perdikaris. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM Journal on Scientific Computing*, 43(5):A3055–A3081, 2021.

A note on scale. All numbers in the main text come from the final run or later experiments on its checkpoints, except those it marks as from the development study (the $269\times$ calibration, the operator comparison, the calibration ladder's remedies and oracle, the Poisson check and the potential-space utility of 0.418): training on the official $30{,}000$-system split, evaluation on the disjoint $6{,}000$-system test split, exact fitter throughout. The appendices on the source generator, the renderer, training on the residual and substructure generation document the *development* study at $5{,}400$ systems in which the protocol, renderers and diagnostics were designed; the appendices on additional baselines, other

interventions, the second domain, realism and costs report final-run or new experiments; the validity, intervention and realism experiments use the class-only substructure model, the utility results the host-conditioned one, and validity holds by construction for either; where a number differs between an appendix and the main text, the main text supersedes it. Renderer validation (Appendix B) and the speed benchmark (Appendix H) are scale-independent properties of the operators and hardware.

A Setup and source generator

Units. The macro pipeline operates on arcsinh-stretched, per-image min–max normalised maps, a common input convention for this data. Ray-tracing resamples surface brightness pointwise, $I(\boldsymbol{\theta}) = s(\boldsymbol{\beta}(\boldsymbol{\theta}))$, so a monotone map commutes with it *exactly* only under nearest-neighbour resampling. Ours interpolates bilinearly, and a linear interpolation between neighbouring pixels does not commute with a non-linear stretch, so working in compressed brightness is an approximation and not an identity. Two further mismatches are exact rather than approximate: the per-image normalisation constant differs between source and image, which the flux calibration below absorbs, and the stretch scale is set per map. We keep the convention because it is what the source model trains on, and we flag it as the reason the generative half is not yet a likelihood-grade forward model. The renderer validation (Appendix B) and the substructure and dark-matter experiments (Appendix I) instead use the raw linear fields, where none of this applies.

Source generator. The source model is a compact time-conditioned U-Net trained as a v-prediction diffusion model (Ho et al., 2020; Salimans & Ho, 2022) with a cosine schedule (Nichol & Dhariwal, 2021) and an EMA, sampled with 100 DDIM steps. On these sparse sources (a compact Sersic on a mostly black field) plain ϵ-prediction drifts to an empty source and min-SNR or brightness weighting over-spreads it, whereas v-prediction is stable and yields 0% black samples. Before rendering we canonicalise the source position by shifting its flux centroid to the frame centre. For the realism experiments of Appendix G we also calibrate each sample's flux to a draw from the training source-flux distribution; the validity results use uncalibrated generated sources, which leaves them unchanged (the constraint holds at any flux), and the utility test renders the true sources of its hosts. Sources are generated, rendered and scored at 64^2; the substructure density is generated at 128^2 and resampled. The deflection scale is $s_\alpha \approx 0.016$ at $0.11''$/pixel. In the final run every learned component (source model, both baselines, the class-conditional baseline and the two substructure density models) receives a matched optimiser-update budget (about 14k updates at 64^2, 15k for the diffusion transformers, 7.5k for the 128^2 substructure density models) on the same $30{,}000$ training systems; the host-conditioned one takes the three class indicators and the host's seven standardised lens parameters as input, with the same calibration; the substructure density is normalised by a global arcsinh (never per image: amplitude is signal), its generated amplitude spectrum is calibrated per class against the training set, and its potential follows from an FFT Poisson solve with zero-padding, whose unit scale is measured by regressing shipped deflections on solved ones.

B Renderer validation

Renderer fidelity has an unambiguous ground truth: rendering a system's true source through its own lens should reproduce that system's simulated image. On 450 held-out systems the exact deflection-field renderer attains a median per-system pixel correlation of 0.94 (mean 0.90) and wins on $433/450$ against the analytic renderer the baselines are conditioned on, which reaches 0.79 (mean 0.74) and breaks high-ellipticity systems into disconnected arcs (Figure 6).

That analytic operator is, however, a poor version of the profile the data were made with, and the gap above is partly its own fault rather than a limit of analytic modelling. It uses $q = 1-2|e|$ instead of $q = (1-|e|)/(1+|e|)$, adds a $0.15''$ core to a singular profile, is pseudo-elliptical rather than a true isothermal ellipsoid, and takes its field of view from an argmax that overshoots the known $0.11''$ pixel scale by 7.4%. Replacing it with the singular isothermal ellipsoid as implemented in `lenstronomy` (Birrer & Amara, 2018), correctly parameterised and with no core, raises the median correlation from 0.79 to 0.90 (mean 0.74 to 0.85), recovering 68% of the distance to the exact renderer. We keep the operator the baselines were trained and scored under so that the comparison

stays internally consistent, and we report this measurement because it sizes how much of the analytic renderer's disadvantage is intrinsic and how much is our specification of it.

The corrected operator also lets us test a tempting guard: that comparing *within* a model protects against operator misspecification, since the same forward model appears in both regimes. That is a prediction, and it fails, which is why the final protocol scores every model under one operator. Re-scoring the identical baseline samples under the corrected SIE moves the scale-free ratio from 2.17 to 6.04 for the conditional DDPM and from 2.43 to 5.57 for FlowLensing (Table 3).

Table 3: The same baseline images scored under two forward models. The ratio is not operator-invariant.

Model	Operator	In-dist	OOD	Scale-free ratio
Conditional DDPM	analytic (ours)	0.00166	0.00563	2.17
Conditional DDPM	corrected SIE	0.00029	0.00295	6.04
FlowLensing	analytic (ours)	0.00199	0.00672	2.43
FlowLensing	corrected SIE	0.00034	0.00277	5.57

The mechanism is visible in the columns. In distribution a baseline image *is* approximately a lensing of something, so a better operator fits it far better: for the DDPM the residual falls by $5.7\times$ ($5.8\times$ for FlowLensing). Out of distribution the image is a lensing of nothing, so the residual is set by the generator's error rather than the operator's and falls only $1.9\times$ ($2.4\times$). A poor operator therefore masks the in-distribution gap and compresses the ratio. Our reported figures are the compressed ones and hence the conservative choice, but the honest statement is that the within-model ratio reduces the operator's influence without removing it. This is the paper's own thesis applied to the paper: a physics-based diagnostic inherits the misspecification of its forward model, and so does the guard we built against it.

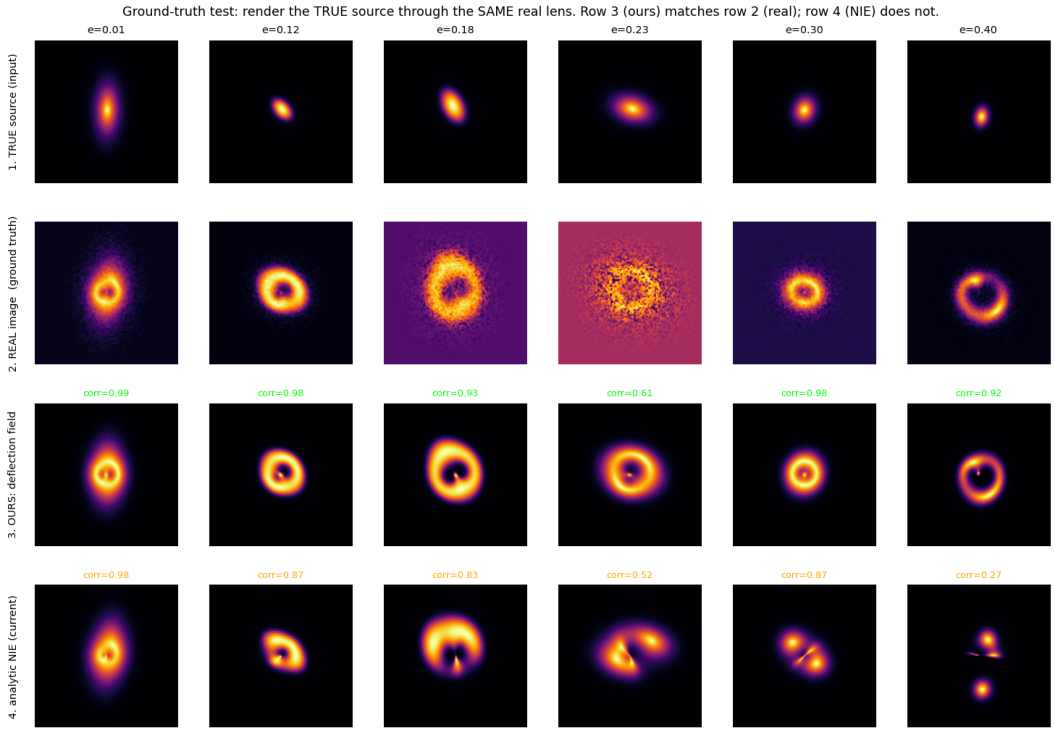


Figure 6: Row 1: true source. Row 2: the simulator's image for that system (ground truth). Row 3: our exact deflection-field render of the true source through the same lens, with correlation to row 2. Row 4: analytic renderer.

C TRAINING ON THE RESIDUAL

Section 5 shows that our validity residual is itself a forward model and that a misspecified one changes its verdict. A natural next question is what happens when that residual is used not to *judge* an output but to *train* one. We test it on a supervised source-inversion task (a U-Net recovering the unlensed source from the image, trained against the true source; development study at $5{,}400$ systems) by adding, to the brightness-weighted MSE, either a differentiable SSIM term or a re-lensing term $\|\mathcal{R}(\hat{s};\ell) - I\|^2$ that asks the recovered source to reproduce the input through the same lens. Four arms, identical in architecture, data, epochs, seed and batch order; only the loss changes (Table 4). The SSIM term is weighted by the same brightness map as the MSE, since a full-frame SSIM loss on sources that are about 95% black would mostly optimise empty sky.

Table 4: Physics-informed loss terms on supervised inversion. Centroid and flux errors are the honest columns: no added term optimises them.

Loss	Region SSIM	Centroid (px)	Flux error	Re-lensing MSE
MSE (baseline)	0.822	2.50	0.224	0.047
+ SSIM	**0.876**	**0.78**	0.148	0.049
+ re-lensing	0.679	2.44	4.43	**0.019**
+ SSIM + re-lensing	0.881	0.79	0.138	0.048

The SSIM term helps, and it helps on columns it does not optimise: the centroid error falls by a factor 3.2 and the flux error by 1.5. The re-lensing term does the opposite, and instructively so. It achieves the best re-lensing MSE by a factor 2.5, since it optimises exactly what it measures, while returning a *worse* source than the baseline, at region SSIM 0.679 and a flux error of 4.43, meaning a recovered source carrying over four times too much light. Our forward operator is not exact ($\chi^2/\nu = 1.57$ at best), and the term buys agreement with it by inflating the source to absorb the mismatch. Added on top of SSIM it changes nothing beyond the noise floor: 0.881 against 0.876, where re-running the baseline arm alone moves region SSIM by 0.014.

The lesson is the one Section 5 draws about the diagnostic, now in a second and independent setting: a physics-informed objective inherits the misspecification of the forward model it encodes, and optimising it can improve agreement with the physics *model* while degrading fidelity to the physics. This holds with the lens known and in compressed brightness; a single seed per arm and an untuned unit weight on both terms are the main caveats.

D ADDITIONAL BASELINES: IMPLEMENTATION

Ray-map DDPM. The conditional DDPM of Table 1 with a single change: the input convolution takes three channels instead of one, the noisy image and the two components of the source-plane position $\boldsymbol{\theta} - s_\alpha \nabla\psi(\boldsymbol{\theta}) - \gamma(\theta_1, -\theta_2)$ of every pixel, computed by the *same* finite-difference code as the scoring operator and normalised by the half-width of the field. It keeps the seven-parameter embedding. Training uses the true potential of each system at the nominal deflection scale and zero added shear; at test time it receives the ray map of the potential it is scored under, at the nominal lens in distribution and at the perturbed lens ($s_\alpha \times 1.6$, $\gamma = 0.28$) out of distribution. Architecture, seed (1), optimiser, batch size, EMA and the 30-epoch budget ($14{,}070$ updates) are those of the conditional DDPM; sampling uses the same seed (7) and 100 DDIM steps.

Soft constraint, raw and scale-free. The penalty is $\lambda V_\ell(\hat{x}_0)$ with $\hat{x}_0 = \sqrt{\bar{\alpha}_t}\, x_t - \sqrt{1-\bar{\alpha}_t}\, \hat{v}$ clipped to $[-1, 1]$ and mapped to $[0, 1]$. The inner problem runs 12 FISTA steps under `no_grad` with a fixed step $1/L$, L measured once by 12 power iterations on 256 training potentials ($\times 3$ safety, $\times 1.5$ margin); the penalty is then evaluated at the inner solution with gradients flowing to $\hat{x}_0$ only, which is exact at the inner optimum by the envelope argument of Section 3. The scale-free variant divides each image's residual by its mean squared intensity (floored at 10^{-4}, about $700\times$ below the real images' mean) before averaging over the batch. All arms share the conditional DDPM's architecture, seed and budget.

Second training seed. The three arms the conclusions rest on were retrained with training seed 2 (sampling seed unchanged) and scored under the same protocol (Table 5). No conclusion changes; the largest shift in a validity ratio is the scale-free arm's, from 2.86 to 2.54.

Table 5: Seed replicate of the decisive arms, $n = 512$, same protocol as Table 2. Each cell gives seed 1 / seed 2.

Arm	raw ratio OOD/in	scale-free OOD	arc en. OOD / in	shear imprint Δe_1
soft, $\lambda = 30$	1.00 / 1.00	0.223 / 0.214	0.26 / 0.27	+0.086 / +0.101
soft, scale-free, $\lambda = 1$	2.86 / 2.54	0.259 / 0.253	0.65 / 0.59	+0.023 / +0.021
ray-map DDPM	12.5 / 12.7	0.135 / 0.133	0.97 / 1.01	+0.586 / +0.578

Numerical check of Proposition 1. For each image we compute the projection $P = \mathcal{R}_\ell s^\star$ with the exact fitter (400 iterations) and report $\rho = \langle x - P, x\rangle / \|x - P\|^2$ (Proposition 1 states $\rho \geq 1$), the normalised cross term $\langle x - P, P\rangle / \|x\|^2$ (states ≥ 0), and $V_\ell(x/2)/V_\ell(x)$ (states $\leq 1/4$). On 256 real test images under their own lens and $256 + 256$ DDPM samples (in distribution, and drawn out of distribution but scored at the training lens): medians $\rho = 1.000, 1.000, 1.008$; cross term $\geq -10^{-4}$; $V_\ell(x/2)/V_\ell(x) = 0.250, 0.250, 0.250$. The bound holds with equality to the fitter's tolerance.

E A SECOND DOMAIN: DIVERGENCE-FREE FIELDS

Fields $v = (u, w)$ live on a periodic 32^2 grid. Training data are $v = \nabla^\perp \psi$ for Gaussian stream functions whose spectrum is an annulus of unit width around a wavenumber k_0, drawn uniformly in $[2, 5]$ for 20,000 training fields; the out-of-distribution request is $k_0 = 8$. The valid set is the subspace $\{\nabla \cdot v = 0\}$ and its exact projection is the Leray projector, applied by FFT with the Nyquist mode removed from the derivative operators so that curl and projection are exact on generated fields; the residual is $V(v) = \|v - Pv\|^2$. Every arm is the same v-prediction U-Net (64 base channels, conditioned on t and k_0), with the same seed and 6,000 updates at batch 128; the hard arm generates ψ (one channel) and returns its curl. Scores use 1,024 samples per regime and data drawn at the same k_0. On the unconstrained model's samples, $\langle \nabla V, v\rangle / 2V = 1.0000$ and $V(v/2)/V(v) = 0.2500$ (medians): the Proposition holds with equality (Table 6).

Table 6: Divergence-free fields. Energy and mean wavenumber relative to data drawn at the same requested k_0. The soft penalty lowers the OOD residual by emptying the field and losing the requested scale; the reparameterised model is exact but does not extrapolate the latent.

	residual V			energy / data		wavenumber / data
Model	in-dist	OOD	scale-free OOD	in	OOD	OOD
unconstrained	4.5×10^{-6}	0.143	0.364	1.04	0.69	1.25
soft, $\lambda = 1$	4.5×10^{-6}	0.108	0.322	1.04	0.59	1.18
soft, $\lambda = 10$	5.3×10^{-6}	0.0121	0.092	1.03	0.23	0.67
soft, $\lambda = 100$	9.1×10^{-6}	0.00075	0.009	0.99	0.15	0.56
soft, scale-free, $\lambda = 1$	5.6×10^{-6}	0.0415	0.206	1.14	0.35	0.94
unconstrained + projection	0	0	0	1.04	0.44	1.10
hard (stream function)	0	0	0	0.96	14.7	1.73

F OTHER OUT-OF-DISTRIBUTION INTERVENTIONS

Each intervention perturbs the requested lens of the same 512 systems; every model is sampled at the perturbed request (ours: the same generated source rendered at the new lens; baselines: perturbed conditioning; ray-map DDPM: the perturbed ray map), scored with the same operator and exact fitter, and compared with in-distribution samples of the same model. Training ranges are $\theta_E \in [0.80'', 3.26'']$ and $\gamma \in [-0.15, 0.15]$. The first row reproduces the paper's intervention with fresh sampling noise (Tables 7 and 8).

Table 7: Validity under five interventions: raw OOD/in-distribution ratio (scale-free OOD residual). The anchor is a real image scored under its own perturbed lens.

Intervention	beyond θ_E range	anchor	DDPM	FlowLensing	ray-map DDPM	soft $\lambda = 30$	soft, scale-free $\lambda = 1$
$\theta_E \times 1.6, \gamma + 0.28$ (paper)	4%	12.8 (0.210)	12.2 (0.124)	11.6 (0.115)	13.2 (0.131)	1.0 (0.221)	2.9 (0.258)
$\gamma + 0.28$	0%	9.7 (0.130)	11.4 (0.107)	10.1 (0.114)	12.4 (0.108)	2.3 (0.225)	3.8 (0.263)
$\gamma - 0.28$	0%	9.7 (0.125)	11.9 (0.107)	12.7 (0.123)	13.6 (0.110)	1.9 (0.218)	3.6 (0.262)
$\theta_E \times 1.6$	4%	5.7 (0.167)	2.8 (0.046)	3.8 (0.069)	3.0 (0.047)	1.0 (0.106)	1.1 (0.086)
$\theta_E \times 2.5$	54%	9.5 (0.239)	6.4 (0.085)	12.7 (0.144)	6.9 (0.086)	0.4 (0.144)	0.4 (0.072)

Table 8: Obedience under the interventions of Table 7. **Top:** the response each request should produce, measured by the statistic it changes (the ellipticity imprint Δe_1 for a shear, the ring-radius ratio for an Einstein-radius change), next to the physical target obtained by rendering the true sources through the true perturbed lens. `PhysLensDiff` (bold) matches the target in every row. **Bottom:** arc energy kept out of distribution (1: none lost); the soft penalty empties its images.

Intervention	measured response	target	`PhysLensDiff`	DDPM	FlowLensing	ray-map DDPM	soft $\lambda = 30$	scale-free $\lambda = 1$
Response to the request (closer to the target is better)								
$\theta_E \times 1.6, \gamma + 0.28$ (paper)	shear imprint Δe_1	$+0.438$	$+0.437$	$+0.425$	$+0.497$	$+0.575$	$+0.084$	$+0.023$
$\gamma + 0.28$	shear imprint Δe_1	$+0.472$	$+0.471$	$+0.375$	$+0.464$	$+0.525$	$+0.060$	$+0.017$
$\gamma - 0.28$	shear imprint Δe_1	-0.471	-0.466	-0.393	-0.501	-0.537	-0.067	$+0.003$
$\theta_E \times 1.6$	ring radius (OOD/in)	$\times 1.32$	$\times 1.32$	$\times 1.32$	$\times 1.42$	$\times 1.35$	$\times 1.41$	$\times 1.40$
$\theta_E \times 2.5$	ring radius (OOD/in)	$\times 1.72$	$\times 1.73$	$\times 1.82$	$\times 1.99$	$\times 1.86$	$\times 2.01$	$\times 1.90$
Arc energy kept out of distribution (OOD / in-distribution)								
$\theta_E \times 1.6, \gamma + 0.28$ (paper)		–	1.02	1.11	1.66	0.98	0.27	0.67
$\gamma + 0.28$		–	0.92	0.94	0.92	0.86	0.63	0.95
$\gamma - 0.28$		–	0.92	0.89	0.87	0.86	0.54	0.91
$\theta_E \times 1.6$		–	1.25	1.38	2.04	1.30	0.66	0.81
$\theta_E \times 2.5$		–	1.38	1.52	3.92	1.38	0.16	0.18

G OBSERVATION LAYER AND REALISM

Our images are renders of the model: they carry no photon noise, no sky pedestal and no instrument effects, so a CNN trained to separate them from real images succeeds almost perfectly. We measure realism with a two-sample classifier test (Lopez-Paz & Oquab, 2017): a CNN is trained on half of $1{,}024$ generated and $1{,}024$ real test images and scored on the other half (0.50 = indistinguishable). The critic here is stricter than a default one (30 epochs, five seeds); with the weaker critic (15 epochs, three seeds) a hand-tuned observation layer already appeared indistinguishable (0.53), which the stricter one refutes. Views: the full image, a low-pass version (Gaussian blur, $\sigma = 1.5$ px: morphology) and the high-pass remainder (noise and fine detail).

Hand-tuned layers (a pedestal and an intensity-dependent noise law measured on training images, optionally a Gaussian PSF and a render at 128^2 before downsampling) plateau at 0.60–0.63 (low-pass 0.54–0.60, high-pass 0.70–0.79): the morphology is close, the texture is not. The learned layer is the same v-prediction U-Net as the image DDPM, with the clean render as a second input channel, trained on the $30{,}000$ training pairs (clean render of the true source through the true lens, observed image) for the same update budget. It is trained to add only the instrument: the model image beneath it is exactly valid, and the observed images are as consistent with their lens as real ones (Table 9).

Table 9: Realism: strict two-sample critic accuracy (0.50 = indistinguishable), two sampling draws, and validity of the observed images (scale-free residual under their own lens).

Images	full image draw 1	full image draw 2	high-pass draw 1	low-pass draw 1	scale-free residual
real vs. real (floor)	0.50		0.50	0.50	0.032
ours, noise-free render	0.99	0.99	–	–	$< 10^{-8}$
ours, hand-tuned observation layer	0.60	0.60	0.79	0.54	–
ours, learned observation layer	**0.56**	**0.57**	0.62	0.55	0.027
image DDPM	0.54	0.56	0.50	0.54	–
FlowLensing	0.55	–	–	–	–

H WHAT THE HARD CONSTRAINT COSTS

All three generators use 100 sampling steps at 64^2 on one RTX 5070 laptop GPU, one model resident at a time, median of three timed runs after a warm-up. Times are milliseconds per image (Table 10).

Table 10: Generation cost. The render is the hard constraint itself: one bilinear resample of the source onto the deflected grid.

Batch	Ours (source + render)	*render alone*	Image DDPM	FlowLensing DiT
1	228.6	0.238	233.4	810.7
8	52.3	0.029	51.4	657.2
32	61.7	0.007	61.8	721.2

The constraint is free: the render is 0.1% of generation at batch 1 and 0.01% at batch 32, so validity by construction costs no measurable time. And it is free *relative to the alternative*: our pipeline and the image-space DDPM agree to within about 2% at every batch size, as they should, since they differ only in what the same-sized network is asked to produce.

What these numbers do *not* show is that generating the source is faster than generating the image. The DiT is 3.5–$12.6\times$ slower (Table 10), but it also carries 30.2M parameters against 3.9M for either DDPM, at identical step counts; that gap is a capacity choice, not a consequence of synthesising the image directly. Reading it as a win for the hard constraint would repeat the confound we retract in Section 5. The table times the source model and the render; the complete pipeline also samples the substructure density with a 128^2 model and costs 315 ms per image at batch 32 ($5\times$ the image DDPM), a cost of the substructure model, not of the constraint. Validity does not depend on the number of sampling steps: from 1 to 100 steps the scale-free residual of our OOD samples stays below 3×10^{-8}, and at 10 steps (a complete image in 32 ms) the shear imprint is $+0.431$ against $+0.437$ for the target in the same run. For scale, a `lenstronomy` and `pyHalo` simulation of a comparable system (CDM subhalo population, SIE plus shear, Sérsic source, Poisson noise, 128^2 pixels; not the exact Model A pipeline) takes 17.0 s on one CPU core (median of 20); the hardware differs, so we give this as an order of magnitude only.

I SUBSTRUCTURE GENERATION

The generated substructure has the wrong spectral shape. This appendix documents the development study's potential-space generator; the final density route and its slopes are in Section 5. Reading a class label off a residual is a weak test of physics: what distinguishes axion, CDM and warm dark matter is the subhalo mass function, and hence the small-scale power of the perturbation. Fitting a log–log slope to the radial power spectrum of the substructure residual over $k = 3$–25 gives -2.27, -1.92 and -1.83 for real axion, CDM and WDM systems, against -3.99, -3.53 and -3.78 for ours. Our generated fields are much steeper, meaning far too little power at small scales: a 128^2 diffusion model produces a smooth field where the simulator produces a granular one. The integrated power in that band reaches only 0.85, 0.58 and 0.55 of the real value.

What does survive is the ordering: at 18 of 22 wavenumbers the three classes rank in the same order as in the data. That is why a classifier still transfers, and it also bounds what the transfer means. The dark-matter signal we preserve is carried by relative amplitude, not by the small-scale statistics a physicist would use to tell the models apart. Reporting the classifier number without this would overstate the result.

The generator is a class-conditional v-prediction diffusion model on $\psi_{\rm sub}$ at 128^2, the resolution at which the substructure texture survives. The field is concentrated near zero with long tails (standard deviation 31, extremes ± 200), so a linear map into $[0, 1]$ would leave a standard deviation of 0.12 where the sampler expects roughly 0.25; we therefore compress with $\operatorname{arcsinh}(\psi/10)$ before a *global* min–max, monotone hence exactly invertible. Generated amplitudes recover 75–81% of the real per-class values and preserve their ordering (axion $>$ CDM $>$ WDM), which is where part of the class signal lives (Figures 7 and 8).

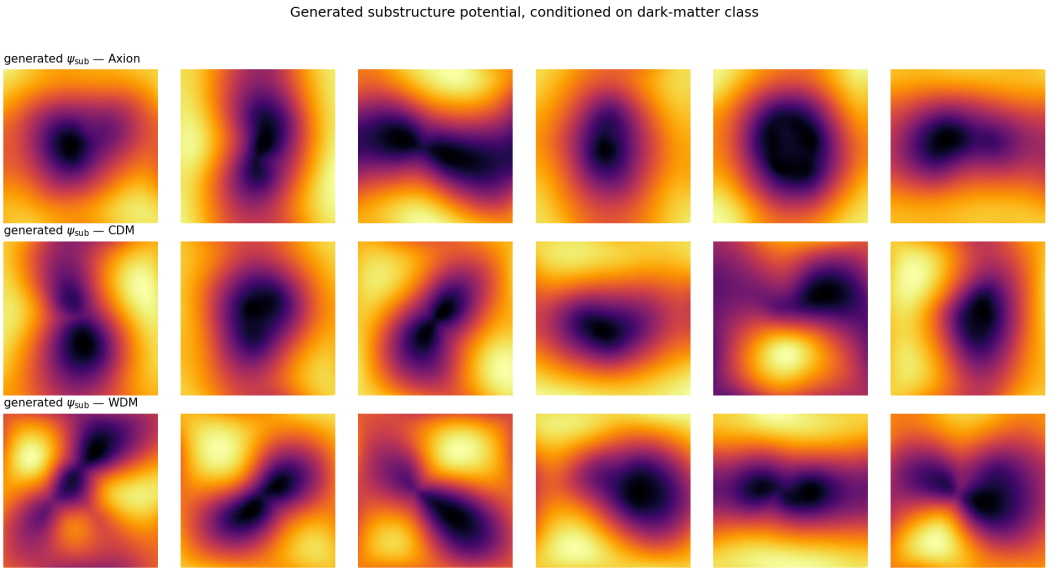


Figure 7: Generated substructure potential ψ_{sub}, six samples per dark-matter class.

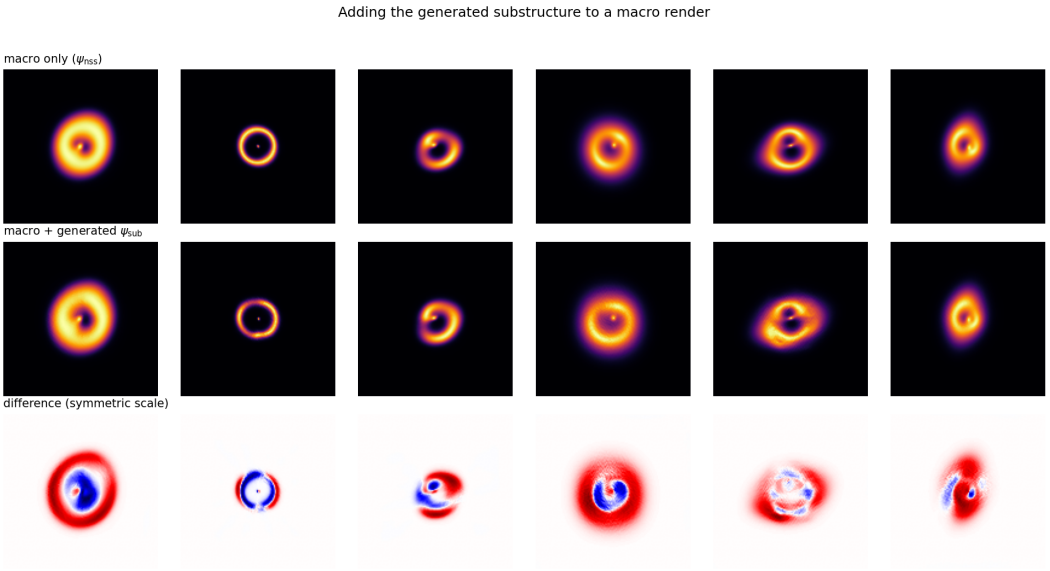


Figure 8: Adding the generated substructure. Top: macro render through ψ_{nss} alone. Middle: render through $\psi_{\mathrm{nss}} + \psi_{\mathrm{sub}}$. Bottom: the difference, which is the generated substructure residual the classifier reads. The output of the middle row is still a lensing of a source by construction, since only the deflection field changed.