Optical Diffusion Models for Image Generation

Ilker Oguz[1] **Niyazi Ulas Dinc**[1] **Mustafa Yildirim**[1] **Junjie Ke**[2] **Innfarn Yoo**[2]
Qifei Wang[3] **Feng Yang**[2*] **Christophe Moser**[1*] **Demetri Psaltis**[1*]
[1] École Polytechnique Fédérale de Lausanne [2] Google Research [3] Google
{ilker.oguz,niyazi.dinc,mustafa.yildirim,christophe.moser,demetri.psaltis}@epfl.ch
{junjiek,innfarn,qfwang,fengyang}@google.com

Abstract

Diffusion models generate new samples by progressively decreasing the noise from the initially provided random distribution. This inference procedure generally utilizes a trained neural network numerous times to obtain the final output, creating significant latency and energy consumption on digital electronic hardware such as GPUs. In this study, we demonstrate that the propagation of a light beam through a semi-transparent medium can be programmed to implement a denoising diffusion model on image samples. This framework projects noisy image patterns through passive diffractive optical layers, which collectively only transmit the predicted noise term in the image. The optical transparent layers, which are trained with an online training approach, backpropagating the error to the analytical model of the system, are passive and kept the same across different steps of denoising. Hence this method enables high-speed image generation with minimal power consumption, benefiting from the bandwidth and energy efficiency of optical information processing.

1 Introduction

Diffusion models create new samples that resemble their training sets by gradually undoing the diffusion process, which requires the learned reverse process to be applied numerous times [1]. While this method demonstrated unprecedented capabilities by producing highly realistic samples [2–8], it is also highly time-consuming and expensive in terms of energy consumption and computing resources since a large number of steps are required for generating each sample [2]. This prolonged processing time not only limits accessibility but also contributes to a significant environmental footprint.

Currently, generating new samples with diffusion models relies on electronic, general-purpose computing hardware such as GPUs or TPUs. However, due to the repetitive nature of the reversal process required in this task, deploying specialized hardware instead of general-purpose ones could significantly enhance the efficiency of sampling. For instance, the use of ASICs in cryptocurrency mining for hashing algorithms has demonstrated substantial improvements in computational speed and energy efficiency [9]. However, both GPUs and ASICs, among other electronic digital computers, face the same challenges like heat dissipation, energy consumption, and the diminishing returns of Moore's Law, as transistors shrink and encounter quantum effects and physical limits that hinder further gains [10]. Therefore, exploring alternative computing modalities, such as optical computing—which offers high bandwidth and low loss—is increasingly important [11]. Optical computing has already shown promise in various applications, including high-speed data transmission and real-time signal processing. Optical computing addresses the inefficiencies of electronic hardware by leveraging the inherent parallelism that light allows for the simultaneous processing of multiple data channels, significantly speeding up the computational process. Several optical neural networks have been

[*] Equal advising.

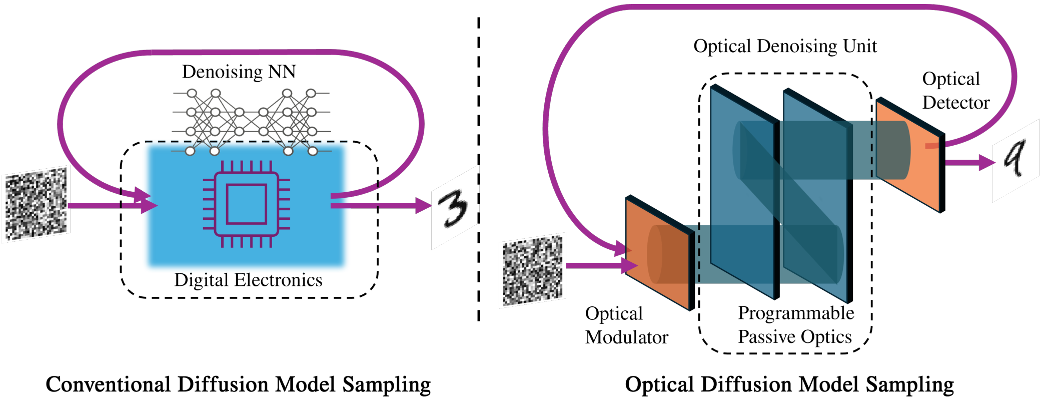
38th Conference on Neural Information Processing Systems (NeurIPS 2024).

https://doi.org/10.52202/079017-1887

reported to perform complex calculations at reduced latency and energy consumption compared to traditional electronic systems [12].

In this study, we demonstrate that optical wave propagation can be programmed to act as a computing engine specifically designed for implementing denoising diffusion models. As light passes through specially engineered transparent layers, features related to the original distribution are filtered out without any additional power consumption or computing latency, as depicted in Figure 1. This is due to the passive nature of the transparent layers, which are minimally absorptive and do not require active components or external power to function. These layers are designed to manipulate the light solely through their physical structure, which allows for the noise prediction to exit the system efficiently. Near zero energy consumption of passive optical components reduces the overall power requirements, making the system more energy-efficient.

Through iterative noise prediction and removal, the Optical Denoising Unit (ODU) can generate new images using a minimal number of these passive optical modulation layers. Since these layers do not need power or active control, they do not introduce any latency or energy overhead. Constrained only by optoelectronic input and readout hardware, this approach has the potential to significantly reduce the computational time and energy consumption of diffusion models, specifically performing inference in more sustainable and scalable ways.



Conventional Diffusion Model Sampling

Optical Diffusion Model Sampling

Figure 1: Comparison between conventional and proposed methods of image generation based on diffusion models. The conventional method runs on digital electronics based computing units such as GPUs or TPUs. The proposed method utilizes an optical denoising unit that is formed by passive optical layers. The image to be denoised is sent to the system with a modulator and the output is read out with a detector.

The main contributions of this study are:

- The propagation of light through multiple modulation layers is programmed to perform denoising diffusion image generation by predicting and transmitting the noise term in the input images. The system uses only a single modulation plane and multiple reflections.
- A time-aware denoising policy is specifically designed for analog optical computing hardware. This policy facilitates the use of passive building blocks to achieve multi-step computing at low power, translating the time-embedding in digital Denoising Diffusion Probabilistic Models (DDPMs) into optical hardware.
- An online learning algorithm is introduced for training ODUs in real-life scenarios, where alignment and calibration errors exist. The algorithm tracks and alleviates experimental discrepancies with constant updates to a digital twin (DT) during training time.

2 Related Work

Diffusion models have become popular, with their superior image generation performance compared to Generative Adversarial Networks (GANs) [13]. High-resolution, guided diffusion process is currently widely utilized for on-demand image generation [6, 7, 14]. One of the main concerns with these highly capable models is the significant time taken for generating new samples, which can exceed 10 seconds for each high-resolution image [6]. Different methods have been proposed to alleviate this condition and improve the efficiency of diffusion models. While Latent Diffusion Models [6] work on a lower dimensional representation of images to decrease the computational load, Denoising Diffusion Implicit Models [15] introduce a deterministic and non-Markovian sampling process to reduce the number of required steps. Similarly, FastDPM [16], uses domain-specific conditional information for faster sampling with diffusion models. Another approach is to distill multiple denoising steps to a single one with a teacher-student setting [17]. As these methods aim to decrease iterative denoising steps required for sampling through algorithmic innovations, the potential improvements obtained by exploiting the repetitive nature of these models on the computing hardware side remain to be seen.

Optical processors have shown substantial energy efficiency improvements, particularly with larger model sizes, potentially outperforming current digital systems [18]. They can be implemented through various architectures, each leveraging different aspects of optical technology to perform computations. Free-space optical networks use spatial light modulators or fixed modulation layers for performing matrix multiplications and convolutions as light propagates [19–21], making them highly efficient for image processing tasks. These applications include super-resolution [22], noise removal [23], and implementation of convolutional neural network layers [24], which are shown to competitively perform different downstream tasks such as segmentation [25]. On the other hand, photonic integrated circuits utilize optical components like Mach-Zehnder interferometers, microring resonators, and waveguides on a single chip, enabling compact vector-matrix multipliers [26, 27]. Together, these developments highlight the transformative potential of optical computing in enhancing the performance and efficiency of computationally intensive tasks.

Considering the significant computational demands of denoising tasks, there is a clear need for specialized hardware to scale these operations effectively. Despite the advancements in optics, deep learning, and optical image processing, the realization of an optical diffusion denoiser remains a gap in current research. Bridging this gap could leverage the synergy between these fields to develop highly efficient and scalable solutions for denoising diffusion models.

In addition to their wide range of advantages, optical computing systems also have disadvantages related to the energy cost of modulation and detection of light, its limited programmability, and experimental precision. Calculating the gradients of the experimental loss through the optical wave propagation model allows for a close match between optical experiments and computational models [19]. Similarly, a pre-trained neural network-based emulator of a physical system can also be used for the same purpose [28]. Moreover, it is crucial to perform as many computations as possible with the data while in the optical domain to avoid energy and time expenditure of optoelectronic devices.

3 Description of the Study

3.1 Denoising Diffusion Models

DDPMs progressively corrupt data with Gaussian noise in a forward process and subsequently learn to reverse this corruption through a denoising process. This way they can generate new data samples that closely resemble the training data distribution. The forward diffusion process involves the sequential corruption of a data sample $r_0 \sim q(r_0)$ through the addition of Gaussian noise over T timesteps. At each timestep t, the data sample r_{t-1} is perturbed to produce r_t, $r_t = \sqrt{1-\beta_t} r_{t-1} + \sqrt{\beta_t}\epsilon_t$, where $\beta_t \in (0, 1)$ is a variance schedule that determines the amount of noise added and $\epsilon_t \sim \mathcal{N}(0, \mathbf{I})$ is standard Gaussian noise. This process transforms the original data into nearly pure noise by timestep T.

The reverse denoising process in DDPMs aims to reconstruct the original data from a highly noisy sample. Starting from completely Gaussian noise $r_T \sim \mathcal{N}(0, \mathbf{I})$, the sample is iteratively denoised by removing the prediction of ϵ_t in the image, $\epsilon_\theta(r_t, t)$, which is provided by a trained neural network:

$$r_{t-1} = \frac{1}{\sqrt{1-\beta_t}}(r_t - \beta_t \epsilon_\theta(r_t, t)), \tag{1}$$

The training objective of the neural network can be simplified to minimize the mean squared error (MSE) between the true noise ϵ_t and the predicted noise $\epsilon_\theta(r_t, t)$, $\mathcal{L} = \mathbb{E}_{t,r_0,\epsilon_t}\left[\|\epsilon_t - \epsilon_\theta(r_t, t)\|^2\right]$

https://doi.org/10.52202/079017-1887

where t is uniformly sampled from $\{1, \ldots, T\}$. Finally, to generate new data samples, the model starts with a sample $r_T \sim \mathcal{N}(0, \mathbf{I})$ and applies the learned reverse transitions iteratively.

3.2 Propagation of Modulated Light Beams

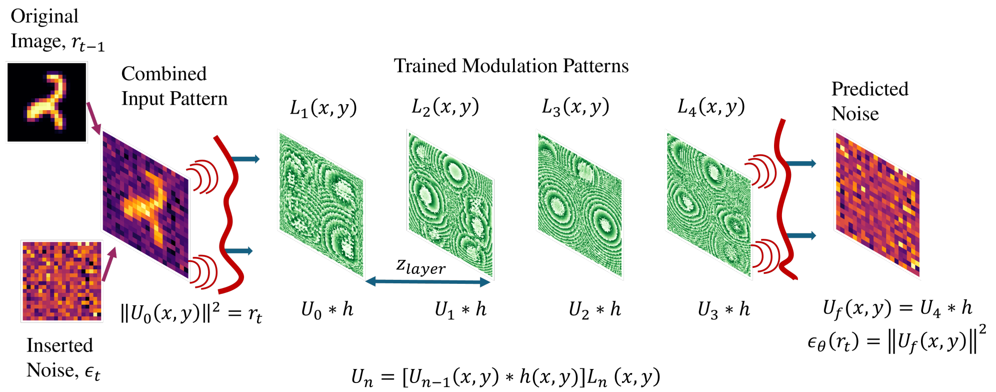


Figure 2: The main operation principle of ODU. Consequent modulation and free space propagation events can be represented with multiplication and convolution operations. When the input beam $U_0(x, y)$, which is patterned with noisy input images, r_t, is introduced to the ODU, the output intensity pattern $\|U_f(x, y)\|^2$ corresponds to the trained optical system's prediction of the noise component in the input pattern, $\epsilon_\theta(r_t)$.

In this study, a denoising framework is presented by combining the modulation of a light beam with consequent transparent or reflective patterns and its propagation in free space (environments such as vacuum or air, where the refractive index of light is approximately 1), as shown in Figure 2. This process can be explained by the Fresnel diffraction theory since the features on the layers are not only larger than the optical wavelengths but only sufficiently smaller than the distance between different modulation layers [29]. According to this formalism, the electromagnetic field after propagating a distance z in free space, $U(x, y, z)$, can be calculated from its distribution at $z = 0$ by convolution with "the impulse response of free space", $h(x, y)$:

$$U(x, y, z) = U(x, y, 0) * h(x, y), \text{ where } h(x, y) = \frac{e^{jkz}}{j\lambda z} \exp\left[\frac{jk}{2z}\left(x^2 + y^2\right)\right] \tag{2}$$

Here, k denotes the wavenumber of the field, and λ is the wavelength. In other words, the field's value at the plane of $z = z_0$, at a given location (x, y), is the weighted sum of the values at $z = 0$, where the weight of each location is determined by the response function. Being complex numbers, all of these weights have the same magnitude but their phase depends on the location. In the frequency domain, the transfer function of free space becomes

$$H\left(f_X, f_Y\right) = \exp\left[j2\pi\frac{z}{\lambda}\sqrt{1 - \left(\lambda f_X\right)^2 - \left(\lambda f_Y\right)^2}\right]. \tag{3}$$

This indicates that for spatial frequencies larger than $1/\lambda$, the magnitude of the transfer function decays to zero exponentially. Hence, only features that are larger than the wavelength of the light can propagate to the far field. Moreover, frequency domain expression of diffraction, Eqn. 3, allows for also the efficient digital simulation of the propagation of light in free space with the utilization of Fast Fourier Transforms (FFTs) in a parallelized manner. Later on, we will benefit from this fact for GPU accelerated training of the diffractive modulation layers.

The proposed method applies trainable weights to the light beam at consequent planes with thin modulation layers. The interaction between layers and light can be represented as a point-wise multiplication between the incident field and the layer, which is followed by the propagation of the field in free space until the next layer,

https://doi.org/10.52202/079017-1887

$$U_n(x,y) = [U_{n-1}(x,y) * h(x,y)]L_n(x,y), \tag{4}$$

where $U_n(x,y)$ is the field distribution right before reaching the modulation layer n, and $L_n(x,y)$ is the complex modulation coefficient distribution of the trained modulation layers or the weights of the optical diffusion model. $L(x,y) = |L(x,y)|e^{i\phi(x,y)}$ can be only a real number($\phi(x,y) = 0$), just phase modulation ($|L(x,y)| = 1$) or an arbitrary complex number depending on the implemented modulation principle. In this paper, we demonstrate our approach with a phase-only liquid crystal spatial light modulator (SLM), which can set $\phi(x,y)$ to any value in the range of $[0, 2\pi]$ electronically, and $|L(x,y)| \approx 1$ everywhere.

3.3 Training of Optical Modulation Layers

As described in section 3.2, propagation of light can be analytically explained in a succinct manner for the scale considered in this study ($z_{layer} >> d_{pixel} > \lambda$). This allows for defining some free variables in this representation, such as refractive index distribution $L(x,y)$ or input wavefront distribution $U_0(x,y)$, and optimizing these variables for minimizing a cost function. The gradients of these variables can be found with either manual calculation [30] or automatic differentiation packages [20]. In this study, our first goal is to find optimal modulation layers, or refractive index distributions, such that after the light beam encoded with the noisy images propagates through them only the predicted noise term reaches the detector, as shown in Figure 2. Moreover, the denoising network should be aware of the given timestep in the diffusion process while predicting the noise, $\epsilon_\theta(x_t, t)$, so that it would have *a priori* information about the variance of the noise term. Since noise level awareness is a crucial aspect of successful sample generation, most of the current implementations of diffusion models utilize time-embedding layers to modify activations of the neural network across different layers depending on the diffusion time step. Instead, the proposed method divides the diffusion timeline consisting of T timesteps into M subsets, and for each subset of time frames $\{S_m\}_{m=1}^{M}$, trains a separate set of modulation layers $\{L_n^m\}_{n=1}^{N}$ each containing N layers. Then, for $t \in S_m$, the noise prediction,$\epsilon_{\theta_m}(x)$ becomes only a function of x. In this scheme, the training objective for each time step is

$$\mathcal{L}_t = \mathbb{E}_{x_0,\epsilon}\left[\|\epsilon - \epsilon_{\theta_m}(x_t)\|^2\right] \tag{5}$$

where total loss is the sum over all ranges:

$$\mathcal{L} = \sum_{i=m}^{M} \sum_{t \in S_m} \mathcal{L}_t. \tag{6}$$

This decoupling of denoising at different timesteps by removing time-embedding layers also eliminates the necessity for digital computations to modifications at different layers. By circumventing this problem we perform denoising all-optically. Moreover, a fixed optical modulation pattern performs denoising at multiple consequent timesteps. For instance, we later demonstrate that for $T = 1000, M = 10$ creates optimal results. So, a single layer set can process 100 timesteps and the entire sampling workflow can be operated with only 10 fixed parallel devices, or with only 10 updates to the SLM.

After defining the forward calculation of the system with the analytical explanation of light propagation and the loss function as the mean square error of noise prediction (Eqn. 6), the trainable parameters of the system $\{L_n^m\}_{n=1}^{N}(x,y)$ are optimized by automatic differentiation [31].

4 Results

4.1 All-Optical Denoising based Image Generation

Following the same experimental settings with the initial DDPM study [2], we set $T = 1000$ and β values to be in the linear range between $\beta_1 = 10^{-4}$ to $\beta_T = 0.02$. The results in Figure 3 are reported with the beam propagation model (Eqn. 3) of the optical system designed to have 300×300 pixels per layer and four modulation layers. The number of layer sets (M) is 10. Several intermediate results alongside final outputs at $T = 1000$ are reported in Figure 3 for 3 classes of the

https://doi.org/10.52202/079017-1887

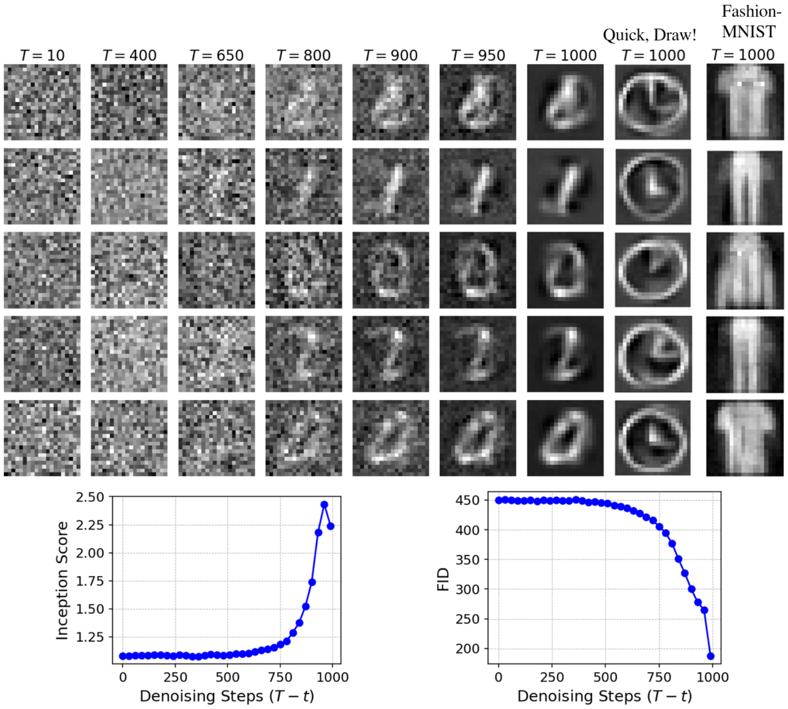


Figure 3: Images generated by the Optical Diffusion Model at different timesteps and when trained with various datasets. The generated images and their corresponding Inception and FID scores are calculated between timesteps $T = 10$ to $T = 950$ are acquired after training with the MNIST digits dataset. Final outputs at time $T = 1000$, acquired from ODUs trained for the MNIST digits samples have FID = 206.6, for Fashion MNIST, FID = 227.7 and for Quick, Draw!, FID = 131.4

MNIST digits [32], Fashion-MNIST [33] and the clock category of the Quick, Draw! datasets [34]. Furthermore, the evaluation of image generation quality metrics, Inception Score (IS) and Fréchet Inception Distance (FID), which are detailed in Appendix A.6, across different generation timesteps captures the improved realism of images with the optical diffusion procedure.

4.2 Effects of Optical Model's Dimensionality on the Image Denoising and Generation Performance

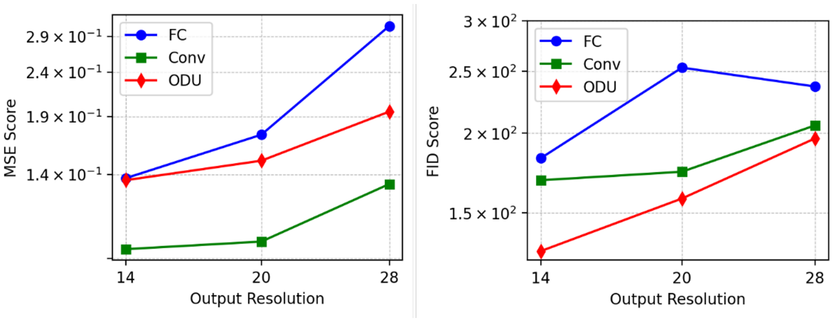


Figure 4: Scaling of the denoising capabilities (left) and generation performance (right) of Optical Diffusion, and pure digital convolutional U-Net and fully connected networks with the output image resolution.

This section provides further analysis with different output dimensions and parameter counts along with the comparisons with purely digital implementations to quantify Optical Diffusion's scalability to large-scale diffusion problems. The first investigation is into the performance with higher resolution datasets; two digital architectures of similar performance with the ODU, one being fully connected and the other convolutional U-Net [35], are trained for the same tasks with the ODU, generating images with the MNIST digits dataset at different resolutions. Their architecture is detailed in Appendix Table 1. The results shown in Figure 4 indicate that ODU consistently outperforms the two digital neural networks, and all three scale in a similar manner both in terms of denoising and generation performances when the generated image dimension is increased while the model sizes are kept constant.

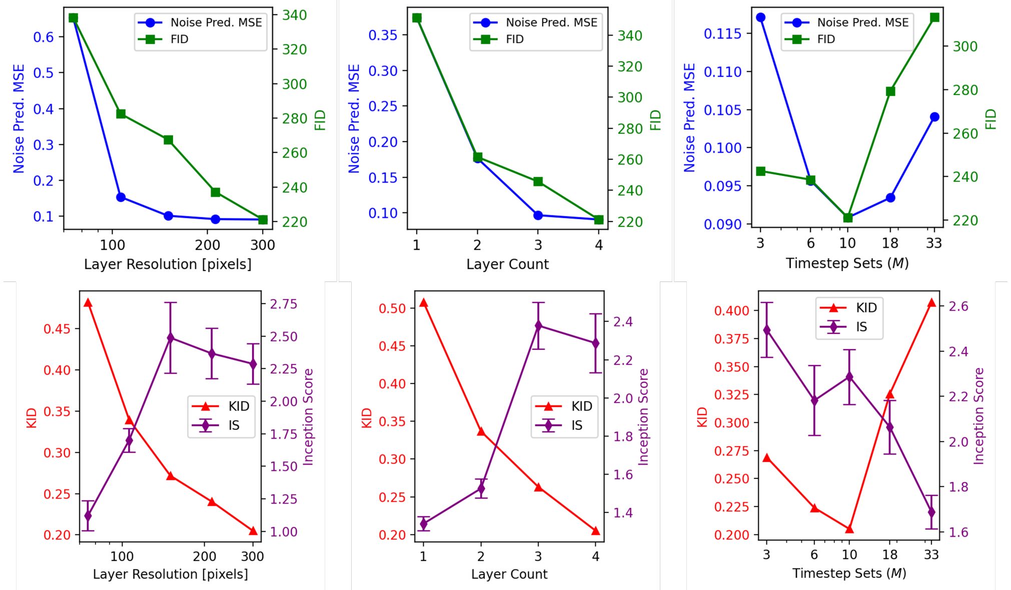


Figure 5: The dependency of denoising performance (MSE) and generation quality scores(FID, KID and Inception score), on the hyperparameters of the ODUs (number of pixels of optical modulation layers, number of modulation layers and number of denoising layer sets (M)).

Secondly, the scaling of Optical Diffusion's performance with respect to the number of total parameters is probed through three hyperparameters: layer resolution, layer count, and timestep sets. The effects are tracked with different metrics, MSE for denoising, FID, IS and Kernel Inception Distance (KID) for the generation quality. In Figure 5, we observe that, as in digital neural networks, there is a clear tendency to perform better with a larger number of trainable parameters when layer resolution and layer count are increased.

On the other hand, having a larger number of denoising layer/timestep sets improves the results only until they reach a certain level. Afterward, increasing the number of sets is detrimental as shown in Figure 5. As the total number of training steps is fixed in this experiment, increasing the number of timestep sets decreases the training sample count per layer set, hence potentially deteriorating the performance after a particular threshold, which is found to be $M = 10$.

Through the aggregation of data points acquired with different layer resolutions and counts in Figure 5, the relationship between the total trainable parameter count of ODUs and their image generation performance is depicted in Figure 6. This relationship remarkably follows the same widely accepted, power-law trend of digital generative models [36]. Most significantly, when the optical implementation is fitted to a power-law equation, the exponential of the power law (-0.15) is approximately the same as the reported value (-0.16) for large-scale image generation networks in [36]. This fit parameter gives the slope of the line in the logarithmic plot, indicating how fast the generation performance scales with the number of parameters, in this case showing that ODU improves its performance at a similar speed with large-scale digital image generation networks while its parameter count is increased. The single outlier in this trend is the case where there is only a single modulation layer, which does not benefit from the multiple optical modulations aspect of the proposed architecture.

https://doi.org/10.52202/079017-1887

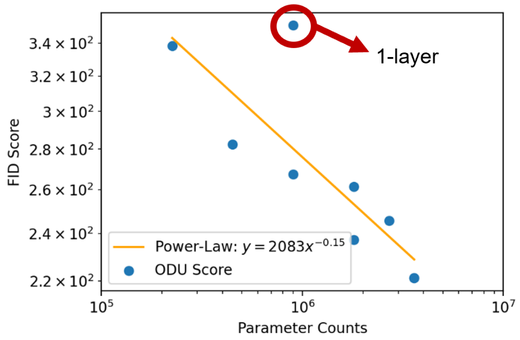


Figure 6: The relationship between the total number of parameters in an ODU and its generation performance in terms of FID scores.

4.3 Higher Experimental Fidelity with the Online Learning Algorithm

To address the challenge of training an optical system with imperfect calibration, as faced in many other analog computing paradigms, we propose an online learning algorithm that updates and leverages a DT during training. During inference, the DT does not incur any additional overhead. The DT ($\tilde{f}_{\theta_{\text{layers}},\theta_{\text{alignment}}}$) again utilizes Fresnel diffraction based model of light propagation, as a surrogate to compute gradients and guide the optimization of the system's trainable parameters. However, matching the DT's parameters ($\theta_{\text{alignment}}$), for instance, input beam angle, precise locations of the layers, and their angles, perfectly to the experimental conditions of the physical system is a challenging task. Therefore, during each iteration of training, the output of the experiment ($f_{\theta_{\text{layers}}}$) and DT ($\tilde{f}_{\theta_{\text{layers}},\theta_{\text{alignment}}}$) is compared and the DT's parameters are updated accordingly, as shown in Algorithm 1.

Algorithm 1 Online Learning Algorithm

```
Initialize physical system $f_{\theta_{\text{layers}}}$ with parameters $\theta_{\text{layers}}$
Initialize DT $\tilde{f}_{\theta_{\text{layers}},\theta_{\text{alignment}}}$ with parameters $\theta_{\text{layers}}, \theta_{\text{alignment}}$
while not converged do
    Forward Pass:
    Input data $\mathbf{x}$ into the physical system $f_{\theta_{\text{layers}}}$
    Obtain physical system output $\mathbf{y}_f = f_{\theta_{\text{layers}}}(\mathbf{x})$
    Compute error $E = \text{loss}(\mathbf{y}_f, \mathbf{y}_{\text{target}})$
    Backward Pass:
    Compute Jacobian of DT at $\mathbf{x}$, $\mathbf{J} = \frac{\partial \mathbf{y}_{\tilde{f}}}{\partial \theta_{\text{layers}}}$
    Compute gradients $\nabla_{\theta_{\text{layers}}} = \mathbf{J}^T \cdot \frac{\partial E}{\partial \mathbf{y}_f}$
    Update physical system parameters $\theta_{\text{layers}} \leftarrow \theta_{\text{layers}} - \eta \nabla_{\theta_{\text{layers}}}$
    DT Refinement:
    Obtain DT output $\mathbf{y}_{\tilde{f}}$
    Compute MSE between DT and physical system outputs $L = \text{MSE}(\mathbf{y}_{\tilde{f}}, \mathbf{y}_f)$
    Compute gradients $\nabla_{\theta_{\text{alignment}}} = \frac{\partial L}{\partial \theta_{\text{alignment}}}$
    Update DT alignment parameters $\theta_{\text{alignment}} \leftarrow \theta_{\text{alignment}} - \alpha \nabla_{\theta_{\text{alignment}}}$
end while
```

In parallel, the DT is also employed to compute the gradients of the trainable parameters of the experiment, with respect to the output($\frac{\partial \mathbf{y}_{\tilde{f}}}{\partial \theta_{\text{layers}}}$). These gradients are then utilized to update the physical system's parameters through backpropagation, informed by the error obtained from the physical system. Concurrently, the DT is refined using the latest inputs and outputs from the physical system to better approximate its behavior, despite the initial parameter mismatches. This iterative process of forward and backward passes, coupled with the continuous refinement of the DT, enables the physical system to progressively improve its performance and align more closely with the desired outcomes.

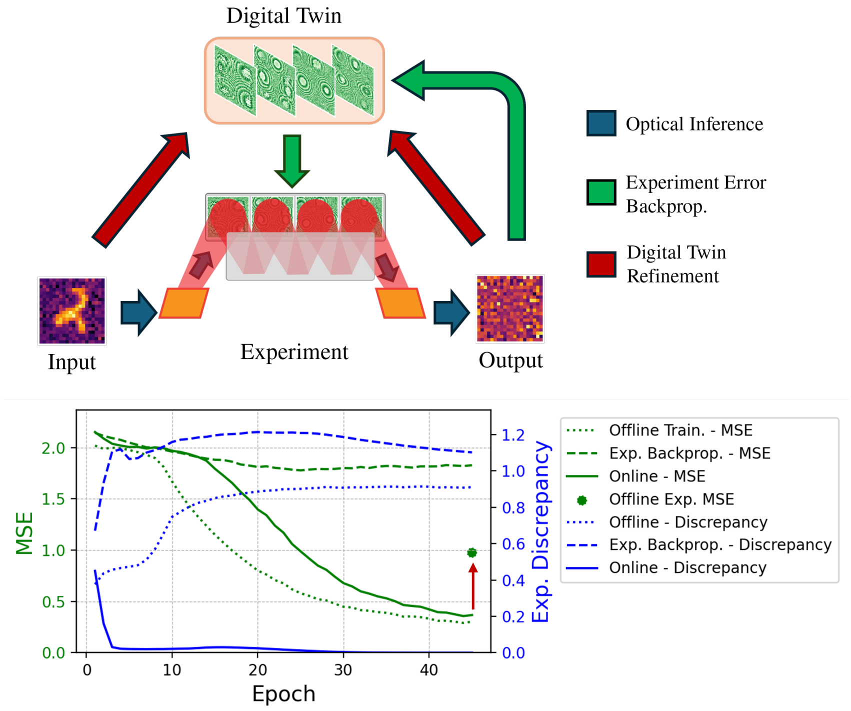


Figure 7: The upper block illustrates the online training scheme. The forward pass is calculated with the experiment (blue), while the gradients of the prediction error are backpropagated using a DT of the experiment (green) and updating the physical trainable parameters. The difference between the outputs of the experimental setup and the DT continuously refines the DT (red). The lower graph block compares offline and online training methods. Offline training relies on a pre-trained DT for both the forward and backward passes, with the experimental performance of this method indicated by the star. Experimental backpropagation executes the physical forward pass but does not incorporate DT refinement.

In Figure 7, we explore the efficiency of the proposed algorithm by modeling a possible experiment scenario. In this scenario, we define two different optical models, while actually both of them are simulations of the optical wave propagation for an exact insight into the algorithm, we designate the first one as the “optical experiment” by configuring it with the calibration angles obtained from the physical experiment. These four angles account for the slight misalignment of the experiment and define the input angle of the beam to the cavity and the angle between the mirror and the SLM, in x and y axes, all being in the range of a few milliradians and their measurement details being provided in Appendix A.1. The second model, considered as the DT, is initialized with their calibration angles 20% higher. We used 3 different algorithms, offline, experimental error backpropagation [28], and the proposed online training schemes. During training, MSE (training loss), and the discrepancy between the DT and the experiment’s outputs are tracked. The discrepancy, D, is inversely related to cross-correlation, C, of the experimental and the DT’s normalized outputs, O_{exp} and O_{dt} respectively, $D = 1 - C$, where $C = \sum_x \sum_y O_{\text{exp}}(x, y) O_{\text{dt}}(x, y)$.

Offline training improves the loss function when evaluated with the DT, but when evaluated in the experimental setting, it has a higher MSE, as shown with a star in Figure 7. When the online training method is used, the DT is aligned with the actual experiment swiftly and the experimental loss approximates the MSE of perfect calibration case, the results of this approach with the optical experiment are also provided in Appendix Figure 10. Backpropagating the experimental loss, MSE does not decrease significantly. However, for smaller misalignment, this method was also demonstrated to converge. This experiment implements multiple modulation layers on a single device, with a single phase-only SLM and a mirror in parallel to resource-efficiently prototype the proposed computing method, as shown in Figure 7 and detailed in Appendix A.3.

5 Conclusion

In this study, we introduced an optical diffusion denoising framework for image generation, utilizing a time-aware denoising strategy that enables optical low-power realization. By exploiting light propagation through transparent media, ODU effectively reduces noise in images, with a much smaller energy budget compared to electronics since optical wave propagation has a very small intrinsic loss while acquiring comparable, or better quality. This is especially interesting because diffusion models are currently one of the most costly generative artificial intelligence models due to their repetitive denoising process, with a correspondingly large environmental impact [37].

The integration of a time-aware policy enables Optical Diffusion to adjust light modulation dynamically according to different stages of the denoising process, thus improving image quality with a minimal number of changes to the modulation layers or parallel optical processing units. Looking ahead, the incorporation of larger modulation layers with more parameters and the exploration of nonlinear optical effects could enhance the functionality of the system. Scaling analyses also show evidence for the ODU to improve its performance at the same rate as digital models while increasing its size. These potential improvements suggest promising directions for future research in expanding the range of applications for this technique.

The proposed method can utilize off-the-shelf consumer electronics such as digital micromirror devices that can be found in portable projectors for input modulation and CMOS cameras for recording output prediction. The online learning algorithm accounts for variations between these devices and closes the gap between the analytical and experimental realization of the ODU. On the other hand, as analyzed in detail in Appendix A.4, these devices have the potential of implementing denoising steps on the order of microsecond latencies while consuming a few Watts only. With the utilization of high-speed light modulation technologies, million-frames-per-second-level denoising can be achieved again with Watt level energy consumption, while still utilizing the proposed approach for predicting noise in provided images [38].

Acknowledgments and Disclosure of Funding

This work was supported by Google Research under grant number 901381. We would like to thank Research Computing Platform at EPFL and Google Cloud Services for providing access to compute units.

References

[1] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep Unsupervised Learning using Nonequilibrium Thermodynamics," in *Proceedings of the 32nd International Conference on Machine Learning*. PMLR, Jun. 2015, pp. 2256–2265, iSSN: 1938-7228. [Online]. Available: https://proceedings.mlr.press/v37/sohl-dickstein15.html

[2] J. Ho, A. Jain, and P. Abbeel, "Denoising Diffusion Probabilistic Models," in *Advances in Neural Information Processing Systems*, vol. 33. Curran Associates, Inc., 2020, pp. 6840–6851. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html

[3] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-Based Generative Modeling through Stochastic Differential Equations," Oct. 2020. [Online]. Available: https://openreview.net/forum?id=PxTIG12RRHS&utm_campaign=NLP%20News&utm_medium=email&utm_source=Revue%20newsletter

[4] A. Q. Nichol and P. Dhariwal, "Improved Denoising Diffusion Probabilistic Models," in *Proceedings of the 38th International Conference on Machine Learning*. PMLR, Jul. 2021, pp. 8162–8171, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v139/nichol21a.html

https://doi.org/10.52202/079017-1887

[5] A. Jalal, M. Arvinte, G. Daras, E. Price, A. Dimakis, and J. Tamir, “Robust Compressed Sensing MRI with Deep Generative Priors,” Nov. 2021. [Online]. Available: https://openreview.net/forum?id=wHoIjrT6MMb

[6] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-Resolution Image Synthesis with Latent Diffusion Models,” in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. New Orleans, LA, USA: IEEE, Jun. 2022, pp. 10 674–10 685. [Online]. Available: https://ieeexplore.ieee.org/document/9878449/

[7] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, S. K. S. Ghasemipour, R. Gontijo-Lopes, B. K. Ayan, T. Salimans, J. Ho, D. J. Fleet, and M. Norouzi, “Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding,” Oct. 2022. [Online]. Available: https://openreview.net/forum?id=08Yk-n5l2Al

[8] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, “Image Super-Resolution via Iterative Refinement,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4713–4726, Apr. 2023, conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence. [Online]. Available: https://ieeexplore.ieee.org/document/9887996

[9] S. Küfeoğlu and M. Özkuran, “Bitcoin mining: A global review of energy and power demand,” *Energy Research & Social Science*, vol. 58, p. 101273, Dec. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214629619305948

[10] C. E. Leiserson, N. C. Thompson, J. S. Emer, B. C. Kuszmaul, B. W. Lampson, D. Sanchez, and T. B. Schardl, “There’s plenty of room at the Top: What will drive computer performance after Moore’s law?” *Science*, vol. 368, no. 6495, p. eaam9744, Jun. 2020, publisher: American Association for the Advancement of Science. [Online]. Available: https://www.science.org/doi/10.1126/science.aam9744

[11] G. Wetzstein, A. Ozcan, S. Gigan, S. Fan, D. Englund, M. Soljačić, C. Denz, D. A. B. Miller, and D. Psaltis, “Inference in artificial intelligence with deep optics and photonics,” *Nature 2020 588:7836*, vol. 588, no. 7836, pp. 39–47, Dec. 2020, 101 citations (Semantic Scholar/DOI) [2022-01-23] Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41586-020-2973-6

[12] P. L. McMahon, “The physics of optical computing,” *Nature Reviews Physics*, vol. 5, no. 12, pp. 717–734, Dec. 2023, publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s42254-023-00645-5

[13] P. Dhariwal and A. Q. Nichol, “Diffusion Models Beat GANs on Image Synthesis,” Nov. 2021. [Online]. Available: https://openreview.net/forum?id=AAWuCvzaVt

[14] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, “Hierarchical Text-Conditional Image Generation with CLIP Latents,” Apr. 2022. [Online]. Available: https://arxiv.org/abs/2204.06125v1

[15] J. Song, C. Meng, and S. Ermon, “Denoising Diffusion Implicit Models,” Oct. 2020. [Online]. Available: https://openreview.net/forum?id=St1giarCHLP

[16] Z. Kong and W. Ping, “On Fast Sampling of Diffusion Probabilistic Models,” Jun. 2021, arXiv:2106.00132 [cs]. [Online]. Available: http://arxiv.org/abs/2106.00132

[17] T. Salimans and J. Ho, “Progressive Distillation for Fast Sampling of Diffusion Models,” Jun. 2022, arXiv:2202.00512 [cs, stat]. [Online]. Available: http://arxiv.org/abs/2202.00512

[18] M. G. Anderson, S.-Y. Ma, T. Wang, L. G. Wright, and P. L. McMahon, “Optical Transformers,” Feb. 2023, arXiv:2302.10360 [physics]. [Online]. Available: http://arxiv.org/abs/2302.10360

[19] T. Zhou, X. Lin, J. Wu, Y. Chen, H. Xie, Y. Li, J. Fan, H. Wu, L. Fang, and Q. Dai, “Large-scale neuromorphic optoelectronic computing with a reconfigurable diffractive processing unit,” *Nature Photonics*, vol. 15, no. 5, pp. 367–373, May 2021, number: 5 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41566-021-00796-w

[20] X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, and A. Ozcan, “All-optical machine learning using diffractive deep neural networks,” *Science*, vol. 361, no. 6406, pp. 1004–1008, Sep. 2018, 523 citations (Semantic Scholar/DOI) [2022-01-23] Publisher: American Association for the Advancement of Science _eprint: 1804.08711.

[21] J. Hu, D. Mengu, D. C. Tzarouchis, B. Edwards, N. Engheta, and A. Ozcan, "Diffractive optical computing in free space," *Nature Communications*, vol. 15, no. 1, p. 1525, Feb. 2024, publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41467-024-45982-w

[22] Işıl, D. Mengu, Y. Zhao, A. Tabassum, J. Li, Y. Luo, M. Jarrahi, and A. Ozcan, "Super-resolution image display using diffractive decoders," *Science Advances*, vol. 8, no. 48, p. eadd3433, Dec. 2022, publisher: American Association for the Advancement of Science. [Online]. Available: https://www.science.org/doi/10.1126/sciadv.add3433

[23] Işıl, T. Gan, F. O. Ardic, K. Mentesoglu, J. Digani, H. Karaca, H. Chen, J. Li, D. Mengu, M. Jarrahi, K. Akşit, and A. Ozcan, "All-optical image denoising using a diffractive visual processor," *Light: Science & Applications*, vol. 13, no. 1, p. 43, Feb. 2024, publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41377-024-01385-6

[24] J. Chang, V. Sitzmann, X. Dun, W. Heidrich, and G. Wetzstein, "Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification," *Scientific Reports*, vol. 8, no. 1, p. 12324, Aug. 2018, number: 1 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41598-018-30619-y

[25] K. Wei, X. Li, J. Froech, P. Chakravarthula, J. Whitehead, E. Tseng, A. Majumdar, and F. Heide, "Spatially varying nanophotonic neural networks," *arXiv preprint arXiv:2308.03407*, 2023.

[26] Y. Shen, N. C. Harris, S. Skirlo, M. Prabhu, T. Baehr-Jones, M. Hochberg, X. Sun, S. Zhao, H. Larochelle, D. Englund, and M. Soljacic, "Deep learning with coherent nanophotonic circuits," *Nature Photonics*, vol. 11, no. 7, pp. 441–446, Jun. 2017, 925 citations (Semantic Scholar/DOI) [2022-01-23] Publisher: Nature Publishing Group _eprint: 1610.02365.

[27] A. N. Tait, T. Ferreira de Lima, M. A. Nahmias, H. B. Miller, H.-T. Peng, B. J. Shastri, and P. R. Prucnal, "Silicon Photonic Modulator Neuron," *Physical Review Applied*, vol. 11, no. 6, p. 064043, Jun. 2019, publisher: American Physical Society. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevApplied.11.064043

[28] L. G. Wright, T. Onodera, M. M. Stein, T. Wang, D. T. Schachter, Z. Hu, and P. L. McMahon, "Deep physical neural networks trained with backpropagation," *Nature*, vol. 601, no. 7894, pp. 549–555, Jan. 2022, number: 7894 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41586-021-04223-6

[29] J. Goodman, *Introduction to Fourier Optics*, 4th ed. New York: W. H. Freeman, May 2017.

[30] U. S. Kamilov, I. N. Papadopoulos, M. H. Shoreh, A. Goy, C. Vonesch, M. Unser, and D. Psaltis, "Learning approach to optical tomography," *Optica*, vol. 2, no. 6, pp. 517–522, Jun. 2015, publisher: Optica Publishing Group. [Online]. Available: https://opg.optica.org/optica/abstract.cfm?uri=optica-2-6-517

[31] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," Oct. 2017. [Online]. Available: https://openreview.net/forum?id=BJJsrmfCZ

[32] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[33] H. Xiao, K. Rasul, and R. Vollgraf. (2017) Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms.

[34] "googlecreativelab/quickdraw-dataset," Oct. 2024, original-date: 2017-05-09T18:28:32Z. [Online]. Available: https://github.com/googlecreativelab/quickdraw-dataset

[35] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer, 2015, pp. 234–241.

[36] T. Henighan, J. Kaplan, M. Katz, M. Chen, C. Hesse, J. Jackson, H. Jun, T. B. Brown, P. Dhariwal, S. Gray *et al.*, "Scaling laws for autoregressive generative modeling," *arXiv preprint arXiv:2010.14701*, 2020.

[37] S. Luccioni, Y. Jernite, and E. Strubell, "Power hungry processing: Watts driving the cost of ai deployment?" in *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and*

https://doi.org/10.52202/079017-1887

Transparency, ser. FAccT '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 85–99. [Online]. Available: https://doi.org/10.1145/3630106.3658542

[38] C. L. Panuski, I. Christen, M. Minkov, C. J. Brabec, S. Trajtenberg-Mills, A. D. Griffiths, J. J. D. McKendry, G. L. Leake, D. J. Coleman, C. Tran, J. St Louis, J. Mucci, C. Horvath, J. N. Westwood-Bachman, S. F. Preble, M. D. Dawson, M. J. Strain, M. L. Fanto, and D. R. Englund, "A full degree-of-freedom spatiotemporal light modulator," *Nature Photonics*, vol. 16, no. 12, pp. 834–842, Dec. 2022, publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41566-022-01086-9

A Appendix

A.1 Differentiable Modelling of Light Propagation in ODUs

To benefit from parallelized and optimized FFT algorithm and automatic differentiation, the wave propagation in the proposed system is modeled in PyTorch environment with a Split-Step Fourier formalism which is derived from Eqn. 2. The diffraction step of the propagation is calculated with a nonparaxial diffraction kernel [1] in Fourier domain and effects such as reflection or modulation of the light beam with layer parameters are applied in the spatial domain. Such that the electric field after propagating a distance Δz becomes:

$$E(x, y, z + \Delta z) = \mathcal{F}^{-1}\{\mathcal{F}\{E(x, y, z)R(x, y)\}e^{-\frac{j\Delta z\left(k_x^2+k_y^2\right)}{k+\sqrt{k^2-k_x^2-k_y^2}}}\} \tag{7}$$

In addition to the parameters of layers $L_n^m(x, y)$ (or simply θ_{layers}), the spatial term $R(x, y)$ can include the angle changes of the beam. For instance, if the beam is not perpendicular to the SLM or the mirror, the reflection creates a change in the angle of the beam, $\Delta\alpha = (\alpha_x, \alpha_y)$. Then, on the SLM plane $R_m(x, y) = L_m(x, y)e^{-jk(x\sin\alpha_x+y\sin\alpha_y)}$, where $R_m(x, y)$ is the compound spatial term, $L_m(x, y)$ is the modulation parameters at layer m, and $e^{-jk(x\sin\alpha_x+y\sin\alpha_y)}$ is the operator that changes the direction of the wave propagation vector. Similarly to the SLM, also the angle of the mirror determines the propagation direction of the beam, which can be included in the model as $R_{\text{mirror}}(x, y) = e^{-jk(x\sin\alpha_x+y\sin\alpha_y)}$. To calibrate the model of the experiment with the actual experiment, we define three trainable alignment parameters, z_{gap} (distance between the mirror and the SLM), $\Delta\alpha_{mirror}$ (twice the angle between the mirror and the SLM), and $\Delta\alpha_{beam}$ (twice the angle between the input beam and the SLM). This group of trainable model parameters is called $\theta_{alignment}$ and as its constituents appear in the forward model within differentiable functions, the auto-differentiation algorithm [2] can calculate their derivatives with respect to the error between the predicted camera images and the acquired ones. $\theta_{alignment}$ is initially pre-trained with experiments placing square shaped π phase differences randomly on the SLM. During the online training procedure, it is further trained with the data from denoising experiments.

A.2 Details of Scaling Studies

A.2.1 Scaling with Output Image Resolution

Table 1: Properties of Optical, Convolutional U-Net and Fully Connected Denoising Networks, the same training settings with the main text are used in all experiments with the MNIST digits dataset.

Architecture	Parameters	FLOPS/Step	Energy/Image [J]	Images/s
Fully Connected	19.6 M	39.0 M	1.74	41.3
Convolutional U-Net	220 K	3.11 M	5.37	13.9
ODU	3.6 M	Not Applicable	0.23	23.0

We investigated the change in the performance of Optical Diffusion when the image resolution changed while the model size was kept the same. This scaling behavior is also compared with well-established digital neural network architectures under the same diffusion settings on the MNIST digits dataset. The energy consumption and speed of the ODU in Appendix Table 1 are indicated for the simple laboratory implementation where the efficiency is not optimized, while the digital benchmarks are run on an Nvidia L4 GPU, one of the state-of-the-art devices available today.

Fully Connected Denoising Neural Network. This fully connected architecture consists of 4 fully connected layers with 1200 neurons, SiLU nonlinearity, and one-dimensional batch normalization layers following each fully connected layer. The outputs of the first and third layers are summed with time-embedding representations. Inputs are interpolated to 76 × 76 and flattened to vectors of 5776 elements, while the last layer outputs a same-sized vector which is again reshaped to 76 × 76 and scaled to the target resolution. During inference time, this network generated a batch of 64 images in $1.55\,\text{s}$, on an NVIDIA L4 GPU utilized 100% at $72\,\text{W}$. This amounts to 41.3 images/s at 1.74 J/image.

Convolutional U-Net Denoising Neural Network. This U-Net architecture [3] has 2 downsampling, 1 bottleneck, and 2 upsampling blocks, featuring a total of 32 convolutional layers with 3×3 kernels and SiLU nonlinearity. Every block also includes time embedding and batch normalization. Similarly, inputs are interpolated to 76×76 pixels and the same-sized outputs are scaled to the target resolution. During inference time, this network generated a batch of 64 images in $4.60\,\mathrm{s}$, on an NVIDIA L4 GPU utilized 100% at $72\,\mathrm{W}$. This amounts to 13.9 images/s at 5.37 J/image.

ODU. The ODUs consist of 10 sets of 4 optical layers, each with 300×300 modulation parameters, as detailed in Section 4.1. At an image rate of 23 kfps and total energy consumption of $5.3\,\mathrm{W}$ between the DMD and the camera, the generation can be operated at 23.0 images/s at 0.23 J/image. In this scenario, the SLM is assumed to be a passive device due to the very small number of updates.

In addition to the comparison in Figure 4 using the MNIST digits dataset up to a resolution of 28×28, we studied further the generation quality by training with the AFHQ dataset's cat class [4] at 40×40 resolution. The results in Appendix Figure 8 confirm the same successful scaling trend with the ODU overperforming the digital networks. Even though this relatively more complex problem necessitates larger and more capable denoising networks, similarly with the smaller scale experiments, Optical Diffusion obtained the best FID.

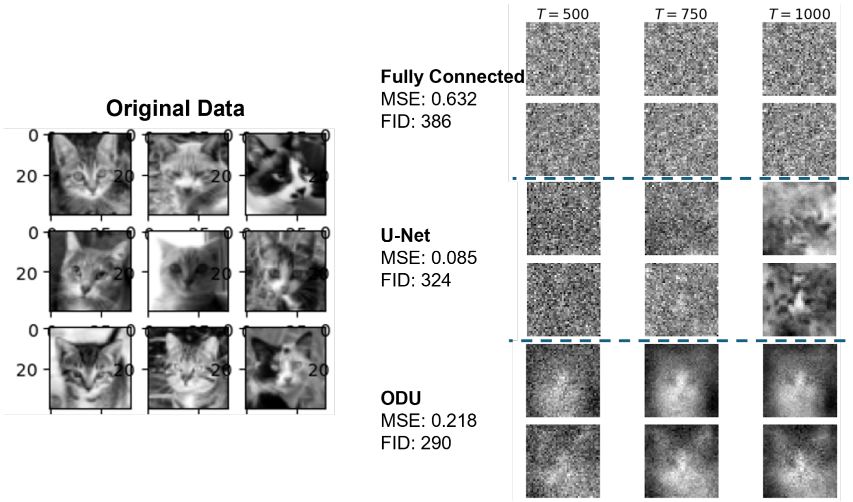


Figure 8: Comparison of image generation performances on the AFHQ dataset's cat class [4] at 40-by-40 resolution.

A.3 Details of the Experimental System and Online Learning

In the ODU, the trainable parameters are the pixel values on the modulation layers and are digitally adjusted using a computer model, as outlined in Appendix A.1. After optimizing these parameters on the computer, they are implemented across various layers in the experimental setup, utilizing the SLM. The beam generated by a continuous-wave dye laser (M-Squared Solstis 2000) at $\lambda = 850\,\mathrm{nm}$ reaches the phase-only SLM (Meadowlark HSP 1920-500-1200) after reflecting from a digital micromirror device (DMD), which can be used for spatial amplitude modulation. After 4 reflections on the modulating area SLM, the beam is imaged onto a CMOS camera (FLIR BFS-U3-04S2M-CS). An $11.6\,\mathrm{mm}$ mirror, placed $17.1\,\mathrm{mm}$ from the SLM display, captures four reflections. To direct the input beam toward the SLM, 4F imaging was employed, transferring the beam from the DMD, which serves as a programmable aperture ensuring the beam's alignment with the first modulation layer of the SLM. We assigned 260×260 pixel patches for the modulation layers on the SLM, which has a pixel pitch of $9.2\,\mu\mathrm{m}$. After the fourth reflection, the beam is imaged with another 4F imaging system onto a CMOS camera that records the output intensity as a 130×130 pixel image, where the camera's pixel pitch is $3.45\,\mu\mathrm{m}$. The examples of input patterns to the optical system, the corresponding output intensities at the camera plane and the resulting noise pattern predictions are provided in Appendix Figure 9. The noise predictions are obtained by the downsampling and normalization of the pixel values in the output intensity recordings.

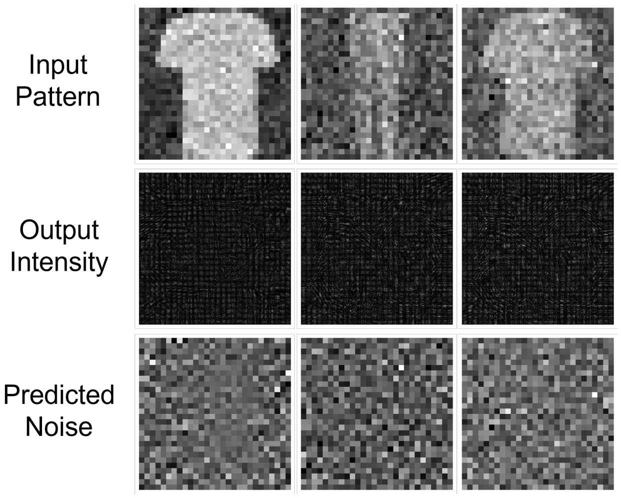


Figure 9: Some input patterns, output intensities at the camera plane, and the corresponding noise prediction for Fashion MNIST. The intensities on the camera plane are converted to noise predictions by downsampling and normalization.

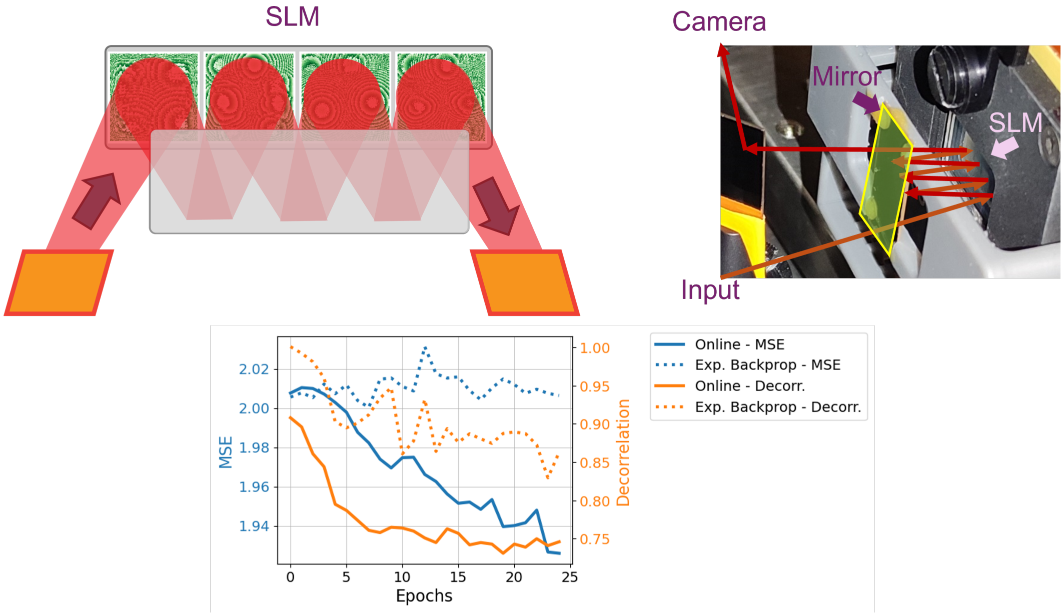


Figure 10: Schematic representation and photograph of the experimental system. Denoising error (MSE) and the decorrelation between the experimental system and the DT are plotted for online and only experimental backpropagation based trainings of the experimental system.

A.4 Scalability and Efficiency Outlook on Optical Diffusion Models

In this section, we investigate the potential improvement in the proposed method's performance by using the same type of commercially available hardware and optimally scaling images onto the DMD and the camera. The widespread availability of newer optoelectronic technologies would enhance calculated accuracies further.

Considering fixed and passive modulation layers, the main energy and time consumption of the optical system stems from the input optical modulator, DMD and output array detector, CMOS camera. The DMD unit, Texas Instruments' DLP9500, can display 23,148 patterns per second at a resolution of 1920×1080, with an electrical consumption of 4.5 W at board level, including data transfer (i.e. looping back from the detector). When representing 8-bit images with the aggregation of 256 binary pixels as a superpixel, this device is capable of displaying 8-bit images with $24\,\mathrm{nJ/px}$. On the detection side, the CMOS camera (FLIR BFS-U3-04S2M-CS) can obtain 8-bit, 0.4 MP images at 522 fps rate with $3\,\mathrm{W}$ power consumption, acquiring images at $14\,\mathrm{nJ/px}$. If we consider 1000 timesteps with 20×20 pixels images as in the rest of this study, the consumption due to optoelectric conversion and transfer would amount to $15\,\mathrm{mJ/image}$.

Another potential energy expense item could be the light intensity required to provide a sufficient signal-to-noise ratio (SNR) at the detector. The SNR, especially in the context of a digital device like a detector or an ADC (analog-to-digital converter), is typically calculated as $SNR(dB) = 6.02 \times n + 1.76$ where n represents the number of bits. Considering the shot-noise-limited scenario, the required number of photons can be calculated using $N_{\text{input}} = \frac{SNR^2}{\eta_{\text{modulation layers}} \cdot \eta_{\text{detector}}}$ where $\eta_{\text{modulation layers}}$ accounts for a portion of light scattered out in passive layers and η_{detector} accounts for the conversion efficiency of the detector. Assuming 90% transmittance for each modulation layer and 50% efficiency for the detector, and using an 850 nm wavelength to calculate the energy of each photon: $E = \frac{hc}{\lambda}$, the required energy for light is only a few picojoules, which is negligible compared to the energy consumption of optoelectronic devices.

With the optimal configuration of the ODU, where each pixel on the analytical model is precisely mapped one-to-one to the optoelectronic devices, the same set of hardware can generate images with a consumption of approximately $15\,\mathrm{mJ/image}$ which is significantly lower than the 1.7 J required by conventional methods (see Appendix A.2.1, making the ODU more than 100 times more energy-efficient.

A.5 Training of Modulation Layers

As shown in Figure3, we utilized 3 different datasets to demonstrate the proposed approach.

- First 3 digits from MNIST-digits
- First 3 classes from Fashion-MNIST
- 20000 "Clock" images from Quick, Draw! dataset

After being downsampled to 20×20, the images are used for 250 epochs to train the ODUs, using Adam optimizer with a learning rate of 0.006, which took $\sim$ 10 hours on an A100 GPU.

A.6 Definition of Performance Metrics

Mean Squared Error (MSE)

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{8}$$

where y_i are the true values and $\hat{y}_i$ are the predicted values.

Fréchet Inception Distance (FID)

$$\text{FID}(x, g) = \|\mu_x - \mu_g\|^2 + \text{Tr}(\Sigma_x + \Sigma_g - 2(\Sigma_x \Sigma_g)^{\frac{1}{2}}) \tag{9}$$

where (μ_x, Σ_x) and (μ_g, Σ_g) are the mean and covariance of the feature vectors of the real and generated data, respectively.

Kernel Inception Distance (KID)

$$\text{KID}(x, g) = \frac{1}{n(n-1)} \sum_{i \neq j} k(f(x_i), f(x_j)) + \frac{1}{m(m-1)} \sum_{i \neq j} k(f(g_i), f(g_j)) - \frac{2}{nm} \sum_{i,j} k(f(x_i), f(g_j)) \tag{10}$$

where k is a polynomial kernel and f is the Inception network function that extracts features.

Inception Score

$$\mathrm{IS}(G) = \exp\left(\mathbb{E}_{\mathbf{x}\sim p_g}\left[D_{KL}(p(y|\mathbf{x})\|p(y))\right]\right) \tag{11}$$

where $p(y|\mathbf{x})$ is the conditional label distribution given generated image $\mathbf{x}$ and $p(y)$ is the marginal distribution over all generated images.

A.7 Code Availability

The source code is available at `https://ioguz.github.io/opticaldiffusion/`.

Appendix References

[1] M. Feit and J. Fleck, "Beam nonparaxiality, filament formation, and beam breakup in the self-focusing of optical beams," *JOSA B*, vol. 5, no. 3, pp. 633–640, 1988.

[2] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," Oct. 2017. [Online]. Available: https://openreview.net/forum?id=BJJsrmfCZ

[3] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer, 2015, pp. 234–241.

[4] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, "Stargan v2: Diverse image synthesis for multiple domains," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8188–8197.

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: See Section 3

   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: See Section 4.2

   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]

https://doi.org/10.52202/079017-1887

Justification: There is no new theoretical formulation.

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: See Appendix A.3.

   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?

https://doi.org/10.52202/079017-1887

Answer: [Yes]

Justification: The access link is provided in Appendix A.7.

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, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See Appendix A.5.

   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: See Figure 5.

   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.

https://doi.org/10.52202/079017-1887

- 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: See Appendix A.5.

   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:

   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: [Yes]

    Justification:

    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

https://doi.org/10.52202/079017-1887

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: There is no release of new data or model.

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:

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?

https://doi.org/10.52202/079017-1887

Answer: [NA]

Justification:

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:

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:

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.

https://doi.org/10.52202/079017-1887