

---

# Conditional Diffusion Models for Uncertainty Estimation in Super Resolution Microscopy

---

Anonymous Author(s)

Affiliation

Address

email

## Abstract

1 Deep learning has recently attracted considerable attention from researchers in  
2 the natural sciences, particularly microscopists, for fast extraction of physically  
3 relevant information from images. However, simple and interpretable uncertainty  
4 quantification is lacking in these applications, and remains a necessary modeling  
5 component in high-risk research. In order to quantify uncertainty in otherwise  
6 deterministic image translation architectures, we propose a hybrid generative  
7 modeling framework based on denoising diffusion probabilistic models (DDPMs).  
8 Specifically, our model combines a deterministic neural network with a DDPM,  
9 which can improve conditional synthesis speed and fidelity of the DDPM, while  
10 providing a natural mechanism for uncertainty estimation via Langevin dynamics.  
11 We apply our model to the task of single molecule localization in fluorescence  
12 microscopy, and demonstrate that blending the DeepSTORM architecture with  
13 a DDPM permits simultaneous high-fidelity super-resolution with uncertainty  
14 estimation of kernel density estimates (KDEs) regressed by DeepSTORM. Our  
15 results suggest the proposed solution is an interesting addition to the modeling  
16 toolkit for fluorescence microscopists and the field of deep image translation in  
17 general.

## 18 1 Introduction

19 Deep learning has attracted tremendous attention from researchers in the natural sciences, with  
20 several foundational applications arising in microscopy, e.g., (Weigert 2018; Falk 2019). Recently,  
21 the application of deep image translation in single-molecule localization microscopy (SMLM) has  
22 received considerable interest (Ouyang 2018; Nehme 2020; Speiser 2021). SMLM techniques  
23 are a mainstay of fluorescence microscopy and can be used to produce a pointillist representation  
24 of biomolecules in the cell at diffraction-unlimited precision (Rust 2006; Betzig 2006). As this  
25 technology enables increasingly precise measurements of the cellular environment, there is an  
26 increasing need for machine learning methods to report uncertainty for quality control.

27 In previous applications of deep models to localization microscopy, super-resolution images can be  
28 recovered from a sparse set of localizations with conditional generative adversarial networks (Ouyang  
29 2018) or kernel density estimation can be performed using convolutional networks (Nehme 2020;  
30 Speiser 2021). Here, we focus on the latter class of models which perform single molecule localization  
31 using neural networks. In this approach, one estimates molecular coordinates by predicting kernel  
32 density estimates (KDEs)  $y$ , which are latent in the raw data  $x$ , using a convolutional neural network.  
33 Importantly, inferences in SMLM are often necessarily made on a single measurement, thus common  
34 measures of model performance are based on localization errors computed over ensembles of  
35 simulated images. However, this choice precludes computation of aleatoric uncertainty at test time  
36 under a fixed model, and may result in the application of models to out of distribution datasets.



Figure 1: Generative model of single molecule localization microscopy images

Bayesian probability theory offers us mathematically grounded tools to reason about model uncertainty, but these usually come with a prohibitive computational cost (Gal 2022). A few approaches to avoiding this intractability in deep models have been deterministic uncertainty quantification (Amersfoort 2020), ensembling (Lakshminarayanan et al., 2017) or Monte Carlo dropout (Gal and Ghahramani, 2016). Here, we report a method which models estimates uncertainty in KDE predictions by combining deterministic deep learning with deep generative modeling in a hybrid algorithm. Our approach produces pixel-wise uncertainties in model predictions with no modification to the existing architecture, and can be used for downstream filtering of erroneous image regions. We choose to model a distribution on high-resolution KDE predictions conditioned on a low-resolution input using a denoising diffusion probabilistic model (DDPM) (Ho 2020; Song 2021), referred to here as simply “diffusion model”. Such models are one class of *score based generative models* which implicitly compute the score of the data distribution at each noise scale (Song 2021) and are well suited conditional image generation tasks (Saharia 2021). Most importantly, score-based models provide a natural mechanism for uncertainty quantification. Our approach could be readily integrated with existing localization performance measures to address both model accuracy on training data and precision on datasets produced by experiments.

## 2 Background

### 2.1 Image Likelihood and Localization Error

The central objective of single molecule localization microscopy is to infer a set of molecular coordinates  $\theta$  from measured low resolution images  $\mathbf{x}$ . The likelihood on a particular pixel  $k$ , i.e.,  $p(\mathbf{x}_k|\theta)$  is taken to be a convolution of Poisson and Gaussian distributions, due to shot noise  $p(s_k) = \text{Poisson}(\omega_k)$  and sensor readout noise  $p(\zeta_k) = \mathcal{N}(o_k, \sigma_k^2)$

$$p(\mathbf{x}_k|\theta) = A \sum_{q=0}^{\infty} \frac{1}{q!} e^{-\omega_k} \omega_k^q \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{(\mathbf{x}_k - g_k q - o_k)^2}{2\sigma_k^2}} \approx \text{Poisson}(\omega'_k) \quad (1)$$

where  $A$  is some normalization constant and  $\omega'_k = \omega_k + \sigma_k^2$ . For the sake of generality, we include a per-pixel gain factor  $g_k$ , which is often unity. In practice, the summation in (1) can be difficult to work with, and it is common to instead use a Poisson-Normal approximation for simplification, valid under a range of experimental conditions (Huang 2013). This result can be seen from the fact the the convolution of two Poisson distributions is also Poisson. The expectation of the Poisson process at each pixel of the image is computed from the optical transfer function  $O(u, v)$ , which is often a two-dimensional isotropic Gaussian.

$$\omega = i_0 \int O(u) du \int O(v) dv \quad (2)$$

The above integration can be carried out by computing differences of error functions, as detailed in Appendix A. The complete generative process is depicted in Figure 1.

Reliable estimation of  $\theta$  from  $\mathbf{x}$ , for example by maximum likelihood estimation or with a deep model, requires performance metrics for model selection. We use the Fisher information as an information theoretic criteria to assess the quality of the model tested here, with respect to the root mean squared error (RMSE) of our predictions of  $\theta$  (Chao 2016). The Poisson log-likelihood  $\ell(\mathbf{x}|\theta)$  is also convenient for computing the Fisher information matrix (Smith 2010) and thus the

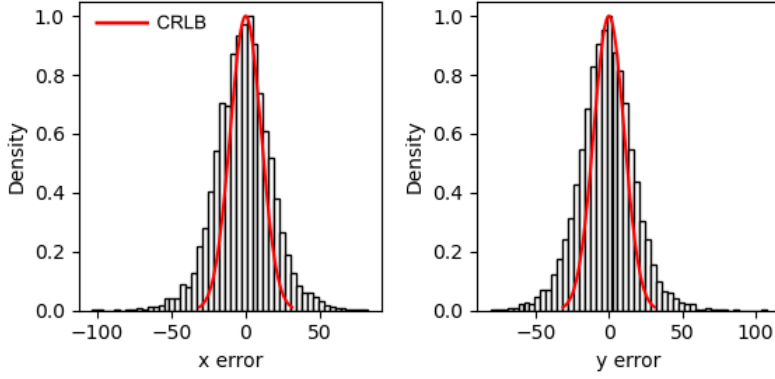


Figure 2: Localization errors of the trained model

73 Cramer-Rao lower bound, which bounds the variance of a statistical estimator of  $\theta$ , from below  
 74 i.e.,  $\text{var}(\hat{\theta}) \geq I^{-1}(\theta)$ . The Fisher information is straightforward to compute under the Poisson  
 75 log-likelihood, which is detailed in the Appendix

$$\mathcal{I}_{ij}(\theta) = \mathbb{E}_{\theta} \left( \frac{\partial \ell}{\partial \theta_i} \frac{\partial \ell}{\partial \theta_j} \right) = \sum_k \frac{1}{\omega'_k} \frac{\partial \omega'_k}{\partial \theta_i} \frac{\partial \omega'_k}{\partial \theta_j} \quad (3)$$

## 76 2.2 Kernel density estimation with deep networks

77 Direct optimization of the likelihood in (1) from observations  $\mathbf{x}$  alone is challenging when fluorescent  
 78 emitters are dense within the field of view and fluorescent signals significantly overlap. However, con-  
 79 volutional neural networks (CNN) have recently proven to be powerful tools fluorescence microscopy  
 80 to extract parameters describing fluorescent emitters such as color, emitter orientation,  $z$ -coordinate,  
 81 and background signal (Zhang 2018; Kim 2019; Zelger 2018). For localization tasks, CNNs typically  
 82 employ upsampling layers to reconstruct Bernoulli probabilities of emitter occupancy (Speiser 2021)  
 83 or kernel density estimates with higher resolution than experimental measurements (Nehme 2020).  
 84 Kernel density estimates, denoted by  $\mathbf{y}$ , are the most common data structure used in SMLM, and can  
 85 be easily generated from molecular coordinates, alongside observations  $\mathbf{x}$ , using well-understood  
 86 models of the optical impulse response (Zhang 2007).

## 87 3 Conditional Diffusion for Uncertainty-Aware Super Resolution

88 We consider datasets  $(\mathbf{x}_i, \mathbf{y}_i, \hat{\mathbf{y}}_i)_{i=1}^N$  of observed images  $\mathbf{x}_i$  true kernel density estimate (KDE) images  
 89  $\mathbf{y}_i$ , and KDE estimates  $\hat{\mathbf{y}}_i = \phi(\mathbf{x}_i)$ . Observations  $\mathbf{x}_i$  are simulated under the Poisson likelihood (1)  
 90 and KDEs are generated using (2) alone, followed by appropriate normalization.

### 91 3.1 Problem Statement

92 Point estimates  $\hat{\mathbf{y}}_i$  produced by the traditional deep architectures for super resolution microscopy  
 93 produce strong results, but lack uncertainty quantification. Recent advances in generative modeling,  
 94 particularly DDPMs, therefore present a unique opportunity to integrate uncertainty awareness into the  
 95 super-resolution microscopy toolkit. However, sampling from DDPMs is computationally expensive,  
 96 given that generation amounts to solving a complex stochastic differential equation, effectively  
 97 mapping a simple base distribution to the complex data distribution. The solution of such equations  
 98 requires numerical integration with very small step sizes, resulting in thousands of neural network  
 99 evaluations (Saharia 2021; Vahdat 2021). Furthermore, for conditional generation tasks in high-risk  
 100 applications, generation complexity is further exacerbated by the need for the highest level of detail  
 101 in generated samples.

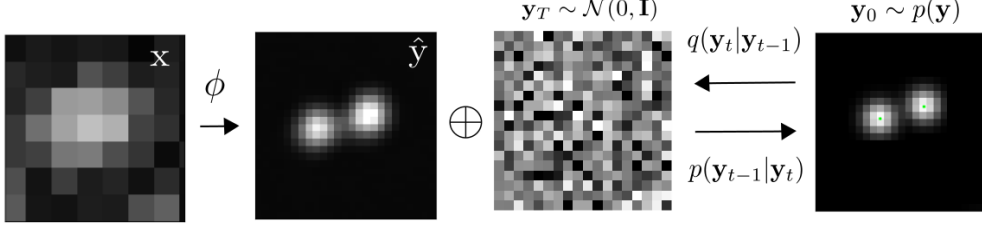


Figure 3: Conditional diffusion model for sampling kernel density estimates

Since we are primarily interested in the conditional distribution  $p(\mathbf{y}|\mathbf{x})$ , we propose that DDPM sampling is preceded by a deterministic transformation  $\phi$ , trained to predict  $\mathbf{y}$  from  $\mathbf{x}$ . Reasoning for this choice in the current application is two-fold:

**Synthesis Speed.** By training a preprocessor  $\phi$  to obtain an approximate estimate of  $\mathbf{y}$ , we can reduce the number of iterations, since the DDPM only needs to model the remaining mismatch, resulting in a less complex model from which sampling becomes easier. Speed is critical in SMLM applications, which can produce large volumes of image data in a single experiment. Moreover, we note that this approach is analogous to preconditioned stochastic gradient langevin dynamics (Li 2016), wherein  $\phi$  identifies the posterior mode followed by Langevin dynamics to sample from the posterior.

**Sample Fidelity.** Since Langevin dynamics will often be initialized in low-density regions of the data distribution, inaccurate score estimation in these regions will negatively affect the sampling process (Song 2019). Moreover, mixing can be difficult because of the need of traversing low density regions to transition between modes of the distribution. Preprocessing with a deterministic mapping  $\phi$  can ameliorate this issue, by eliminating the need for score estimation in low density regions.

The preprocessor  $\phi$  is realized by a CNN with upsampling layers. Consider the Markov chain wherein the KDE  $\mathbf{y}$  is latent in and inferred from a noisy measurement  $\mathbf{x}$ , i.e.,  $\mathbf{x} \rightarrow \phi(\mathbf{x}) \rightarrow \hat{\mathbf{y}}$ . By the data processing inequality the function  $\phi$  can only destroy information in  $\mathbf{x}$  pertaining to  $\mathbf{y}$  i.e.,  $I(\mathbf{x}; \mathbf{y}) \geq I(\phi(\mathbf{x}); \mathbf{y})$  or  $h(\mathbf{y}|\phi(\mathbf{x})) \geq h(\mathbf{y}|\mathbf{x})$  where  $I$  is the mutual information and  $h$  is the entropy. In other words, the function  $\phi$ , while deterministic, can introduce additional uncertainty about  $\mathbf{y}$  in downstream stochastic models by destroying information. Here, we are interested in measuring the upper bound  $h(\mathbf{y}|\phi(\mathbf{x}))$ , as this is the relevant quantity when a deterministic transformation  $\phi$  is an unavoidable first step.

In practice, a DDPM  $\Psi$  can be trained on pairs  $(\mathbf{y}_i, \hat{\mathbf{y}}_i)_{i=1}^N$ . The conditional DDPM generates a target KDE  $\mathbf{y}_0$  in  $T$  refinement steps. Starting with a pure noise image  $\mathbf{y}_T \sim \mathcal{N}(0, \mathbf{I})$ , the model iteratively refines the KDE through successive iterations according to learned conditional transition distributions  $p(\mathbf{y}_{t-1}|\mathbf{y}_t)$  such that  $\mathbf{y}_0 \sim p(\mathbf{y}|\hat{\mathbf{y}})$

### 3.2 Gaussian Diffusion

Diffusion models (Sohl-Dickstein 2015; Ho 2020) are a class of generative models inspired by nonequilibrium statistical physics, which slowly destroy structure in a data distribution  $p(\mathbf{y}_0|\mathbf{x})$  via a fixed Markov chain referred to as the *forward process*. In the present context, we apply leverage recent results from (Ho 2020; Song 2021; Saharia 2021) for applying this framework to sampling from  $p(\mathbf{y}|\mathbf{x}, \hat{\mathbf{y}})$ . We note that this approach is analogous to preconditioned stochastic gradient langevin dynamics (Li 2016), wherein  $\phi$  identifies the posterior mode followed by sampling with Langevin dynamics. The forward process gradually adds Gaussian noise to the KDE  $\mathbf{y}$  according to a variance schedule  $\beta_{0:T}$

$$q(\mathbf{y}_t|\mathbf{y}_0) = \prod_{t=1}^T q(\mathbf{y}_t|\mathbf{y}_{t-1}) \quad q(\mathbf{y}_t|\mathbf{y}_{t-1}) = \mathcal{N}\left(\sqrt{1-\beta_t}\mathbf{y}_{t-1}, \beta_t\mathbf{I}\right) \quad (4)$$

The usual procedure is then to learn a parametric representation of the *reverse process*, and therefore generate samples from  $p(\mathbf{y}_0)$ , starting from noise. Formally,  $p_\theta(\mathbf{y}_0|\hat{\mathbf{y}}) = \int p_\theta(\mathbf{y}_{0:T}|\hat{\mathbf{y}})d\hat{\mathbf{y}}_{1:T}$  where

140  $\mathbf{y}_t$  is a latent representation with the same dimensionality of the data.  $p_\theta(\mathbf{y}_{0:T}|\hat{\mathbf{y}})$  is a Markov process,  
 141 starting from a noise sample  $p_\theta(\mathbf{y}_T) = \mathcal{N}(0, \mathbf{I})$ .

$$p_\theta(\mathbf{y}_{0:T}) = p_\theta(\mathbf{y}_T) \prod_{t=1}^T p_\theta(\mathbf{y}_{t-1}|\mathbf{y}_t) \quad p_\theta(\mathbf{y}_{t-1}|\mathbf{y}_t) = \mathcal{N}(s_\theta(\mathbf{y}_t), \beta_t \mathbf{I}) \quad (5)$$

142 where we reuse the variance schedule of the forward process (Ho 2020). We omit conditioning  
 143 on  $\hat{\mathbf{y}}$  for each transition density  $p_\theta(\mathbf{y}_{t-1}|\mathbf{y}_t)$ , as this is only considered at  $t = 0$  i.e.,  $p_\theta(\mathbf{y}_1|\mathbf{y}_0, \hat{\mathbf{y}})$ .  
 144 An important property of the forward process is that it admits sampling  $\mathbf{y}_t$  at an arbitrary timestep  
 145  $t$  in closed form (Ho 2020). Using the notation  $\alpha_t := 1 - \beta_t$  and  $\gamma_t := \prod_{s=1}^t \alpha_s$ , we have  
 146  $q(\mathbf{y}_t|\mathbf{y}_0) = \mathcal{N}(\sqrt{\gamma_t}\mathbf{y}_0, (1 - \gamma_t)\mathbf{I})$ . Training is performed by optimizing the usual variational bound  
 147 on negative log likelihood:

$$\mathcal{L}(\theta) = \mathbb{E}[-\log p_\theta(\mathbf{y}_0|\mathbf{x})] \leq \mathbb{E}\left[-\log \frac{p_\theta(\mathbf{y}_{0:T}|\mathbf{x})}{q(\mathbf{y}_{1:T}|\mathbf{y}_0)}\right] \quad (6)$$

148 The objective in (6) can be expanded in terms of  $D_{\text{KL}}(p(\mathbf{y}_{t-1}|\mathbf{y}_t)||q(\mathbf{y}_t|\mathbf{y}_{t-1}))$  as detailed in (Ho  
 149 2020). We choose to adopt the simplified form of the variational bound, which emphasizes that the  
 150 DDPM estimates the score  $\nabla_{\mathbf{y}} \log p(\mathbf{y}|\mathbf{x})$  at each noise level (Song 2021)

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \mathbb{E}_{(\hat{\mathbf{y}}, \mathbf{y}_0)(\epsilon, \gamma)} \left[ s_\theta\left(x, \sqrt{\gamma}\mathbf{y}_0 + \sqrt{1 - \gamma}\epsilon \mid \mathbf{y}_t, \gamma\right) - \epsilon \right], \quad (7)$$

151 After training, samples can be generated by

$$\mathbf{y}_{t-1} = \frac{1}{\sqrt{1 - \beta_t}} (\mathbf{y}_t + \beta_t s_\theta(\mathbf{y}_t)) + \sqrt{\beta_t} \xi \quad (8)$$

152 The distribution of  $\gamma$  has a big impact on the quality of the model and the generated outputs. For our  
 153 training noise schedule, we use a piecewise distribution for  $\gamma$ ,  $p(\gamma) = \frac{1}{T} \sum_{t=1}^T U(\gamma_{t-1}, \gamma_t)$  (Nanxin  
 154 2021). Specifically, during training, we first uniformly sample a time step  $t \sim \{0, \dots, T\}$  followed by  
 155 sampling  $\gamma \sim U(\gamma_{t-1}, \gamma_t)$ . We set  $T = 100$  in all our experiments.

### 156 3.3 Optimization of the DeepSTORM architecture

157 A first pass at localization treats localization as a binary classification problem, such that 0 denotes  
 158 a vacant pixel and 1 denotes an occupied pixel containing an emitter. Direct learning of pixel-wise  
 159 classification with cross-entropy loss leads to an imbalance of occupied and unoccupied pixels in  
 160 dense localization problems (Nehme 2020). CE loss is usually either weighted [51], replaced with a  
 161 Focal loss [52], or applied to a "blobbed" version of the desired boolean volume e.g. by placing a disk  
 162 around each GT position [53–55]. Alternative methods take a soft version of the binary classification  
 163 problem. That is, by placing a small Gaussian around each GT position (e.g. with std of 1 pixel),  
 164 and matching continuous heatmaps, backpropagation yields more meaningful gradients and eases the  
 165 learning process convergence.

166 Localization heatmaps thus form a natural encoding for SMLM images, which can be input to our  
 167 conditional diffusion model. Therefore, to encode raw data  $\mathbf{x}$  into a more tractable representation, we  
 168 train the DeepSTORM architecture (Nehme 2020). Raw coordinates  $\theta$  are binned into an upsampled  
 169 image  $\mathbf{z}$ .

## 170 4 Experiments

171 All training data was simulated under the likelihood and impulse response (2,10), drawing coordinates  
 172 uniformly over a disc with a radius of 7 pixels.

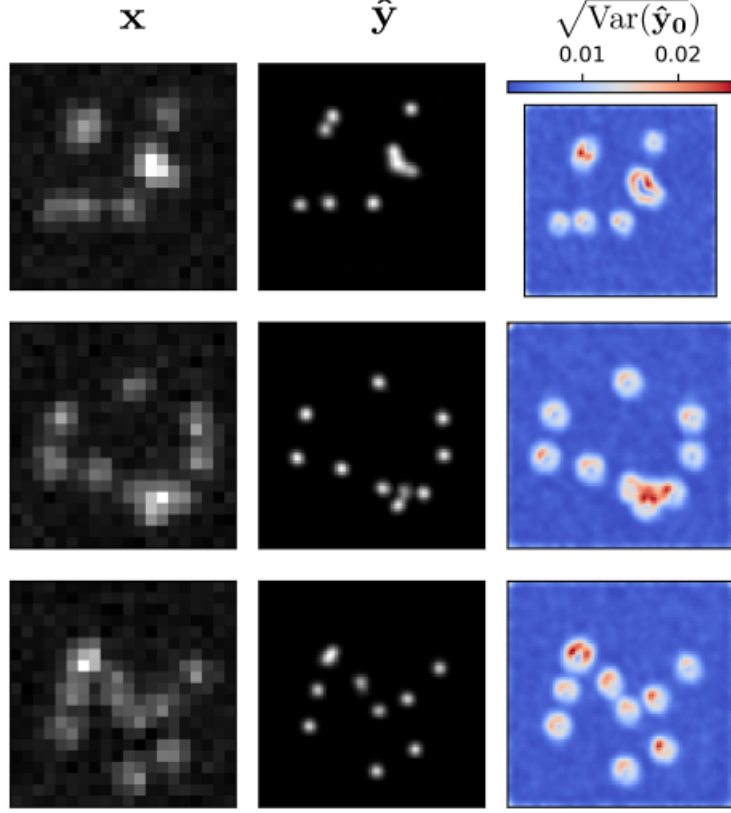


Figure 4: Kernel density estimates for various signal to noise ratios (SNR)

#### 173 4.1 Model Precision on Simulated Ensembles

#### 174 4.2 Model Uncertainty

175 We set  $T = 100$  for all experiments and treat forward process variances  $\beta_t$  as hyperparameters,  
 176 with a linear schedule from  $\beta_0 = 10^{-4}$  to  $\beta_T = 10^{-2}$ . These constants were chosen to be small  
 177 relative to data scaled to  $[-1, 1]$ , ensuring that reverse and forward processes have approximately  
 178 the same functional form while keeping the signal-to-noise ratio at  $x_T$  as small as possible ( $L_T =$   
 179  $D_{KL}(q(x_T|x_0)||\mathcal{N}(0, I)) \approx 10^{-5}$  bits per dimension in our experiments).

180 To represent the reverse process, we used the DDPM architecture based on a U-Net backbone (Ho  
 181 2020). Parameters are shared across time, which is specified to the network using the Transformer  
 182 sinusoidal position embedding ?. We use self-attention at the  $16 \times 16$  feature map resolution ??.  
 183 Details are in Appendix A.

184 and the channel multipliers at different resolutions (see Appendix A for details). To condition the  
 185 model on the input  $x$ , we up-sample the low-resolution image to the target resolution using bicubic  
 186 interpolation. The result is concatenated with  $y_t$  along the channel dimension. We experimented with  
 187 more sophisticated methods of conditioning, such as using, but we found that the simple concatenation  
 188 yielded similar generation quality.

## 189 5 Related Work

### 190 5.1 Diffusion Models

191 Prior work of diffusion models ?? require 1-2k diffusion steps during inference, making generation  
 192 slow for large target resolution tasks. We adapt techniques from ? to enable more efficient inference.  
 193 Our model conditions on  $\gamma$  directly (vs  $t$  as in ?), which allows us flexibility in choosing the number

of diffusion steps, and the noise schedule during inference. This has been demonstrated to work well for speech synthesis ?, but has not been explored for images. For efficient inference, we set the maximum inference budget to 100 diffusion steps, and hyper-parameter search over the inference noise schedule. This search is inexpensive as we only need to train the model once ?. We use FID on held-out data to choose the best noise schedule, as we found PSNR did not correlate well with image quality.

## 6 Conclusion

## References

- [1] Nehme, E., et al. *DeepSTORM3D: dense 3D localization microscopy and PSF design by deep learning*. Nature Methods 17, 734–740 (2020).
- [2] Ouyang, W., et al. *Deep learning massively accelerates super-resolution localization microscopy*. Nature Biotechnology 36, 460–468 (2018).
- [3] Speiser, A., et al. *Deep learning enables fast and dense single-molecule localization with high accuracy*. Nature Methods 18, 1082–1090 (2021).
- [4] Sohl-Dickstein J., et al. *Deep unsupervised learning using nonequilibrium thermodynamics*. ICLR (2015).
- [5] Ho J., et al. *Denoising Diffusion Probabilistic Models*. Advances in Neural Information Processing Systems (2015).
- [6] Nanxin C., et al. *WaveGrad: Estimating Gradients for Waveform Generation*. ICLR (2021).
- [4] Chao, J., et al. *Fisher information theory for parameter estimation in single molecule microscopy: tutorial*. Journal of the Optical Society of America A 33, B36 (2016).
- [5] Schermelleh, L. et al. *Super-resolution microscopy demystified*. Nature Cell Biology vol. 21 72–84 (2019).
- [6] Zhang, B., et al. *Gaussian approximations of fluorescence microscope point-spread function models*. (2007).
- [7] Smith, C.S., *Fast, single-molecule localization that achieves theoretically minimum uncertainty*. Nature Methods 7, 373–375 (2010).
- [8] Nieuwenhuizen, R., et al. *Measuring image resolution in optical nanoscopy*. Nature Methods 10, 557–562 (2013).
- [9] Huang, F., et al. *Video-rate nanoscopy using sCMOS camera-specific single-molecule localization algorithms*. Nat Methods 10, 653–658 (2013).
- [10] Rust, M., et al. *Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM)*. Nat Methods 3, 793–796 (2006).
- [11] Betzig, E., et al. *Imaging intracellular fluorescent proteins at nanometer resolution*. Science 313, 1642–1645 (2006).
- [12] Weigert, M., et al. *Content-aware image restoration: pushing the limits of fluorescence microscopy*. Nat. Methods 15, 1090 (2018).
- [13] Falk, T., et al. *U-net: deep learning for cell counting, detection, and morphometry*. Nat. Methods 16, 67–70 (2019).
- [14] Boyd, N., et al. *DeepLoco: fast 3D localization microscopy using neural networks*. Preprint at bioRxiv <https://doi.org/10.1101/267096> (2018)
- [15] Zelger, P., et al. *Three-dimensional localization microscopy using deep learning*. Opt. Express 26, 33166–33179 (2018)
- [16] Zhang, P., et al. *Analyzing complex single-molecule emission patterns with deep learning*. Nat. Methods 15, 913 (2018)
- [17] Saharia, C., et al. *Image Super-Resolution via Iterative Refinement*. Preprint at arXiv <https://doi.org/10.48550/arXiv.2104.07636> (2021)
- [18] Kim, T., et al. *Information-rich localization microscopy through machine learning*. Nat Commun 10, 1996 (2019).

## A Appendix

Standard SMLM localization algorithms based on maximum likelihood estimators or least squares optimization require tight control of activation and reactivation to maintain sparse emitters, presenting a tradeoff between imaging speed and labeling density. Recently, deep models have generalized SMLM to densely labeled structures by predicting high-resolution kernel density estimates (KDEs) from low resolution images with convolutional networks. However, estimated KDEs may contain irregularities due to finite sample sizes and limited model capacity.

The DeepSTORM CNN, initially proposed in (Nehme 2020) for 3D localization, can be viewed as a deep kernel density estimator, reconstructing kernel density estimates  $\mathbf{y}$  from low-resolution inputs  $\mathbf{x}$ . We utilize a simplified form of the original architecture for 2D localization, which we denote  $\phi$  hereafter, which consists of three main modules: a multi-scale context aggregation module, an upsampling module, and a prediction module. For context aggregation, the architecture utilizes dilated convolutions to increase the receptive field of each layer. The upsampling module is then composed of two consecutive 2x resize-convolutions, computed by nearest-neighbor interpolation, to increase the lateral resolution by a factor of 4. For a common sCMOS camera, each pixel has a lateral size of approximately 108 nanometers, giving approximately 27 nanometer pixels in the KDE. The terminal prediction module contains three additional convolutional blocks for refinement of the upsampled image, followed by an element-wise HardTanh.

Single molecule localization microscopy (SMLM) relies on the temporal resolution of fluorophores whose spatially overlapping point spread functions would otherwise render them unresolvable at the detector. Common strategies for the temporal separation of molecules involve molecular photoswitching from dark to fluorescent states, permitting resolution of fluorophores beyond the diffraction limit. Estimation of molecular coordinates is typically carried out by modeling the optical impulse response of the imaging system and fitting model functions to the data. However, such models are only well-suited to isolated molecules, reducing the number of molecules in the field of view and limiting temporal resolution in super resolution microscopy. This issue has incited a series of efforts to increase the density of fluorescent molecules imaged in a single frame while developing appropriate models for dense localization.

In fluorescence microscopy, each pixel is treated as a Poisson random variable (Smith 2010; Nehme 2020; Chao 2016), with expected value

$$\omega = i_0 \int O(u)du \int O(v)dv \quad (9)$$

where  $i_0 = \eta N_0 \Delta$ . The scalar parameters  $\eta, \Delta$  are the photon detection probability of the sensor and the exposure time, respectively. Without loss of generality, we assume  $\eta = \Delta = 1$ . Most importantly,  $N_0$  represents the signal amplitude, which we assume maintains a fixed value. The optical impulse response  $O(u, v)$  is often approximated as a 2D isotropic Gaussian with standard deviation  $\sigma$  (Zhang 2007). This approximation has the convenient property, that the effects of pixelation can be expressed in terms of error functions. For example, given a fluorescent emitter located at  $\theta = (u_0, v_0)$ , we have that

$$\int O(u)du = \frac{1}{2} \left( \text{erf} \left( \frac{u_k + \frac{1}{2} - u_0}{\sqrt{2}\sigma} \right) - \text{erf} \left( \frac{u_k - \frac{1}{2} - u_0}{\sqrt{2}\sigma} \right) \right) \quad (10)$$

where we have used the common definition  $\text{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt$ . Our generative model also incorporates a normally distributed white noise per pixel  $\zeta$  with offset  $o$  and variance  $\sigma^2$ . Ultimately, we have a Poisson component of the signal, which scales with  $N_0$  and a Gaussian component, which does not.

Consider,

$$\zeta_k - o_k + \sigma_k^2 \sim \mathcal{N}(\sigma_k^2, \sigma_k^2) \approx \text{Poisson}(\sigma_k^2) \quad (11)$$

Since  $\mathbf{x}_k = \mathbf{s}_k + \zeta_k$ , we transform  $\mathbf{x}'_k = \mathbf{x}_k - o_k + \sigma_k^2$ , which is distributed according to



283 Consider the factorization  $p(\hat{\mathbf{y}}|\mathbf{x}, \mathbf{y})p(\mathbf{x}|\mathbf{y})p(\mathbf{y}) = p(\mathbf{x}|\mathbf{y}, \hat{\mathbf{y}})p(\mathbf{y}|\hat{\mathbf{y}})p(\hat{\mathbf{y}})$ . Given that  $\mathbf{x}$  is condition-  
284 ally independent of  $\hat{\mathbf{y}}$ , we find

$$p_{\Psi}(\hat{\mathbf{y}}|\mathbf{x}, \mathbf{y}) = p(\mathbf{y}|\hat{\mathbf{y}})$$