
Computationally Efficient Information-Driven Optical Design with Interchanging Optimization

Anonymous Author(s)

Affiliation

Address

email

Abstract

Recent work has demonstrated that imaging systems can be evaluated through the information content of their measurements alone, enabling application-agnostic optical design that avoids computational decoding challenges. Information-Driven Encoder Analysis Learning (IDEAL) was proposed to automate this process through gradient-based optimization. In this work, we study IDEAL across diverse imaging systems and find that it suffers from high memory usage, long runtimes, and a potentially mismatched objective function due to end-to-end differentiability requirements. We introduce IDEAL with Interchanging Optimization (IDEAL-IO), a method that decouples density estimation from optical parameter optimization by alternating between fitting models to current measurements and updating optical parameters using fixed models for information estimation. This approach reduces runtime and memory usage by up to 6× while enabling more expressive density models that guide optimization toward superior designs. We validate our method on diffractive optics, lensless imaging, and snapshot 3D microscopy applications, establishing information-theoretic optimization as a practical, scalable strategy for real-world imaging system design.

1 Introduction

In computational imaging, optics and algorithms are designed jointly, unlocking novel trade-offs between hardware complexity, system performance, and cost [1]. A typical system contains an optical encoder that captures a measurement and a software decoder that reconstructs or interprets the scene. With the freedom to implement diverse optical encoding strategies made possible by computational decoding, the optimal encoder design is an open question in many applications.

Traditional approaches to encoder design rely on expert intuition and idealized physical models, often producing suboptimal designs that break down under real-world noise or model mismatch. More recently, end-to-end methods jointly optimize optical parameters and reconstruction algorithms using deep learning [2, 3, 4]. These systems treat the image formation model as a differentiable layer in a neural network and train the optics and decoder simultaneously. While this often yields high-performance designs, it requires substantial computational resources [5] and presents practical challenges when optimizing through state-of-the-art reconstruction algorithms [6].

Recent work has suggested that encoders could be designed independently using information-theoretic principles [7]. This approach is appealing because it decouples the encoder optimization from specific decoder implementations, focusing instead on measurement quality by maximizing the information content of measurements. Furthermore, information-theoretic metrics naturally balance resolution, signal-to-noise ratio, and other performance factors within a unified framework, providing an application-agnostic approach to optical design.

Information-Driven Encoder Analysis Learning (IDEAL) proposes automating this process through gradient-based optimization of mutual information between scenes and measurements. This approach offers compelling potential advantages: it avoids the challenges of designing and optimizing through decoders such as complex reconstruction algorithms, and as a result can readily be applied to a wide variety of systems. However, the limitations of IDEAL’s proposed implementation and effectiveness across diverse imaging systems have not been thoroughly explored.

In this work, we identify a critical limitation of IDEAL in its original formulation: the requirement for end-to-end differentiability through both density estimation and optical parameter optimization introduces prohibitive computational demands and potentially misaligned optimization objectives.

To address these issues, we introduce IDEAL with Interchanging Optimization (IDEAL-IO). Our key insight is that decoupling density model fitting from optical parameter updates enables significantly more efficient optimization while maintaining or improving design quality. We alternate between: (1) fitting a density model to fixed measurements from the current optical system and (2) updating the optics using a differentiable MI estimator while holding the fitted model constant. This eliminates the need for the fitting process itself to be differentiable with respect to optical parameters, reducing memory usage and runtime by up to 6× while enabling the use of more expressive density models that produce better designs.

We demonstrate the effectiveness of IDEAL-IO across diffractive optics, lensless imaging, and snapshot 3D microscopy, showing both computational advantages and design improvements. Our results establish information-theoretic optimization as a practical, scalable strategy for next-generation imaging system design, offering a powerful alternative to traditional approaches without their associated computational burdens.

2 Background

Our goal is to optimize a learnable element θ of an optical system $f(\cdot; \theta)$. For a scene or object \mathbf{O} , the system deterministically gives noise-free data $\mathbf{X} = f(\mathbf{O}; \theta)$ corrupted by a stochastic noise model ϵ to yield noisy measurements $\mathbf{Y}(\mathbf{O}; \theta) = \epsilon(f(\mathbf{O}; \theta))$. We assume access to a dataset of objects $\mathcal{D}_O = \{o_i\}_{i=1, \dots, N}$ drawn from a distribution $p(\mathbf{O})$ for which we would like to optimize the system. In this section, we describe prior work on how to formulate this optimization problem.

End-to-End Learning. Previously developed end-to-end learning approaches [2, 3, 4] train a neural network $g(\cdot; \phi)$ to perform reconstruction on a dataset of measurements $\mathcal{D}_Y = \{y_i\}_{i=1, \dots, N}$ simulated from \mathcal{D}_O . These methods use gradient descent-based strategies to jointly optimize

$$\theta^*, \phi^* = \arg \min_{\theta, \phi} \sum_i \mathcal{L}(g(\epsilon(f(o_i; \theta)); \phi), o_i), \quad (1)$$

where \mathcal{L} is a loss describing the fidelity of the reconstruction to the original object. While effective, these methods require substantial computational overhead to differentiate Eq. 1 through both reconstruction network g and forward model f [8]. Moreover, end-to-end optimization can get stuck in local minima and struggle to obtain accurate gradients when reconstructions have not yet converged [6]. They also yield a design specific to a particular reconstruction algorithm, and this design may become non-ideal if the algorithm is updated or improved.

IDEAL. An alternative strategy, Information-Driven Encoder Analysis Learning (IDEAL) was proposed [7] to avoid differentiation through and sensitivity to the reconstruction algorithm. IDEAL instead maximizes the mutual information $I(\mathbf{O}; \mathbf{Y})$ between the object \mathbf{O} and noisy measurement \mathbf{Y} . Because our optical system is deterministic, i.e. $\mathbf{X} = f(\mathbf{O}; \theta)$, this is equivalent to maximizing $I(\mathbf{X}; \mathbf{Y})$, which can be decomposed as

$$I(\mathbf{X}; \mathbf{Y}) = H(\mathbf{Y}) - H(\mathbf{Y}|\mathbf{X}). \quad (2)$$

$H(\mathbf{Y})$ is the entropy of the noisy sensor measurements and captures variability due to both the scene and sensor noise. $H(\mathbf{Y}|\mathbf{X})$ quantifies the uncertainty in \mathbf{Y} due solely to noise, conditioned on the underlying noiseless measurement. The difference of these terms isolates the portion of measurement variability due to the scene, and forms the core objective for information-driven optical design.

82 The conditional entropy, $H(\mathbf{Y}|\mathbf{X})$ can be computed analytically for Poisson and Gaussian noise;
 83 these analytical expressions are derived in Supplement Sec. 1. Accurately computing $H(\mathbf{Y})$, on the
 84 other hand, requires access to the true distribution $p(\mathbf{Y})$. In practice, this distribution is approximated
 85 by a parameterized model $p_\psi(\mathbf{Y})$, trained on \mathcal{D}_Y . This allows us to estimate an upper bound on the
 86 true entropy using the cross-entropy over a held-out test set of M noisy measurements:

$$H(\mathbf{Y}) \leq \hat{H}(\mathbf{Y}) = \mathbb{E}_{\mathbf{Y}}[-\log p_\psi(\mathbf{Y})] \quad (3)$$

$$\approx -\frac{1}{M} \sum_{i=1}^M \log p_\psi(\mathbf{y}^{(i)}). \quad (4)$$

87 The tightness of the bound in Eq. 3 depends on the expressiveness of the model class p_ψ . More
 88 expressive models—such as PixelCNN [9] or autoregressive transformers [10]—enable more accurate
 89 entropy estimates and, in turn, more informative optimization signals. This approach also allows
 90 systematic comparison across density models: lower cross-entropy values indicate better fit to
 91 the measurement distribution. Since this cross-entropy upper bounds the true entropy, it offers a
 92 systematic approach for model comparison: models yielding lower cross-entropy values provide
 93 more precise distribution estimates. This is used to select the best model for the given scene dataset
 94 and optical forward model.

95 The dependence of Eq. 3 and thus Eq. 2 on the *fitted* probability distribution p_ψ introduces significant
 96 complexity in the optimization. In the IDEAL setting, we aim to solve $\theta^* = \arg \min_{\theta} -I(\mathbf{X}; \mathbf{Y})$.
 97 The appropriate gradient descent update at step t is given by:

$$\theta_{t+1} = \theta_t - \alpha \left. \frac{\partial I(\mathbf{X}; \mathbf{Y})}{\partial \theta} \right|_{\theta=\theta_t} \quad (5)$$

$$\approx \theta_t - \alpha \left\{ -\frac{1}{M} \sum_{i=1}^M \frac{1}{p_{\psi(\theta_t)}(y^i)} \left[\frac{\partial p}{\partial \psi} \frac{\partial \psi}{\partial \theta} + \frac{\partial p}{\partial y} \frac{\partial y}{\partial \theta} \right] - \frac{\partial H(\mathbf{Y}|\mathbf{X})}{\partial \theta} \right\} \Big|_{\theta=\theta_t} \quad (6)$$

98 where α is the step size and a detailed derivation is given in Supplement Sec. 2. Of particular note,
 99 this update includes the quantity $\frac{\partial p}{\partial \psi} \frac{\partial \psi}{\partial \theta}$; the mutual information estimate depends on θ not only
 100 through the simulated measurements y but also through the fitted probability distribution parameters
 101 ψ . In previous work, IDEAL proposed computing gradients in a single step, using a multivariate
 102 Gaussian as p_ψ , estimating its covariance matrix from a batch of measurements, and using the analytic
 103 expression for its entropy. The downside of this approach is that it requires differentiating *through* this
 104 fitting procedure, which is computationally costly for large covariance matrices on high-dimensional
 105 images, and is practically impossible for more expressive parameterizations of p , such as neural
 106 networks, where the fitting process is typically an iterative procedure.

107 3 Methods

108 Our contribution is built on an empirical observation that the quantity $\frac{\partial \psi}{\partial \theta}$ is small; i.e., the rate of
 109 change of the fitted probability distribution parameters with respect to the learned imaging system
 110 parameters is small. We can then ignore the dependence of the mutual information on θ through the
 111 fitted parameters ψ . This allows us to dramatically simplify the optimization procedure by ignoring
 112 the $\frac{\partial p}{\partial \psi} \frac{\partial \psi}{\partial \theta}$ term and simply refitting ψ outside of the differentiation path to maintain an accurate
 113 estimate of p_ψ . Our proposed optimization scheme is thus:

$$\psi_{t+1} = \arg \min_{\psi} \mathbb{E}_{\mathbf{Y}}[-\log p_\psi(\mathbf{Y}(\theta_t))] \quad (7)$$

$$\theta_{t+1} = \theta_t - \alpha \left\{ -\frac{1}{M} \sum_{i=1}^M \frac{1}{p_{\psi_{t+1}}(y^i)} \left[\frac{\partial p}{\partial y} \frac{\partial y}{\partial \theta} \right] - \frac{\partial H(\mathbf{Y}|\mathbf{X})}{\partial \theta} \right\} \Big|_{\theta=\theta_t}. \quad (8)$$

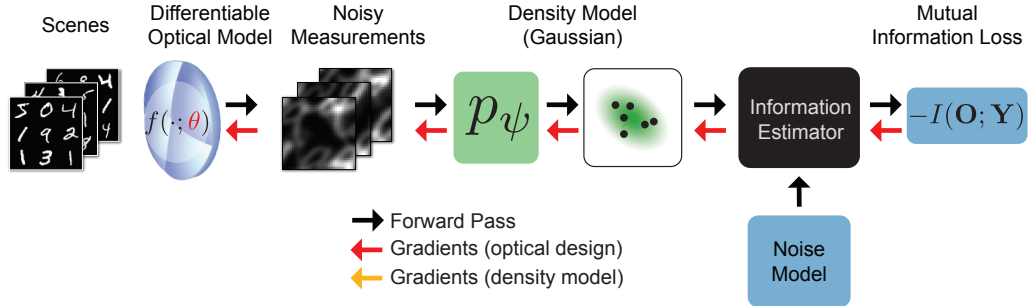
114 Intuitively, this scheme alternates between (1) fitting the probabilistic model to measurements
 115 generated with the current learned optical element and (2) updating the learned optical element using
 116 the most recent probabilistic model. In practice, we implement our method by freezing gradient

tracking during the fitting of the probability distribution, and then relying on auto-differentiation to compute the gradient in Eq. 7. To reduce memory usage, we operate on measurement patches rather than full-frame measurements during both model fitting and mutual information estimation. The full optimization pipeline is illustrated in Fig. 1.

This modification dramatically improves the usability of IDEAL. Because we no longer need to differentiate through the model fitting procedure, we are able to greatly reduce the memory and runtime cost of our algorithm. Further, we no longer have to restrict to parameterizations of p whose fitting process is differentiable, enabling us to use complex, expressive models previously intractable under IDEAL. In particular, we can use neural networks like PixelCNN [9] to model complex, high-dimensional probability distributions that are not Gaussian in nature.

Of note, while the Gaussian model can be fit almost instantaneously, the PixelCNN model requires a significantly longer time to fit (order of minutes). Consequently, when using the PixelCNN model, we opt not to re-fit after each imaging system update but rather every K optical parameter updates, as the loss only needs to provide useful gradient directionality—not an exact estimate of mutual information. For further discussion on PixelCNN re-fit frequency see Supplement Sec. 3.

a) Overview of IDEAL Optimization Pipeline



b) Overview of IDEAL-IO Optimization Pipeline

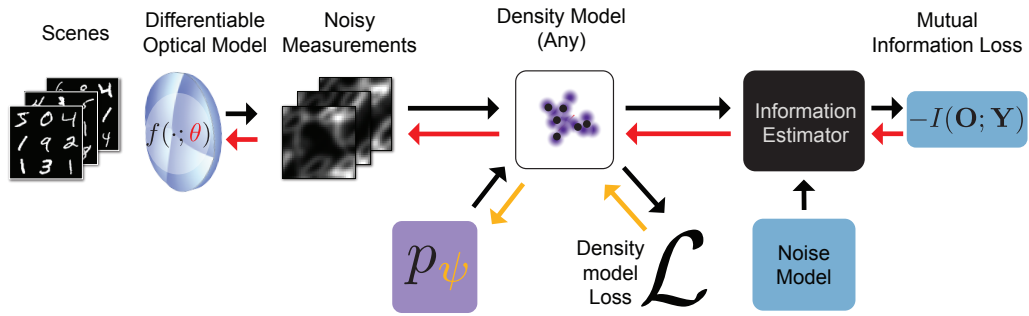


Figure 1: Comparison of Information-Driven Encoder Analysis Learning (IDEAL) and IDEAL with Interchanging Optimization (IDEAL-IO) optimization frameworks. (a) IDEAL jointly optimizes a differentiable density model and imaging system to maximize mutual information (MI) between noiseless and noisy measurements, $I(\mathbf{X}; \mathbf{Y})$. This coupling requires the entire pipeline to be differentiable with respect to the optical parameters, resulting in high memory and compute overhead and limited density model choices. (b) IDEAL-IO decouples model fitting and optical updates. A density model (e.g., PixelCNN) is fit to noisy measurements with gradients frozen. The fit density model and known noise distribution are then used to estimate MI, which is backpropagated through the imaging system. This alternation substantially lowers computational costs, enabling the use of more expressive models.

4 Results

We next describe a series of applications showing the advantages of the IDEAL-IO method. First, we show that IDEAL-IO performs comparably to end-to-end design for designing a learned lens

array for a snapshot 3D microscopy system while requiring substantially less runtime. Next, we show that the IDEAL-IO alternating update scheme enables faster, more memory-efficient optimization of a large diffractive optical element than standard IDEAL. Finally, we show that IDEAL-IO’s compatibility with more expressive probability density parameterizations enables design of a superior point spread function for lensless imaging. We provide an overview of each imaging system within the appropriate subsection, and formal analytical expressions for each system’s forward model are available in Supplement Section 9.

4.1 Snapshot 3D Microscopy: IDEAL-IO is Faster Than End-to-End Learning

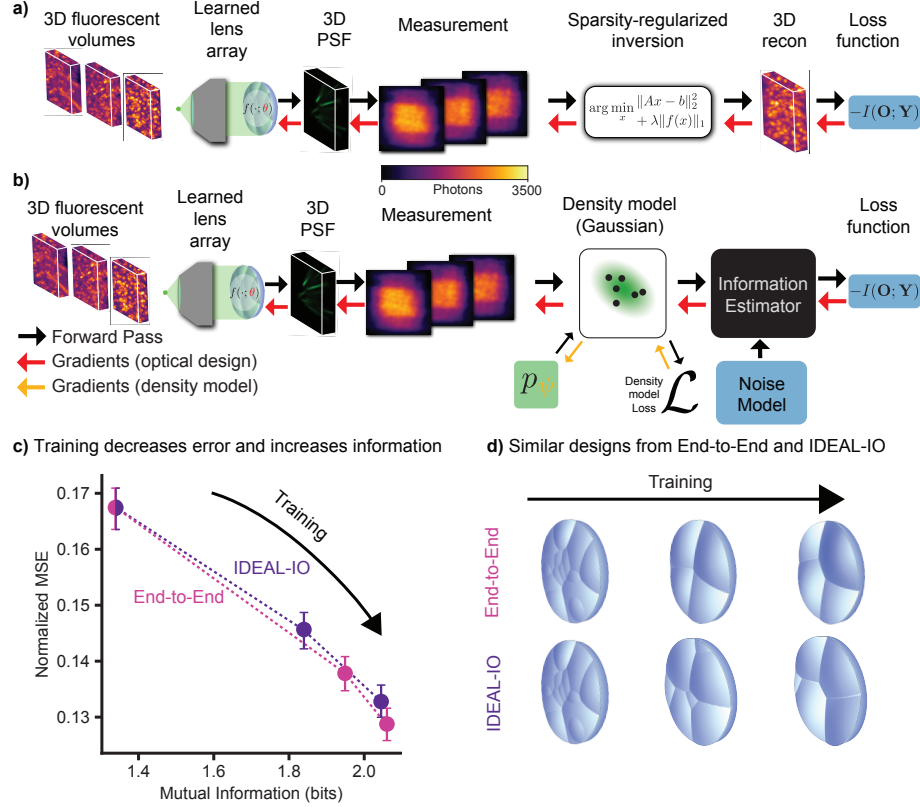


Figure 2: Comparison of IDEAL-IO and end-to-end (E2E) design for snapshot 3D microscopy. (a) E2E pipeline. A learned lenslet array encodes a 3D fluorescent volume into a 2D measurement. A neural decoder (FISTA-Net) reconstructs the volume which is fed into a loss function along with the ground truth volume. Gradients are backpropagated through the full pipeline to optimize the optical encoder. (b) IDEAL-IO pipeline. A density model is fit to simulated measurements (with gradient tracking off), then used to estimate mutual information (MI) (with gradient tracking turned on). The MI estimate is negated and serves as the loss function to drive encoder optimization. (c) MI vs. normalized mean squared error (NMSE) at three checkpoints: initialization, mid-training, and convergence for both IDEAL-IO and E2E. MI is estimated using a Gaussian model; NMSE is computed using a separately trained decoder. (d) Lenslet arrays learned by both methods at the checkpoints in (c). Both methods converge to similar four-lenslet designs that distribute focal points across the volume depth of interest.

First, we evaluate IDEAL-IO on a simulated snapshot 3D fluorescence microscopy system modeled after the Fourier DiffuserScope [11]. This architecture places a lenslet array in the Fourier plane of a conventional fluorescence microscopy setup to encode a 3D fluorescent volume into a 2D sensor measurement. Then, a neural decoder (FISTA-Net [12]) reconstructs the volume. In prior work, the entire system has been optimized end-to-end (E2E) by minimizing reconstruction error [13] through joint optimization of the lenslet array encoder and neural decoder. While effective, E2E optimization

is computationally intensive. We hypothesize that its success stems in part from implicitly increasing the MI between the scene and measurement. If true, directly maximizing MI should yield comparable encoder designs without the overhead of decoder training during optimization.

To test this, we train the E2E system on fluorescence microscopy images of mouse lungs and evaluate encoder quality at three checkpoints: initialization, midway through training, and convergence. At each point, we compute MI using a Gaussian density estimator and assess reconstruction fidelity via normalized mean squared error (NMSE) by training a reconstruction network with the optic fixed. As shown in Fig. 2c, MI increases monotonically as NMSE decreases, supporting the hypothesis that E2E optimization indirectly maximizes MI.

We then apply IDEAL-IO to the same design task. Despite never training a decoder during optimization, IDEAL-IO produces an encoder with comparable MI and NMSE to the E2E-trained system. The learned encoder exhibits a similar lenslet configuration (Fig. 2d), with focal points distributed across depth.

IDEAL-IO converges in ~ 25 minutes on a single RTX A6000 GPU—roughly $4\times$ faster than the ~ 2 hour E2E baseline. Unlike E2E, which required memory-saving techniques such as gradient checkpointing [8], IDEAL-IO ran without special accommodations. These results demonstrate that MI is not only a valid proxy for reconstruction fidelity, but also a practical and scalable objective for optical encoder design.

4.2 Diffractive Optical Elements: IDEAL-IO is Faster, More Memory-Efficient than IDEAL

To compare GPU memory requirements and runtime of our method, IDEAL-IO, to the original implementation of IDEAL, we apply it to a optical design task with a large number of learnable parameters: optimizing a pixelwise height map for a diffractive optical element (DOE) used in single-plane imaging [3]. This problem involves optimizing a 2496×2496 pixel height map resulting in ~ 6.2 million learnable optical parameters. We modify the code from [3] and use the dataset from [14] to perform this optimization.

This DOE is modeled as a flat surface with a spatially-varying phase profile, where each pixel in the height map imparts a phase delay to the incoming wavefront according to

$$\phi(x, y) = \frac{2\pi}{\lambda} \Delta n h(x, y),$$

where $h(x, y)$ is the pixel height, $\lambda = 650$ nm is the design wavelength, and $\Delta n = 1.4599 - 1.0$ is the refractive index contrast between the DOE material and air. The refractive index is assumed to be uniform across the surface. To simulate image formation, we use angular spectrum propagation, a standard wave optics technique that models wave propagation [15]. To generate sensor measurements, we convolve this point spread function with the input object and add noise based on the Gaussian approximation of Poisson noise.

Despite the large number of learnable parameters, IDEAL-IO converges in ~ 120 seconds on a single Nvidia RTX A6000 GPU using only 7.89 GB; in contrast, IDEAL converges in ~ 360 seconds using 44.23 GB. Both optimizations were carried out using patches of size 36×36 pixels and used 6480 patches to fit the Gaussian model.

For this test case, both IDEAL and IDEAL-IO converge to something close to the expected design – a Fresnel zone plate – despite no explicit structural priors being imposed. A Fresnel zone plate is the binary diffractive optical equivalent of a traditional lens, which should be optimal for 2D imaging of dense, natural scenes [16]. Figure 3a shows the training loss curves along with visualizations of the learned DOEs and corresponding measurements at selected iterations. Both methods show similar convergence, though IDEAL-IO’s curve appears noisier due to its smaller test set and mutual information’s sensitivity to outlier patches.

To quantify memory and runtime differences between methods, we ran optimizations using patch sizes ranging from 4×4 to 36×36 pixels. For each patch size, we performed 50 optimization steps using a training set of measurements equal to $5\times$ the number of pixels in the patch (e.g., 80 measurements for a 4×4 patch). This was repeated 10 times per patch size for each method. In Fig. 3b–c, we plot the mean peak GPU memory usage and the mean runtime per optimization step. IDEAL-IO provides a substantial decrease in both GPU memory usage and runtime compared to IDEAL.

Table 1: **Scaling of runtime and GPU memory with patch size in mutual information estimation.** We report the (1) number of additional patch pixels required to increase GPU memory usage by 1 GB, and (2) increase in runtime per optimization step for every additional 100 patch pixels.

Method	Patch pixels per +1 GB memory	Runtime increase per +100 pixels (s)
IDEAL	31 ± 0	0.098 ± 0.007
IDEAL-IO (Ours)	274 ± 4	0.007 ± 0.001

200 To better quantify memory scaling, we fit a linear regression model between the number of pixels in
 201 the patch and peak GPU memory usage. We report the number of additional patch pixels required
 202 to increase GPU usage by 1 GB, computed as the inverse slope of the fit. Similarly, we fit a linear
 203 model to runtime per step as a function of pixel count, and report the runtime increase per 100
 204 additional pixels. These results are presented in Table 1 and demonstrate that our method preserves
 205 the performance benefits of mutual information-based design while dramatically improving scalability
 206 and runtime.

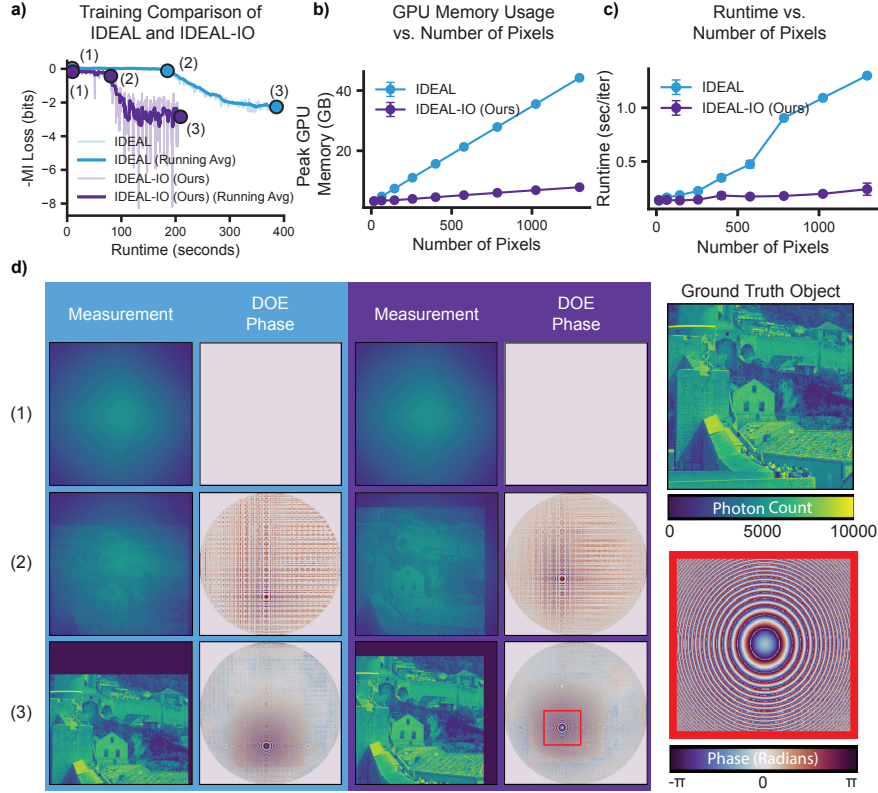


Figure 3: **Scaling behavior and learned designs for diffractive optical element (DOE) optimization with IDEAL and IDEAL-IO.** (a) Optimization of a DOE for single-plane imaging using IDEAL (blue) and IDEAL-IO (ours, purple). Training loss curves are shown for both methods. (b) Runtime per optimization step vs. number of pixels in a patch used for mutual information estimation. IDEAL runtime increases rapidly with patch size; IDEAL-IO remains relatively constant. Error bars denote standard deviation across the 10 repeated trials; when they are not visible, the error is smaller than the data marker. (c) Peak GPU memory usage vs. patch size. IDEAL memory usage grows steeply; IDEAL-IO scales more favorably. (d) Visualizations of the DOE phase profile and resulting measurement from (1) initialization, (2) an intermediate training step, and (3) convergence, for both methods. Despite no structural priors, both IDEAL and IDEAL-IO converge to Fresnel-like designs.

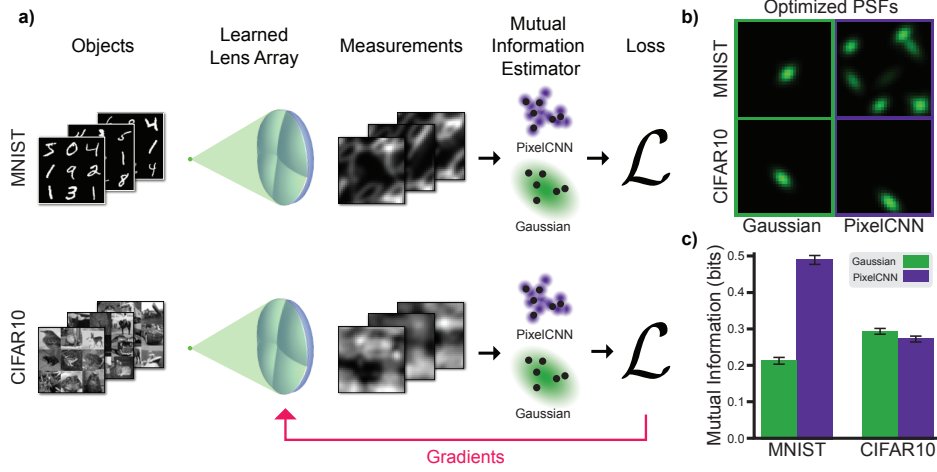


Figure 4: IDEAL-IO improves lensless imaging design by using expressive mutual information estimators. (a) IDEAL-IO is used to optimize the PSF of a lenslet array by maximizing mutual information (MI) between scenes and noisy measurements. We compare two density models used in the MI estimator: a Gaussian model and a more expressive PixelCNN. CIFAR10 measurements are approximately Gaussian, while MNIST measurements are sparse and non-Gaussian. (b) Optimized PSFs differ significantly for MNIST depending on the density model chosen for the MI estimator, reflecting the impact of model expressivity. For CIFAR10, both models yield similar PSFs. (c) PixelCNN-based designs achieve higher MI on MNIST test data, indicating better alignment with the true measurement distribution. For CIFAR10, both estimators perform similarly due to the near-Gaussian nature of the measurements.

To assess how density model expressivity affects optical design, we revisit a key trade-off identified in prior work [7]: simple models like Gaussians enable fast optimization but may fail when measurements exhibit non-Gaussian structure. We use IDEAL-IO’s compatibility with highly-expressive probabilistic models to investigate whether using a more expressive model during optimization leads to better designs, a comparison not possible with prior work.

We test this question with a simulated lensless imaging system in which a phase mask is placed a short distance in front of the sensor. With this configuration, lensless imagers achieve compact form factors and as a consequence have large distributed PSFs, in contrast to the single-spot PSF that is optimal for the Fresnel-lens in Sec. 4.2, where the phase mask to sensor distance was larger. Based on prior work in lensless imaging design, a phase mask comprised of a random array of lenslets trades off multiplexing and signal-to-noise ratio well [17]. Hence, we choose to model the phase mask as an array of lenslets, with its PSF modeled as a sum of 2D Gaussians, each representing the focal spot of an individual lenslet. The mean of each Gaussian determines the focal point location on the sensor, while the covariance controls its shape and orientation. Both the means and covariances are learned during optimization. See Supplement Sec. 7 for the initialized PSF.

To evaluate the role of density model expressivity, we optimize this system using mutual information estimated with a density model of a simple multivariate Gaussian or a more expressive autoregressive PixelCNN and a Poisson noise model. We test both approaches on CIFAR10 [18], where natural image statistics yield approximately Gaussian measurements, and MNIST [19], where sparse digit structures produce highly non-Gaussian, often bimodal, measurement distributions. We fit both density models using 16,024 patches of size of 16×16 pixels. The PixelCNN density model was refit after every 50 optical optimization steps in order to reduce runtime, while the Gaussian model was fit after every optical update due to the minimal computational cost of refitting. The learned estimators were evaluated on 1602 patches to calculate the mutual information loss.

For the CIFAR10 dataset, both estimators converge to similar optical designs with nearly identical loss curves. However, for MNIST, the two estimators yield distinct designs, revealing that model

expressivity significantly affects optimization when the true measurement distribution deviates from Gaussian assumptions (Fig. 4).

To evaluate design quality, we compute MI on held-out test data using a PixelCNN estimator. For CIFAR10, both designs perform similarly (0.283 vs. 0.264 bits/pixel), but for MNIST, the PixelCNN-optimized design significantly outperforms the Gaussian one (0.473 vs. 0.208 bits/pixel), confirming that expressive models yield better designs for non-Gaussian data. These results align with [16].

5 Discussion

In this work, we introduced and rigorously-tested a low computational cost method for imaging system design that builds on the IDEAL framework [7]. Like IDEAL, our approach optimizes optical parameters by maximizing the mutual information between noiseless and noisy sensor measurements. Our key contribution is a decoupled, two-stage optimization procedure that eliminates the need for the density model fitting to be differentiable with respect to the optical parameters. By separating density model fitting from differentiable mutual information estimation, our method supports more expressive models, reduces memory usage, and improves runtime, with maintained or improved performance. We demonstrated the effectiveness of the proposed method across three distinct computational imaging problems: diffractive optical imaging, lensless imaging, and snapshot 3D microscopy.

A key constraint in our framework is the reliance on patch-based entropy estimation, which creates a trade-off between model fidelity and memory efficiency. When using density models like multivariate Gaussians, the number of parameters increases quadratically with patch size. As the dimensionality grows, more samples are needed to fit the model reliably, which increases the computational burden. At the same time, larger patches reduce the number of unique samples that can be extracted from each measurement. This lowers the statistical diversity of the training set and forces the simulation of more measurements to maintain performance. The result is a significant increase in both memory use and runtime, and may limit modeling capacity in some cases. Future work could explore multiscale or hierarchical approaches to relax this limitation without adding excessive computational overhead.

Mutual information offers a strong foundation for reconstruction-based imaging, as preserving scene information in the system measurement theoretically enables perfect reconstruction. However, mutual information may not always align with specialized downstream tasks. In tasks where most information may be concentrated in a small subset of the scene, such as classification or detection, maximizing total information content might lead to designs that prioritize high-entropy regions that are irrelevant to the task. Extending our framework to support task-specific variants of mutual information, such as conditional or class-aware formulations, could improve performance in these settings.

Finally, the computational efficiency of our method enables large-scale exploration of design spaces. Multiple initializations can be evaluated quickly, revealing whether optimization consistently converges to similar designs or is sensitive to initialization. This flexibility also facilitates perturbation analyses, which can quantify sensitivity to noise, model error, or manufacturing tolerances—critical factors in real-world deployment.

By directly maximizing the information content of measurements, IDEAL-IO produces high-performance optical systems without the computational burden of jointly training reconstruction networks while designing the optics. This approach bridges the gap between theoretical guarantees and practical system design, providing both performance benefits and computational efficiency. Our framework serves as a scalable foundation for next-generation imaging system design. Future work could further broaden its applicability by incorporating task-specific constraints, multichannel mutual information estimators, and theoretical analyses of the connections between mutual information and downstream task performance.

References

- [1] Ayush Bhandari, Achuta Kadambi, and Ramesh Raskar. *Computational imaging*. The MIT Press, Cambridge, Massachusetts, 2022.
- [2] Zin Lin, Charles Roques-Carmes, Raphaël Pestourie, Marin Soljačić, Arka Majumdar, and Steven G. Johnson. End-to-End Nanophotonic Inverse Design for Imaging and Polarimetry. 2020. arXiv: 2006.09145.

- 284 [3] Vincent Sitzmann, Steven Diamond, Yifan Peng, Xiong Dun, Stephen Boyd, Wolfgang Heidrich, Felix
285 Heide, and Gordon Wetzstein. End-to-end optimization of optics and image processing for achromatic
286 extended depth of field and super-resolution imaging. *ACM Transactions on Graphics*, 37(4), 2018.
287 Publisher: Association for Computing Machinery.
- 288 [4] Qilin Sun, Congli Wang, Fu Qiang, Dun Xiong, and Heidrich Wolfgang. End-to-end complex lens design
289 with differentiable ray tracing. *ACM Transactions on Graphics (TOG)*, 40(4), 2021.
- 290 [5] Diptodip Deb, Zhenfei Jiao, Alex B. Chen, Misha B. Ahrens, Kaspar Podgorski, and Srinivas C. Turaga.
291 Programmable 3D snapshot microscopy with Fourier convolutional networks. i:1–27, 2021. arXiv:
292 2104.10611.
- 293 [6] Xinge Yang, Qiang Fu, Yunfeng Nie, and Wolfgang Heidrich. Image Quality Is Not All You Want:
294 Task-Driven Lens Design for Image Classification, 2023. arXiv:2305.17185.
- 295 [7] Anonymous. Information-driven design of imaging systems. Under review at NeurIPS 2025, 2025.
- 296 [8] Michael Kellman, Kevin Zhang, Eric Markley, Jon Tamir, Emrah Bostan, Michael Lustig, and Laura Waller.
297 Memory-Efficient Learning for Large-Scale Computational Imaging. *IEEE Transactions on Computational*
298 *Imaging*, 6:1403–1414, March 2020. arXiv: 2003.05551.
- 299 [9] Aaron van den Oord, Nal Kalchbrenner, Oriol Vinyals, Lasse Espeholt, Alex Graves, and Koray
300 Kavukcuoglu. Conditional image generation with PixelCNN decoders. 2016. arXiv:1606.05328.
- 301 [10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz
302 Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing*
303 *Systems*, volume 30. Curran Associates, Inc., 2017.
- 304 [11] Fanglin Linda Liu, Grace Kuo, Nick Antipa, Kyrollos Yanny, and Laura Waller. Fourier diffuserscope:
305 Single-shot 3d fourier light field microscopy with a diffuser. *Optics Express*, June 2020. arXiv: 2006.16343
306 Publisher: OSA.
- 307 [12] Jinxi Xiang, Yonggui Dong, and Yunjie Yang. FISTA-Net: Learning a Fast Iterative Shrinkage Thresholding
308 Network for Inverse Problems in Imaging. *IEEE Transactions on Medical Imaging*, 40(5):1329–1339,
309 2021.
- 310 [13] Eric Markley, Fanglin Linda Liu, Michael Kellman, Nick Antipa, and Laura Waller. Physics-based learned
311 design for fourier diffuserscope. In *OSA Imaging and Applied Optics Congress 2021 (3D, COSI, DH, ISA,*
312 *pcAOP)*, OSA Technical Digest, page CM6B.3. Optica Publishing Group, 2021.
- 313 [14] Herve Jegou, Matthijs Douze, and Cordelia Schmid. Hamming Embedding and Weak Geometric Consis-
314 tency for Large Scale Image Search. In *Proceedings of the 10th European Conference on Computer Vision:*
315 *Part I, ECCV '08*, pages 304–317. Springer-Verlag, 2008.
- 316 [15] Joseph W Goodman. *Introduction to Fourier optics*. Roberts and Company, 2005.
- 317 [16] Leyla Kabuli, Henry Pinkard, Eric Markley, Clara S Hung, and Laura Waller. Designing lensless imaging
318 systems to maximize information capture. *In review*, 2025.
- 319 [17] Leyla Kabuli, Gina Wu, and Laura Waller. High-quality lensless imaging with a random multi-focal lenslet
320 phase mask. In *Optica Imaging Congress (3D, COSI, DH, FFlatOptics, IS, pcAOP), Technical Digest*
321 *Series*, Optica Publishing Group, page paper CW3B.2, 2023.
- 322 [18] Alex Krizhevsky. Learning multiple layers of features from tiny images, 2009.
- 323 [19] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal*
324 *Processing Magazine*, 29(6):141–142, 2012.

NeurIPS Paper Checklist

1. Claims

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

Answer: [\[Yes\]](#)

Justification: The abstract and introduction clearly state the main claims: (1) IDEAL-IO reduces memory/runtime over IDEAL by up to 6×, (2) decouples density estimation and optical optimization, and (3) supports more expressive models. These claims are substantiated by experimental and theoretical results across three imaging tasks.

Guidelines:

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

2. Limitations

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

Answer: [\[Yes\]](#)

Justification: The discussion section acknowledges patch-based entropy estimation as a limiting factor, discusses trade-offs between model fidelity and efficiency, and notes that mutual information may misalign with task-specific goals. These limitations are presented with suggestions for future work.

Guidelines:

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

3. Theory assumptions and proofs

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

Answer: [NA]

Justification: We do not prove any theoretical results in this work.

Guidelines:

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

4. Experimental result reproducibility

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

Answer: [Yes]

Justification: The paper specifies GPU hardware, patch sizes, number of optimization steps, dataset details, noise models, and model training frequencies. These are sufficient to reproduce the reported results. Additionally all data and code is open source. The code is anonymized and included along with the submission.

Guidelines:

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

- (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

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

Answer: [Yes]

Justification: A zip file containing the full anonymized codebase is included, along with a README that maps each experiment and figure to the corresponding code and data for reproduction.

Guidelines:

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

6. Experimental setting/details

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

Answer: [Yes]

Justification: Yes, we supply these details in the methods, results, and supplement of our paper. We also have included the code to replicate all results and figures in the attached supplementary material.

Guidelines:

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

7. Experiment statistical significance

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

Answer: [Yes]

Justification: We provide error bars when applicable in all results and note if the error bars are less than the size of figure markers when applicable.

Guidelines:

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

8. Experiments compute resources

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

Answer: [Yes]

Justification: The paper reports GPU model (RTX A6000), memory usage (GB), and runtime per method. These are presented for all experiments and allow for replication.

Guidelines:

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

9. Code of ethics

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

Answer: [Yes]

Justification: We have reviewed NeurIPS Code of Ethics and do not see any reason this submission would run afoul of them. No ethical concerns are raised by the work. No human data, privacy risks, or malicious use cases are evident.

Guidelines:

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

10. Broader impacts

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

Answer: [NA]

Justification: This work is largely fundamental research and we do not see any clear or direct paths to significant societal impact.

Guidelines:

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

11. Safeguards

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

Answer: [NA]

Justification: We do not believe the paper releases any high-risk assets or models requiring safeguards.

Guidelines:

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

12. Licenses for existing assets

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

Answer: [Yes]

Justification: We cite our use of MNIST, CIFAR10, and Jegou et al.'s dataet. The mouse lung dataset was collected by one of the authors.

Guidelines:

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

13. New assets

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

Answer: [\[Yes\]](#)

Justification: Yes all code and datasets used in the paper is well documented and open sourced.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

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

Answer: [\[NA\]](#)

Justification: No human subjects or crowdsourcing were involved.

Guidelines:

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

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

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

Answer: [NA]

Justification: The work does not involve human participants.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

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

Answer: [NA]

Justification: The method presented in this work does not use LLMs as any important, original, or non-standard component.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.