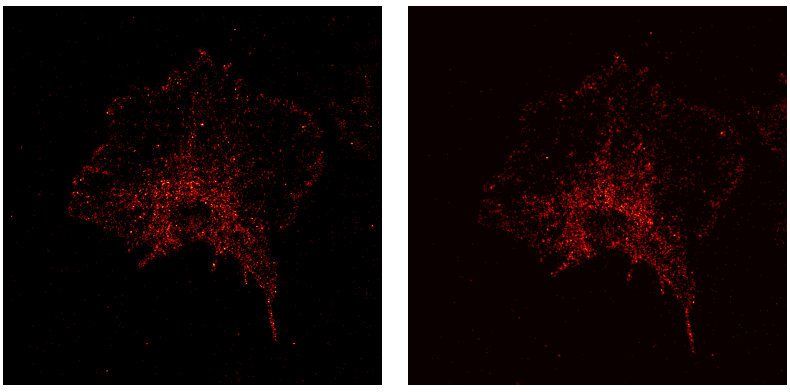
Localization Microscopy Using Deep Neural Networks

A. Bhowmik[*], A. Adiga, [*], C. S. Seelamantula[*], F. Hauser[†], J. Jacak[†], B. Heise[†]
[*] Department of Electrical Engineering, Indian Institute of Science, Bangalore, India
[†] Institute of Applied Physics, Johannes Kepler University, Linz, Austria
Corresponding author: chandrasekhar@iisc.ac.in

Every frame of a stochastic optical reconstruction microscopy (STORM) image is diffraction-limited but due to sparse excitation of fluorophores, with little overlap between point-sources across frames. Localization relies on deconvolution to suppress the blurring introduced by diffraction. The de facto standard is the Gaussian peak-fitting (GF) algorithm. We propose a Bayesian deep deconvolution neural network (BD2N2) for localization. Consider point-source excitation $\mathbf{X} \in \mathbb{R}^{M\times N}$ and a separable point-spread function (PSF) $\mathbf{H} \in \mathbb{R}^{M\times N}$ resulting in the measured signal $\mathbf{Y} \in \mathbb{R}^{M\times N}$ $\mathbf{Y} = \mathbf{H}_c\mathbf{X}\mathbf{H}_r^\top + \mathbf{W}$, where $\mathbf{H}_c$ and $\mathbf{H}_r$ are Toeplitz matrices corresponding to convolution along rows and columns, respectively. The excitation $\mathbf{X}$ is sparse and random with i.i.d. entries and distributed as $g(\mathbf{X})$. Consider the maximum a posteriori (MAP) estimate $\mathbf{X}_{\text{MAP}} = \arg\max_{\mathbf{X}} f(\mathbf{Y}/\mathbf{X};\mathbf{H})g(\mathbf{X})$, where f is the likelihood of the observations. This results in an optimization problem of the form $\mathbf{X}_{\text{MAP}} = \arg\min_{\mathbf{X}} \frac{1}{2}\left\|\mathbf{Y} - \mathbf{H}_c\mathbf{X}\mathbf{H}_r^\top\right\|_{\text{F}}^2 + \lambda\mathcal{G}(\mathbf{X})$, where $\mathcal{G}(\mathbf{X}) = \log g(\mathbf{X})$ acts as the regularizer, λ encapsulates the parameters of the distribution and F denotes the Frobenius norm. In general, the density $g(\mathbf{X})$ is unknown and we learn the regularizer by employing a deep neural network (DNN). The solution can be sought using iterative proximal methods. Considering a step along the gradient of $D(\mathbf{X}) = \left\|\mathbf{Y} - \mathbf{H}_c\mathbf{X}\mathbf{H}_r^\top\right\|_{\text{F}}^2$ and applying a proximal operator P corresponding to $\mathcal{G}$, we get the update in the $(\ell+1)^{\text{st}}$ iteration as $\mathbf{X}^{\ell+1} = \mathcal{P}\left(\mathbf{X}^\ell - \eta\nabla D\left(\mathbf{X}^\ell\right)\right)$ which by matrix identities is rewritten as $\mathbf{X}^{\ell+1} = \mathcal{P}\left(\mathbf{X}^\ell - \eta{\mathbf{H}_c}^{\text{T}}\mathbf{H}_c\mathbf{X}^\ell\mathbf{H}_r{\mathbf{H}_r}^{\text{T}} + \eta{\mathbf{H}_c}^{\text{T}}\mathbf{Y}{\mathbf{H}_r}^{\text{T}}\right)$ and can be interpreted as the feedforward computation through a Neural Network (NN) with input X^ℓ and activation $\mathcal{P}$. Unlike the standard DNN setup, the weight matrix and bias vector are fixed and one learns the activation function, which is equivalent to learning the regularizer. We parametrize the activation using a linear expansion of thresholds (LET) employing derivative of a Gaussian, and learn its weights.

We consider STORM imaging of Actin filaments using Phalloidin conjugated Alexa647 dye, where the measurements consist of 9994 low-resolution frames of size 129×129. The PSF is a Gaussian with $\sigma_{\text{PSF}} = 150$ nm. For training the BD2N2, 50 frames were randomly chosen and the remaining were used for testing. Each localized source on the 129×129 grid is rendered over a higher-resolution grid of size 1290×1290 using a gradient-descent-based method and further associated with a Gaussian uncertainty blob as determined by Thompson's rule. To begin with, the proposed BD2N2 reconstruction turned out to be on par with the benchmark Gaussian peak fitting technique (Fig. 1). However, the proposed framework has significant scope for improvements.



Gaussian peak-fitting Proposed BD2N2 reconstruction

Figure 1: (Color online) *Actin filament* images reconstructed by BD2N2 vis-à-vis the standard Gaussian fitting algorithm.