

# Spatio-angular restoration of polarized diSPIM data: analytical, numerical, and experimental results

August 31, 2018

## 1 Introduction

These notes detail the spatio-angular restoration pipeline for data collected with the polarized diSPIM. In Section 2 we explain our forward model—the relationship between an arbitrary spatio-angular density and the data we collect with the polarized diSPIM. We will focus on our discrete implementation and the computational issues that we faced. In Section 3 we will explain our inversion scheme—the set of computations that allow us to recover a spatio-angular density from noise-corrupted data. In Sections 4 and 5 we will show numerical and experimental results.

## 2 Forward model

### 2.1 Continuous-to-continuous model

The objects we are attempting to simulate and reconstruct are spatio-angular densities of fluorophores  $f(\mathbf{r}_o, \hat{\mathbf{s}}_o)$ —the number of fluorophores at each 3D position  $\mathbf{r}_o$  oriented along a direction  $\hat{\mathbf{s}}_o$ . Therefore, the complete object space  $\mathbb{U}$  is the function space  $L_2(\mathbb{R}^3 \times \mathbb{S}^2)$ . The data that we sample from each arm of the polarized diSPIM is  $g(\mathbf{r}_d, \hat{\mathbf{p}})$ —the irradiance measured at each 3D detector position  $\mathbf{r}_d$  with the illumination polarizer oriented along  $\hat{\mathbf{p}}$ . The data space for each view is a member of the function space  $L_2(\mathbb{R}^3 \times \mathbb{S}^1)$ . We have two views, so the complete data space  $\mathbb{V}$  is the function space  $[L_2(\mathbb{R}^3 \times \mathbb{S}^1)]^2$ . Although this section is named continuous-to-continuous (CC in Barrett's notation), strictly speaking this is already a CC-CD model because we have a discrete number of views.

In previous note sets we have shown that we can write the CC model for the polarized diSPIM as

$$g_v(\mathbf{r}_d, \hat{\mathbf{p}}) = \int_{\mathbb{R}^3} d\mathbf{r}_o \int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o h_v(\mathbf{r}_d - \mathbf{r}_o, \hat{\mathbf{p}}, \hat{\mathbf{s}}_o) f(\mathbf{r}_o, \hat{\mathbf{s}}_o), \quad (1)$$

where  $v \in \{A, B\}$  is the light path index, and  $h_v(\mathbf{r}_d - \mathbf{r}_o, \hat{\mathbf{p}}, \hat{\mathbf{s}}_o)$  is the kernel of the integral transform that maps objects space to data space. The imaging process consists of incoherent excitation and detection steps, so we can write the kernel as a product of an excitation and detection kernel

$$h_v(\mathbf{r}_o, \hat{\mathbf{s}}_o; \hat{\mathbf{p}}) = h_{v,\text{exc}}(\mathbf{r}_o, \hat{\mathbf{s}}_o; \hat{\mathbf{p}}) h_{v,\text{det}}(\mathbf{r}_o, \hat{\mathbf{s}}_o). \quad (2)$$

Notice that we are considering polarized illumination, so the polarizer coordinate  $\hat{\mathbf{p}}$  is a variable in the excitation kernel only.

The excitation kernels for each view are given by

$$h_{B,\text{exc}}(\mathbf{r}_o, \hat{\mathbf{s}}_o; \hat{\mathbf{p}}) = \left[ z_0(\hat{\mathbf{p}}) y_0^0(\hat{\mathbf{s}}_o) + \sqrt{\frac{3}{5}} z_{-2}(\hat{\mathbf{p}}) y_2^{-2}(\hat{\mathbf{s}}_o) - \frac{1}{\sqrt{5}} z_0(\hat{\mathbf{p}}) y_2^0(\hat{\mathbf{s}}_o) + \sqrt{\frac{3}{5}} z_2(\hat{\mathbf{p}}) y_2^2(\hat{\mathbf{s}}_o) \right] \mathfrak{g}(r_x, \sigma_{ls}), \quad (3)$$

$$h_{A,\text{exc}}(\mathbf{r}_o, \hat{\mathbf{s}}_o; \hat{\mathbf{p}}) = \mathcal{R}_{xz} \{ h_{B,\text{exc}}(\mathbf{r}_o, \hat{\mathbf{s}}_o; \hat{\mathbf{p}}) \}, \quad (4)$$

where  $z_n(\hat{\mathbf{p}})$  are circular harmonic functions;  $y_l^m(\hat{\mathbf{s}}_o)$  are spherical harmonic functions;  $\mathfrak{g}(\mu, \sigma)$  is a Gaussian function with mean  $\mu$  and standard deviation  $\sigma$ ; and  $\mathcal{R}_{xz}$  is a rotation operator that swaps the  $x$  and  $z$  coordinates within the function. Note that the operator  $\mathcal{R}_{xz}$  applies a rotation transformation to the circular and spherical harmonics as well.

The detection kernels are given by

$$h_{A,\text{det}}(\mathbf{r}_o, \hat{\mathbf{s}}_o) = \left\{ \left[ a_{1,A}(r_{xy}) + \frac{\alpha_A^2}{4} a_{2,A}(r_{xy}) \right] y_0^0(\hat{\mathbf{s}}_o) + \frac{1}{\sqrt{5}} \left[ -a_{1,A}(r_{xy}) + \frac{\alpha_A^2}{2} a_{2,A}(r_{xy}) \right] y_2^0(\hat{\mathbf{s}}_o) \right\} \mathbf{g}(r_z, \sigma_{\text{det}}), \quad (5)$$

$$h_{B,\text{det}}(\mathbf{r}_o, \hat{\mathbf{s}}_o) = \mathcal{R}_{xz} \{ h_{A,\text{det}}(\mathbf{r}_o, \hat{\mathbf{s}}_o) \}. \quad (6)$$

where

$$a_{n,v}(r) \equiv \frac{n}{\pi} \left[ \frac{J_n(2\pi\nu_{o,v}r)}{2\pi\nu_{o,v}r} \right], \quad (7)$$

and

$$r_{xy} \equiv \sqrt{r_x^2 + r_y^2}, \quad \nu_{o,v} \equiv \frac{\text{NA}_v}{\lambda}, \quad \alpha_v \equiv \frac{\text{NA}_v}{n_0}. \quad (8)$$

Notice that we have added a subscript v to NA,  $\alpha$ , and  $\nu_o$  to allow for the possibility of an asymmetric diSPIM.

Eqs. 1–8 are sufficient to simulate the forward model of a polarized diSPIM, but the integrals in Eq. 1 are extremely expensive to calculate. If we naively discretize  $f(\mathbf{r}_o, \hat{\mathbf{s}}_o)$  into  $N = 1000 \times 1000 \times 1000 \times 1000 = 10^{12}$  spatio-angular points and discretize the data into  $M = 1000 \times 1000 \times 1000 \times 4$  polarizers  $\times 2$  views  $= 8 \times 10^9$  data points (a realistic data set), then simulating the forward model (assuming the kernel is inexpensive to evaluate) would require  $\mathcal{O}(NM) \approx 8 \times 10^{21}$  floating-point operations (FLOPs)—not realistic on a typical  $10^{11}$  floating-point operations per second (FLOPS) machine (a very good laptop). Aside: there’s a cute mnemonic for converting large numbers of seconds— $\pi$  seconds is approximately one nanocentury or  $\pi \times 10^7$  s  $\approx 1$  y. Luckily we can evaluate the forward model much more quickly by exploiting the symmetries and rank of the integral operator.

The first symmetry we’ll exploit is shift invariance. We notice that the spatial integral is in the form of a convolution, so we apply the convolution theorem and rewrite Eq. 1 as

$$g_v(\mathbf{r}_d, \hat{\mathbf{p}}) = \mathcal{F}_{\mathbb{R}^3}^{-1} \left\{ \int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o \mathcal{F}_{\mathbb{R}^3} \{ h_v(\mathbf{r}_o, \hat{\mathbf{p}}, \hat{\mathbf{s}}_o) \} \mathcal{F}_{\mathbb{R}^3} \{ f(\mathbf{r}_o, \hat{\mathbf{s}}_o) \} \right\}, \quad (9)$$

where  $\mathcal{F}_{\mathbb{R}^3}$  denotes a three-dimensional Fourier transform given by

$$\mathcal{F}_{\mathbb{R}^3} \{ f_v(\mathbf{r}_o, \hat{\mathbf{s}}_o) \} \equiv \int_{\mathbb{R}^3} d\mathbf{r}_o e^{i2\pi\mathbf{r}_o \cdot \boldsymbol{\nu}} f_v(\mathbf{r}_o, \hat{\mathbf{s}}_o). \quad (10)$$

Eq. 9 is already much easier to compute than Eq. 1, but it still requires us to evaluate a spherical integral with two 3D Fourier transforms in the integrand. We can simplify this integral by exploiting the angular rank of the kernel. First, we apply the following identity

$$\int_{\mathbb{S}^2} d\hat{\mathbf{s}} f(\hat{\mathbf{s}}) g(\hat{\mathbf{s}}) = \sum_{l=0}^{\infty} \sum_{m=-l}^l \mathcal{F}_{\mathbb{S}^2} \{ f(\hat{\mathbf{s}}) \} \mathcal{F}_{\mathbb{S}^2} \{ g(\hat{\mathbf{s}}) \}, \quad (11)$$

where

$$\mathcal{F}_{\mathbb{S}^2} \{ f(\hat{\mathbf{s}}) \} \equiv \int_{\mathbb{S}^2} f(\hat{\mathbf{s}}) y_l^m(\hat{\mathbf{s}}). \quad (12)$$

Eq. 11 is the generalized Plancharel theorem for spherical functions. It is a special case of the fact that scalar products are invariant under unitary transformations (see Barrett 3.78). Next, we plug Eq. 11 into Eq. 9 to get

$$g_v(\mathbf{r}_d, \hat{\mathbf{p}}) = \mathcal{F}_{\mathbb{R}^3}^{-1} \left\{ \sum_{l=0}^{\infty} \sum_{m=-l}^l H_{l,v}^m(\boldsymbol{\nu}, \hat{\mathbf{p}}) \mathcal{F}_{\mathbb{R}^3 \times \mathbb{S}^2} \{ f(\mathbf{r}_o, \hat{\mathbf{s}}_o) \} \right\} \quad (13)$$

where

$$H_{l,v}^m(\boldsymbol{\nu}, \hat{\mathbf{p}}) \equiv \mathcal{F}_{\mathbb{R}^3 \times \mathbb{S}^2} \{ h_v(\mathbf{r}_o, \hat{\mathbf{p}}, \hat{\mathbf{s}}_o) \} \equiv \int_{\mathbb{R}^3} d\mathbf{r}_o e^{i2\pi \mathbf{r}_o \cdot \boldsymbol{\nu}} \int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o y_l^m(\hat{\mathbf{s}}_o) h_v(\mathbf{r}_o, \hat{\mathbf{p}}, \hat{\mathbf{s}}_o). \quad (14)$$

Eq. 13 is the main result of this section. Right now it is not obvious that Eq. 13 is any more efficient than Eq. 9, but when we calculate the transfer function we will see that the sum over spherical harmonics contains a small number of terms.

We compute the transfer function  $H_{l,v}^m(\boldsymbol{\nu}, \hat{\mathbf{p}})$  by plugging Eq. 2 into Eq. 14

$$H_{l,v}^m(\boldsymbol{\nu}, \hat{\mathbf{p}}) = \mathcal{F}_{\mathbb{R}^3 \times \mathbb{S}^2} \{ h_{v,\text{exc}}(\mathbf{r}_o, \hat{\mathbf{s}}_o; \hat{\mathbf{p}}) h_{v,\text{det}}(\mathbf{r}_o, \hat{\mathbf{s}}_o) \}. \quad (15)$$

We can factor the kernel for both views into a transverse and axial part (a Gaussian) so

$$H_{l,v}^m(\boldsymbol{\nu}, \hat{\mathbf{p}}) = \mathcal{F}_{\mathbb{R}^3 \times \mathbb{S}^2} \{ h_{v,\text{exc}}^{\text{tr}}(\hat{\mathbf{s}}_o; \hat{\mathbf{p}}) h_{v,\text{det}}^{\text{tr}}(\mathbf{r}_o, \hat{\mathbf{s}}_o) g(0, \sigma_{\text{ax}}) \}. \quad (16)$$

Rearranging the Fourier transforms gives

$$H_{l,v}^m(\boldsymbol{\nu}, \hat{\mathbf{p}}) = \mathcal{F}_{\mathbb{S}^2} \{ h_{v,\text{exc}}^{\text{tr}}(\hat{\mathbf{s}}_o; \hat{\mathbf{p}}) \mathcal{F}_{\mathbb{R}^2} \{ h_{v,\text{det}}^{\text{tr}}(\mathbf{r}_o, \hat{\mathbf{s}}_o) \} \mathcal{F}_{\mathbb{R}^1} \{ g(0, \sigma_{\text{ax}}) \} \}. \quad (17)$$

The Fourier transform of a Gaussian is a Gaussian with inverse standard deviation, and in previous notes we evaluated the Fourier transform of the transverse detection kernel as

$$\mathcal{F}_{\mathbb{R}^2} \{ h_{A,\text{det}}^{\text{tr}}(\mathbf{r}_o, \hat{\mathbf{s}}_o) \} = \left[ A_{1,A}(\nu_{xy}) + \frac{\alpha_A^2}{4} A_{2,A}(\nu_{xy}) \right] y_0^0(\hat{\mathbf{s}}_o) + \frac{1}{\sqrt{5}} \left[ -A_{1,A}(\nu_{xy}) + \frac{\alpha_A^2}{2} A_{2,A}(\nu_{xy}) \right] y_2^0(\hat{\mathbf{s}}_o), \quad (18)$$

where

$$A_{1,A}(\nu_{xy}) = \frac{2}{\pi} \left\{ \cos^{-1} \left( \frac{\nu_{xy}}{2\nu_{o,A}} \right) - \frac{\nu_{xy}}{2\nu_{o,A}} \sqrt{1 - \left( \frac{\nu_{xy}}{2\nu_{o,A}} \right)^2} \right\} \Pi \left( \frac{\nu_{xy}}{2\nu_{o,A}} \right), \quad (19)$$

$$A_{2,A}(\nu_{xy}) = \frac{2}{\pi} \left\{ \cos^{-1} \left( \frac{\nu_{xy}}{2\nu_{o,A}} \right) - \left[ 3 - 2 \left( \frac{\nu_{xy}}{2\nu_{o,A}} \right)^2 \right] \frac{\nu_{xy}}{2\nu_{o,A}} \sqrt{1 - \left( \frac{\nu_{xy}}{2\nu_{o,A}} \right)^2} \right\} \Pi \left( \frac{\nu_{xy}}{2\nu_{o,A}} \right). \quad (20)$$

We have closed form solutions for the inner Fourier transforms of Eq. 17, but we still need to evaluate the outer spherical Fourier transform. The function is the product of spherical Fourier transforms, so we need to take a spherical convolution of the coefficients of the spherical Fourier transforms of the individual functions. We could do this by hand and write out the result, but the equations would be long and wouldn't give much insight. Instead, we've implemented a spherical convolution that uses the Gaunt coefficients to efficiently compute the transfer function. The most important result is that we can compute  $H_{l,v}^m(\boldsymbol{\nu}, \hat{\mathbf{p}})$  directly without evaluating expensive Bessel functions then taking the Fourier transform.

We notice that both the excitation and detection kernels contain just a few spherical harmonic functions (four for the excitation kernel, two for the detection kernel), and these terms are all in the  $l = 0$  and  $l = 2$  bands. When we multiply out the spherical harmonics we get a total of 9 terms in the  $l = 0$ ,  $l = 2$ , and  $l = 4$  bands. This means that the sum over spherical harmonics in Eq. 13 contains only 9 terms which is an enormous computational savings over the complete spherical integral.

## 2.2 Continuous-to-discrete model

To convert the CC model to a CD model we weight Eq. 1 by a sampling aperture  $w_{\mathbf{r}_d, \hat{\mathbf{p}}}(\mathbf{r}_d, \hat{\mathbf{p}})$  and integrate over the continuous detector coordinates  $\mathbf{r}_d$  and  $\hat{\mathbf{p}}$  to obtain

$$g_{\mathbf{r}_d, \hat{\mathbf{p}}, v} = \int_{\mathbb{R}^3} d\mathbf{r}_d \int_{\mathbb{S}^1} d\hat{\mathbf{p}} w_{\mathbf{r}_d, \hat{\mathbf{p}}}(\mathbf{r}_d, \hat{\mathbf{p}}) \left[ \int_{\mathbb{R}^3} d\mathbf{r}_o \int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o h_v(\mathbf{r}_d - \mathbf{r}_o, \hat{\mathbf{p}}, \hat{\mathbf{s}}_o) f(\mathbf{r}_o, \hat{\mathbf{s}}_o) \right]. \quad (21)$$

Notice that we use monospace type for discrete (integer) indices— $\mathbf{p}$  is the polarizer setting index,  $\mathbf{v}$  is the view index, and  $\mathbf{r}_d$  is the voxel multi-index.

We will rearrange the integrals and group terms in Eq. 21 to find

$$g_{\mathbf{r}_d, \mathbf{p}, \mathbf{v}} = \int_{\mathbb{R}^3} d\mathbf{r}_o \int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o h_{\mathbf{r}_d, \mathbf{p}, \mathbf{v}}(\mathbf{r}_o, \hat{\mathbf{s}}_o) f(\mathbf{r}_o, \hat{\mathbf{s}}_o), \quad (22)$$

where the new CD kernel is related to the CC kernel by

$$h_{\mathbf{r}_d, \mathbf{p}, \mathbf{v}}(\mathbf{r}_o, \hat{\mathbf{s}}_o) = \int_{\mathbb{R}^3} d\mathbf{r}_d \int_{\mathbb{S}^1} d\hat{\mathbf{p}} w_{\mathbf{r}_d, \mathbf{p}}(\mathbf{r}_d, \hat{\mathbf{p}}) h_{\mathbf{v}}(\mathbf{r}_d - \mathbf{r}_o, \hat{\mathbf{p}}, \hat{\mathbf{s}}_o). \quad (23)$$

We can interpret the sampling aperture  $w_{\mathbf{r}_d, \mathbf{p}}(\mathbf{r}_d, \hat{\mathbf{p}})$  as the sensitivity of the  $(\mathbf{r}_d, \mathbf{p})$ th measurement to the irradiance at position  $(\mathbf{r}_d, \hat{\mathbf{p}})$ . The simplest choice for the sampling aperture is

$$w_{\mathbf{r}_d, \mathbf{p}}(\mathbf{r}_d, \hat{\mathbf{p}}) = \delta(\mathbf{r}_d - \epsilon \mathbf{r}_d) \delta(\hat{\mathbf{p}} - \epsilon_\phi \mathbf{p}), \quad (24)$$

where  $\epsilon$  is the distance between voxels and  $\epsilon_\phi$  is the angle between polarizer settings. With this choice of sampling aperture the CD forward model in Eq. 22 reduces to

$$g_{\mathbf{r}_d, \mathbf{p}, \mathbf{v}} = \int_{\mathbb{R}^3} d\mathbf{r}_o \int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o h_{\mathbf{v}}(\epsilon \mathbf{r}_d - \mathbf{r}_o, \epsilon_\phi \mathbf{p}, \hat{\mathbf{s}}_o) f(\mathbf{r}_o, \hat{\mathbf{s}}_o). \quad (25)$$

Notice that the CD kernel is a sampled version of the CC kernel, but this is only true when the sampling aperture is given by Eq. 24.

A more realistic model of the sampling aperture is given by

$$w_{\mathbf{r}_d, \mathbf{p}}(\mathbf{r}_d, \hat{\mathbf{p}}) = \frac{1}{\epsilon^3} \text{rect}\left(\frac{\mathbf{r}_d - \epsilon \mathbf{r}_d}{\epsilon}\right) \delta(\hat{\mathbf{p}} - \epsilon_\phi \mathbf{p}), \quad (26)$$

where the rect function models the aperture of each pixel. Finally, we mention the possibility of modeling a rotating polarizer by using the following sampling aperture

$$w_{\mathbf{r}_d, \mathbf{p}}(\mathbf{r}_d, \hat{\mathbf{p}}) = \frac{1}{\epsilon^3 \epsilon_\phi} \text{rect}\left(\frac{\mathbf{r}_d - \epsilon \mathbf{r}_d}{\epsilon}\right) \text{rect}\left(\frac{\hat{\mathbf{p}} - \epsilon_\phi \mathbf{p}}{\epsilon_\phi}\right). \quad (27)$$

We might consider these sampling aperture in the future, but for now we will use the sampling aperture in Eq. 24 and move forward with the CD model in Eq. 21.

### 2.3 Discrete-to-discrete model

To convert the CD model to a DD model we need to expand our object-space functions  $f(\mathbf{r}_o, \hat{\mathbf{s}}_o)$  onto a discrete set of expansion functions with

$$f(\mathbf{r}_o, \hat{\mathbf{s}}_o) = \sum_{\mathbf{r}_o, \mathbf{j}} \theta_{\mathbf{r}_o, \mathbf{j}} \phi_{\mathbf{r}_o, \mathbf{j}}(\mathbf{r}_o, \hat{\mathbf{s}}_o). \quad (28)$$

We will choose our expansion functions as

$$\phi_{\mathbf{r}_o, \mathbf{j}}(\mathbf{r}_o, \hat{\mathbf{s}}_o) = \delta(\mathbf{r}_o - \epsilon \mathbf{r}_o) y_j(\hat{\mathbf{s}}_o), \quad (29)$$

where  $j$  is a single index over the spherical harmonics. This choice of basis function may seem odd—we are representing the spatial part of our object in a sampled voxel basis and the angular part in a spherical harmonic basis. The main reason for this choice is that the diSPIM has a small angular rank, so we can't reconstruct more than a few angular coefficients anyway. By expanding our object in a spherical harmonic basis, we significantly

reduce the storage and computation requirements for our simulations and reconstructions without any loss of accuracy.

The expansion functions in Eq. 29 are orthonormal, but they do not form a complete basis for  $\mathbb{L}_2(\mathbb{R}^3 \times \mathbb{S}^2)$ . A good choice for a complete orthonormal basis is

$$\phi_{\mathbf{r}_o,j}(\mathbf{r}_o, \hat{\mathbf{s}}_o) = \frac{1}{\epsilon^3} \text{rect}\left(\frac{\mathbf{r}_o - \epsilon \mathbf{r}_o}{\epsilon}\right) y_j(\hat{\mathbf{s}}_o), \quad (30)$$

but we will use the basis in Eq. 29 for now to simplify our model.

For a given object  $f(\mathbf{r}_o, \hat{\mathbf{s}}_o)$ , we can find the expansion coefficients using

$$\theta_{\mathbf{r}_o,j} = \int_{\mathbb{R}^3} d\mathbf{r}_o \int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o \phi_{\mathbf{r}_o,j}(\mathbf{r}_o, \hat{\mathbf{s}}_o) f(\mathbf{r}_o, \hat{\mathbf{s}}_o). \quad (31)$$

Plugging Eq. 29 into Eq. 31 gives

$$\theta_{\mathbf{r}_o,j} = \int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o f(\mathbf{r}_o - \epsilon \mathbf{r}_o, \hat{\mathbf{s}}_o) y_j(\hat{\mathbf{s}}_o) = \mathcal{F}_{\mathbb{S}^2} \{f(\mathbf{r}_o - \epsilon \mathbf{r}_o, \hat{\mathbf{s}}_o)\} \quad (32)$$

which shows us the procedure to find the expansion coefficients of an object—sample the object on a spatial grid then take the spherical Fourier transform.

If we plug Eq. 28 into Eq. 21 we find

$$g_{\mathbf{r}_d,p,v} = \int_{\mathbb{R}^3} d\mathbf{r}_o \int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o h_v(\epsilon \mathbf{r}_d - \mathbf{r}_o, \epsilon_\phi p, \hat{\mathbf{s}}_o) \left[ \sum_{\mathbf{r}_o,j} \theta_{\mathbf{r}_o,j} \phi_{\mathbf{r}_o,j}(\mathbf{r}_o, \hat{\mathbf{s}}_o) \right]. \quad (33)$$

After plugging in the expansion functions in Eq. 29, evaluating the spatial integral, and rearranging terms we find

$$g_{\mathbf{r}_d,p,v} = \sum_{\mathbf{r}_o,j} \int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o h_v(\epsilon[\mathbf{r}_d - \mathbf{r}_o], \epsilon_\phi p, \hat{\mathbf{s}}_o) y_j(\hat{\mathbf{s}}_o) \theta_{\mathbf{r}_o,j}. \quad (34)$$

We can rewrite this equation as a DD model

$$g_{\mathbf{r}_d,p,v} = \sum_{\mathbf{r}_o,j} h_{\mathbf{r}_d - \mathbf{r}_o, p, v, j} \theta_{\mathbf{r}_o,j}, \quad (35)$$

where

$$h_{\mathbf{r},p,v,j} = \int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o h_v(\epsilon \mathbf{r}, \epsilon_\phi p, \hat{\mathbf{s}}_o) y_j(\hat{\mathbf{s}}_o). \quad (36)$$

Eqs. 35 and 36 specify a discrete forward model, but they are expensive to compute because of the sum over  $\mathbf{r}$  and the integral over  $\mathbb{S}^2$ . Following the simplifications we made in Section 2.1, we apply the discrete convolution multiplication theorem to rewrite the forward model as

$$g_{\mathbf{r}_d,p,v} = \text{DFT}^{-1} \left\{ \sum_j H_{\bar{\nu},p,v,j} \text{DFT} \{ \theta_{\mathbf{r}_o,j} \} \right\}, \quad (37)$$

where

$$H_{\bar{\nu},p,v,j} = \int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o \text{DFT} \{ h_v(\epsilon \mathbf{r}, \epsilon_\phi p, \hat{\mathbf{s}}_o) \} y_j(\hat{\mathbf{s}}_o) = H_{v,j}(\bar{\nu}/\epsilon, \epsilon_\phi p), \quad (38)$$

where the far LHS is the discrete spatio-angular transfer function and the far RHS is the sampled continuous spatio-angular transfer function from Section 2.1.

The forward model in Eq. 37 is extremely fast compared to the naive discretization we considered previously. The 3D DFTs require approximately  $3N \log N$  FLOPs (about  $10^4$  FLOPs for  $N = 10^3$ ). We know that the sum over  $j$  has 15 terms, and we need to calculate these 15 DFTs for each of 4 polarizer settings and 2 views. Therefore, our approximate FLOP count is  $10^4 \times 15 \times 4 \times 2 = 1.2 \times 10^6$ —about 15 orders of magnitude faster than our naive approach.

## 2.4 Implementation details

For small input sizes we can precompute the entire array  $H_{\bar{\nu}, p, v, j}$  and apply Eq. 37 directly. This array quickly becomes too large to store in memory though— $10^9$  spatial points  $\times$  4 polarizers  $\times$  2 views  $\times$  15 spherical harmonics  $\times$  32 bits = 480 GB. To avoid memory issues, we separate  $H_{\bar{\nu}, p, v, j}$  into transverse and axial parts, precompute and store these small arrays, then assemble the complete array in pieces during the application of the forward model. We also exploit the 8-fold spatial symmetry of the array  $H_{\bar{\nu}, p, v, j}$  and only store the positive octant. Finally, the sum over  $j$  needs to be computed at each point in  $\bar{\nu}$ , so we parallelize this loop over multiple CPUs.

## 3 Inverse problem

It will be convenient to rewrite the forward model in Eq. 37 completely in the spatial frequency domain as

$$G_{\bar{\nu}, p, v} = \sum_j H_{\bar{\nu}, p, v, j} \Theta_{\bar{\nu}, j}, \quad (39)$$

where

$$G_{\bar{\nu}, p, v} = \text{DFT} \{ g_{r_a, p, v} \}, \quad (40)$$

$$\Theta_{\bar{\nu}, j} = \text{DFT} \{ \theta_{r_o, j} \}. \quad (41)$$

If we combine the indices  $p$  and  $v$  into a single multi-index  $q = (p, v)$  then the forward model becomes

$$G_{\bar{\nu}, q} = \sum_j H_{\bar{\nu}, q, j} \Theta_{\bar{\nu}, j} \quad (42)$$

which has the form of a matrix multiplication at each discrete spatial frequency  $\bar{\nu}$ . We can rewrite Eq. 42 in matrix notation as

$$\mathbf{G}_{\bar{\nu}} = \mathbf{H}_{\bar{\nu}} \boldsymbol{\Theta}_{\bar{\nu}}. \quad (43)$$

Taking the SVD of  $\mathbf{H}_{\bar{\nu}}$  gives

$$\mathbf{G}_{\bar{\nu}} = \sum_{k=0}^R \sqrt{\mu_{\bar{\nu}, k}} \mathbf{v}_{\bar{\nu}, k} \mathbf{u}_{\bar{\nu}, k}^\dagger \boldsymbol{\Theta}_{\bar{\nu}}, \quad (44)$$

where  $\mu_{\bar{\nu}, k}$ ,  $\mathbf{v}_{\bar{\nu}, k}$ ,  $\mathbf{u}_{\bar{\nu}, k}$ , and  $R$  are the singular values, data space singular vectors, object space singular vectors, and rank of  $\mathbf{H}_{\bar{\nu}}$ , respectively. Notice that Eq. 44 gives us an even more efficient way to compute the forward model—we only need to compute the sum up to the rank instead of summing over the spherical harmonic coefficients. Unfortunately, this trick is only feasible if we can store the singular value decomposition of  $\mathbf{H}_{\bar{\nu}}$  in memory. In our case we have to create  $\mathbf{H}_{\bar{\nu}}$  on the fly, and it would be too expensive to calculate the SVD at each point on the fly.

We would like to estimate  $\boldsymbol{\Theta}_{\bar{\nu}}$  from noise-corrupted measurements  $\mathbf{G}_{\bar{\nu}}$ . We can rearrange Eq. 44 to find

$$\hat{\boldsymbol{\Theta}}_{\bar{\nu}} = \sum_{k=0}^R \frac{1}{\sqrt{\mu_{\bar{\nu}, k}}} \mathbf{u}_{\bar{\nu}, k} \mathbf{v}_{\bar{\nu}, k}^\dagger \mathbf{G}_{\bar{\nu}}. \quad (45)$$

where  $\hat{\cdot}$  denotes an estimate. This solution is called the Moore-Penrose pseudoinverse and it gives a least-squares solution to any set of linear equations. Notice that the sum in Eq. 45 is only up to the rank  $R$ , so this solution avoids division by zero.

In practice, the pseudoinverse solution is susceptible to corruption by noise because of division by small singular values. To avoid this problem, we use a Tikhonov regularization parameter that suppresses the terms in Eq. 45 with small singular values. The Tikhonov-regularized solution is given by

$$\hat{\Theta}_{\bar{\nu}} = \sum_{k=0}^R \frac{\sqrt{\mu_{\bar{\nu},k}}}{\mu_{\bar{\nu},k} + \eta} \mathbf{u}_{\bar{\nu},k} \mathbf{v}_{\bar{\nu},k}^\dagger \mathbf{G}_{\bar{\nu}}, \quad (46)$$

where  $\eta$  is the regularization parameter.

Although we have derived the Tikhonov-regularized solution in a heuristic way in this section, it is possible to show that Eq. 46 is the solution to the following optimization problem

$$\hat{\Theta}_{\bar{\nu}} = \operatorname{argmin}_{\Theta_{\bar{\nu}}} \left\{ \|\mathbf{G}_{\bar{\nu}} - \mathbf{H}_{\bar{\nu}} \Theta_{\bar{\nu}}\|^2 + \eta \|\Theta_{\bar{\nu}}\|^2 \right\}. \quad (47)$$

In this form it is clear that increasing the regularization parameter  $\eta$  favors solutions with a smaller norm.

After calculating  $\hat{\Theta}_{\bar{\nu}}$  we can find the object-space coefficients  $\hat{\theta}_{\mathbf{r}_o, j}$  with an inverse DFT, and we have completed the inversion. In pseudocode the complete reconstruction algorithm is given by

---

**Algorithm 1:** diSPIM spatio-angular reconstruction

---

**Input :** Data array  $g_{\mathbf{r}_d, p, v}$ , regularization parameter  $\eta$   
**Output :** Estimate of spatio-angular density coefficients  $\hat{\theta}_{\mathbf{r}_o, j}$

- 1  $\hat{\Theta}_{\bar{\nu}, j} \leftarrow \text{Zeros}()$
- 2  $G_{\bar{\nu}, p, v} \leftarrow \text{DFT}(g_{\mathbf{r}_d, p, v})$
- 3 **for each**  $\bar{\nu}$  in  $\bar{\nu}$ :
- 4      $H_{p, v, j} \leftarrow \text{CalculateH}(\bar{\nu}, p, v, j)$
- 5      $G_{\bar{\nu}, q} \leftarrow \text{Flatten}(G_{\bar{\nu}, p, v}, [p, v])$
- 6      $H_{q, j} \leftarrow \text{Flatten}(H_{p, v, j}, [p, v])$
- 7      $U_{k, j}, \mu_k, V_{k, q} \leftarrow \text{SVD}(H_{q, j})$
- 8      $\hat{\Theta}_{\bar{\nu}, j} \leftarrow \sum_k^R \sum_q \frac{\sqrt{\mu_k}}{\mu_k + \eta} U_{k, j} V_{k, q} G_{\bar{\nu}, q}$
- 9      $\hat{\theta}_{\mathbf{r}_o, j} \leftarrow \text{DFT}^{-1}(\hat{\Theta}_{\bar{\nu}, j})$ .

---

Similar to the forward model calculation, we assemble  $H_{p, v, j}$  from precalculated matrices to avoid memory limitations, we exploit the 8-fold symmetry of the kernel to reduce the number of SVDs calculated in the loop, and we parallelize the loop over multiple CPUs.

### 3.1 Summary parameters

Algorithm 1 shows the complete spatio-angular reconstruction, but our choice to reconstruct onto a spherical harmonic basis makes the result difficult to interpret. Therefore, we need another set of computations that can convert the spherical harmonic coefficients at each spatial point to interpretable values. Recall that we cannot reconstruct directly to a spatio-angular density because the result will not fit in memory—we need to reconstruct the spherical harmonic coefficients then calculate summary parameters only as necessary.

The most informative (and expensive) set of summary parameters is to calculate the complete orientation distribution function (ODF) from the set of spherical harmonic coefficients. This computation is essentially an inverse spherical Fourier transform given by

$$f_{\mathbf{r}_o}(\hat{\mathbf{s}}_o) = \sum_j \theta_{\mathbf{r}_o, j} y_j(\hat{\mathbf{s}}_o). \quad (48)$$

To calculate a visually informative ODF we need to evaluate  $f_{\mathbf{r}_o}(\hat{\mathbf{s}}_o)$  at approximately 500 points, so calculating the ODF amounts to a  $15 \times 500$  matrix multiplication at each voxel. This matrix multiplication is the most expensive computation in the entire pipeline, and it is not feasible to compute it in every voxel for large data sets. Typically we only calculate and visualize a subset of the ODFs—we either restrict ourselves to an ROI or calculate and visualize ODFs in every  $N$ th voxel.

Another useful summary parameter is the spatial density of fluorophores. For angularly continuous functions the spatial density  $\rho_{\mathbf{r}_o}$  is related to the spatio-angular density  $f_{\mathbf{r}_o}(\hat{\mathbf{s}}_o)$  by

$$\rho_{\mathbf{r}_o} = \int_{\mathbb{S}^2} f_{\mathbf{r}_o}(\hat{\mathbf{s}}_o). \quad (49)$$

The naive approach to computing  $\rho_{\mathbf{r}_o}$  is to calculate  $f_{\mathbf{r}_o}(\hat{\mathbf{s}}_o)$  using Eq. 48 then apply Eq. 49. Instead we can calculate the density directly from the spherical harmonic coefficients using

$$\rho_{\mathbf{r}_o} = \int_{\mathbb{S}^2} \sum_j \theta_{\mathbf{r}_o,j} y_j(\hat{\mathbf{s}}_o), \quad (50)$$

then rearranging the sum and integral

$$\rho_{\mathbf{r}_o} = \sum_j \theta_{\mathbf{r}_o,j} \int_{\mathbb{S}^2} y_j(\hat{\mathbf{s}}_o), \quad (51)$$

and finally noting that only the  $l = 0$  spherical harmonics have a non-zero integral over the sphere

$$\rho_{\mathbf{r}_o} = \sum_l \theta_{\mathbf{r}_o,l} \int_{\mathbb{S}^2} y_l^0(\hat{\mathbf{s}}_o). \quad (52)$$

We can precompute the integrals then apply Eq. 52 as a weighted sum over just three spherical harmonic coefficients.

Another useful summary parameter is the generalized fractional anisotropy (GFA) that was defined in (Tuch 2004) to extend the fractional anisotropy used in diffusion tensor MRI to high-angular resolution MRI. Tuch defines the GFA as

$$\vartheta_{\mathbf{r}_o} = \frac{\text{StandardDeviation } \{f_{\mathbf{r}_o}(\hat{\mathbf{s}}_o)\}}{\text{RootMeanSquare } \{f_{\mathbf{r}_o}(\hat{\mathbf{s}}_o)\}}. \quad (53)$$

Similar to the density, the naive approach to computing  $\vartheta_{\mathbf{r}_o}$  would be to sample using Eq. 48 then apply Eq. 53. We can improve on this by writing Eq. 53 in terms of integrals then simplifying the integrals. We start with

$$\vartheta_{\mathbf{r}_o} = \sqrt{\frac{\int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o [f(\hat{\mathbf{s}}_o) - (\rho_{\mathbf{r}_o}/4\pi)]^2}{\int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o [f(\hat{\mathbf{s}}_o)]^2}}, \quad (54)$$

$$(55)$$

then plug in the spherical harmonic coefficients

$$\vartheta_{\mathbf{r}_o} = \sqrt{\frac{\int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o \left[ \sum_j \theta_{\mathbf{r}_o,j} y_j(\hat{\mathbf{s}}_o) - (\rho_{\mathbf{r}_o}/4\pi) \right]^2}{\int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o \left[ \sum_j \theta_{\mathbf{r}_o,j} y_j(\hat{\mathbf{s}}_o) \right]^2}}. \quad (56)$$

After expanding the square brackets and exploiting the orthogonality of the spherical harmonics we end up with

$$\vartheta_{\mathbf{r}_o} = \sqrt{\frac{\sum_j \theta_{\mathbf{r}_o,j}^2 + \rho_{\mathbf{r}_o}^2/2\pi + (\rho_{\mathbf{r}_o}/4\pi)^2}{\sum_j \theta_{\mathbf{r}_o,j}^2}}. \quad (57)$$

The final summary parameter we would like to calculate is the direction along which the most fluorophores are oriented

$$\hat{\mathbf{s}}_{\mathbf{r}_o,\max} = \operatorname{argmax}_{\hat{\mathbf{s}}_o} f_{\mathbf{r}_o}(\hat{\mathbf{s}}_o). \quad (58)$$

Our problem is to find the maximum point of a function on a sphere given its spherical harmonic coefficients. If we can tolerate an angular error of  $\Delta\hat{\mathbf{s}}$  steradians, then the brute-force approach is to sample the sphere at  $4\pi/\Delta\hat{\mathbf{s}}$  approximately equally spaced points then choose the maximum.

A faster approach is to use a simple optimization scheme that first samples coarsely over the sphere then performs a gradient descent on the best of the coarse samples. In general this type of scheme is not guaranteed to find the global maximum, but I expect that a wise choice of initial sample points will guarantee the global maximum for our bandlimited signals.

Currently I am calculating all of the summary statistics using the naive approaches, and these calculations are the bottle neck in the reconstruction pipeline. I will implement the improved routines soon.

Parameter name	Symbol	Interpretation
Orientation distribution function (ODF)	$f_{\mathbf{r}_o}(\hat{\mathbf{s}}_o)$	Number of fluorophores at $\mathbf{r}_o$ oriented along $\hat{\mathbf{s}}_o$ .
Density	$\rho_{\mathbf{r}_o}$	Number of fluorophores at $\mathbf{r}_o$ .
Generalized fractional anisotropy (GFA)	$\vartheta_{\mathbf{r}_o}$	A scale-invariant statistic of the angular spread of fluorophores at $\mathbf{r}_o$ . $\vartheta_{\mathbf{r}_o} = 1$ for very anisotropic distributions and $\vartheta_{\mathbf{r}_o} = 0$ for isotropic fluorophores.
Peak direction (mode)	$\hat{\mathbf{s}}_{\mathbf{r}_o,\max}$	Direction along which most fluorophores are oriented at $\mathbf{r}_o$ .

Table 1: Summary of angular summary statistics.

## 4 Numerical results

### 4.1 Bead studies

Figures 1–4 show the results of numerical bead studies that we used to debug and verify our forward model. We placed a single bead in the center of a small field-of-view then applied our forward model to generate data that would be collected under the four polarizer settings and two viewing directions.

First, we generated four numerical phantoms— $\hat{\mathbf{x}}$ ,  $\hat{\mathbf{y}}$ , and  $\hat{\mathbf{z}}$  oriented beads and an isotropic bead. The continuous spatio-angular densities of these phantoms are given by

$$f_1(\mathbf{r}_o, \hat{\mathbf{s}}_o) = \delta(\mathbf{r}_o) \delta(\hat{\mathbf{s}}_o - \hat{\mathbf{x}}), \quad (59)$$

$$f_2(\mathbf{r}_o, \hat{\mathbf{s}}_o) = \delta(\mathbf{r}_o) \delta(\hat{\mathbf{s}}_o - \hat{\mathbf{y}}), \quad (60)$$

$$f_3(\mathbf{r}_o, \hat{\mathbf{s}}_o) = \delta(\mathbf{r}_o) \delta(\hat{\mathbf{s}}_o - \hat{\mathbf{z}}), \quad (61)$$

$$f_4(\mathbf{r}_o, \hat{\mathbf{s}}_o) = \delta(\mathbf{r}_o). \quad (62)$$

Next, we set the FOV by choosing the number of voxels (32 per side) and the voxel dimensions (130 nm per

side). Finally, we calculated the object-space coefficients  $\theta_{\mathbf{r}_o, j}$  from the spatio-angular density using

$$\theta_{\mathbf{r}_o, j} = \int_{\mathbb{S}^2} d\hat{\mathbf{s}}_o f(\mathbf{r}_o - \epsilon \mathbf{r}_o, \hat{\mathbf{s}}_o) y_j(\hat{\mathbf{s}}_o). \quad (63)$$

Now that we have a numerical phantom, the next step is to specify a microscope. We used the following parameters for our numerical simulations

$$NA_A = 1.1, \quad (64)$$

$$NA_B = 0.71, \quad (65)$$

$$\lambda_{em} = 525 \text{ nm} \leftarrow \text{Alexa 488 emission}, \quad (66)$$

$$n = 1.33 \leftarrow \text{water}. \quad (67)$$

We use these parameters when we calculate the elements of  $H_{\nu, p, v, j}$ . Finally, we apply Eq. 37 to calculate the data  $g_{\mathbf{r}_d, p, v}$ . Figures 1–4 show summary views of the five-dimensional datasets  $g_{\mathbf{r}_d, p, v}$ . For each polarizer setting and view we plot the maximum intensity projection along each of the three axes.

Although the entire kernel is specified exactly in section 1, it is useful to have a few rules of thumb in mind when you look at the bead results:

1. The probability of exciting a dipole is proportional to  $|\hat{\mathbf{p}} \cdot \mu|^2 = \cos^2 \theta$  where  $\hat{\mathbf{p}}$  is the polarizer orientation and  $\mu$  is the excitation dipole moment—lots of excitation when the polarization is parallel to the dipole moment.
2. A dipole emitter radiates along a direction  $\hat{\mathbf{s}}$  with a power proportional to  $1 - |\hat{\mathbf{s}} \cdot \mu|^2 = \sin^2 \theta$ —lots of radiation in the plane perpendicular to the dipole moment.
3. An objective with  $NA > 0$  will collect the radiation from a cone of  $\hat{\mathbf{s}}$  directions.
4. When a dipole moment is parallel to the detection optical axis, then the light in the back focal plane will be radially polarized and the resulting image will look like a doughnut.

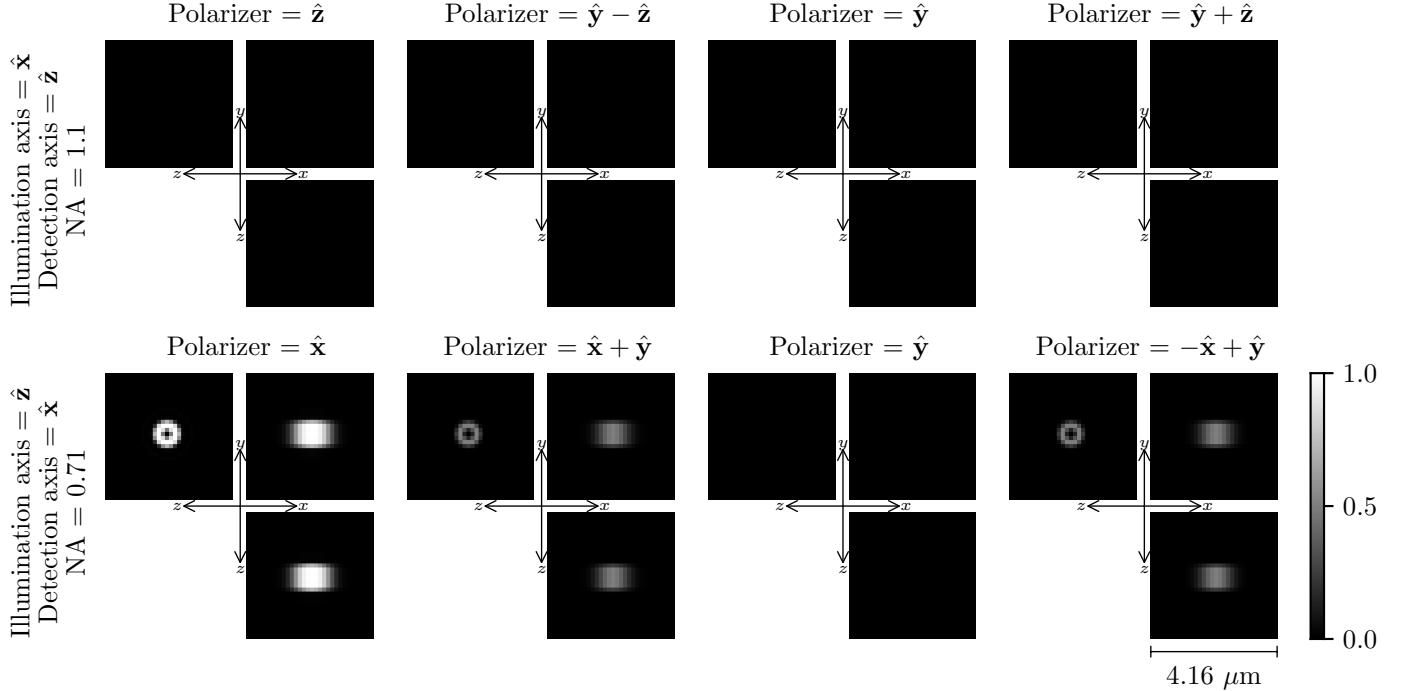


Figure 1: Simulated data from a bead with  $\hat{\mathbf{x}}$ -oriented dipole moments.

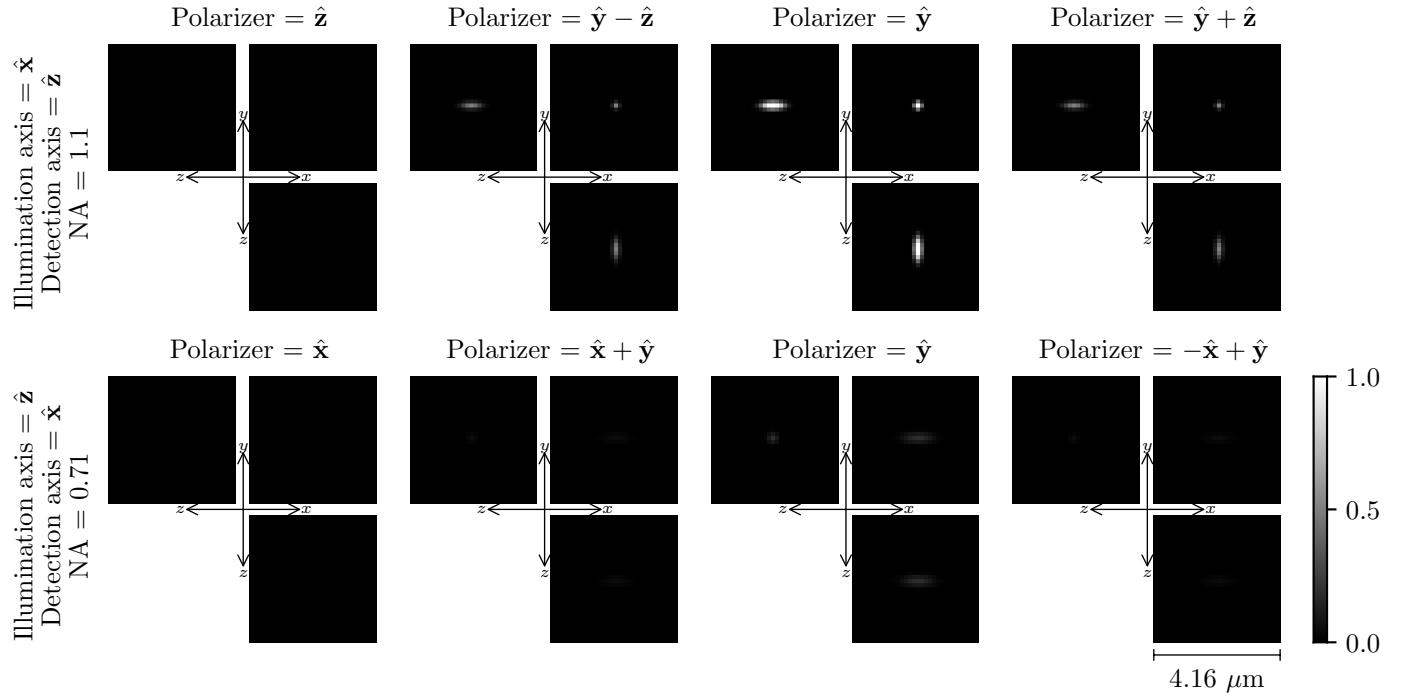


Figure 2: Simulated data from a bead with  $\hat{\mathbf{y}}$ -oriented dipole moments.

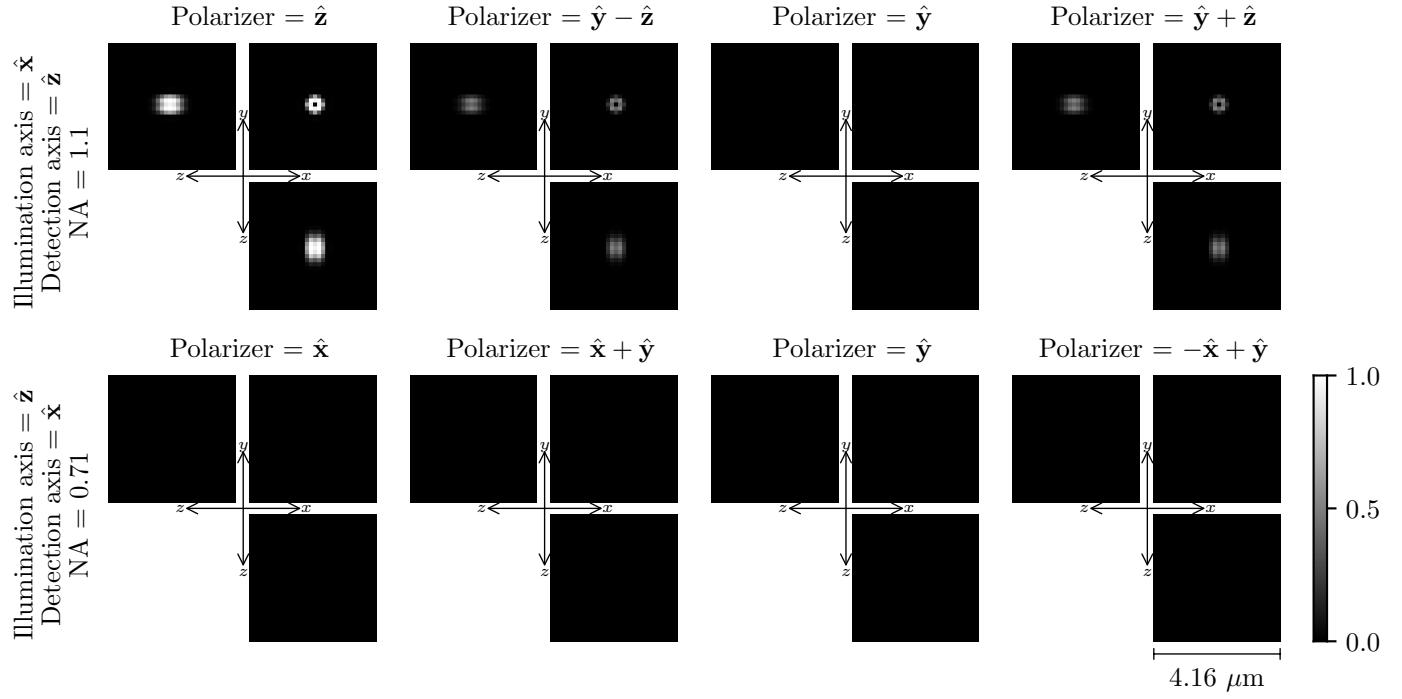


Figure 3: Simulated data from a bead with  $\hat{\mathbf{z}}$ -oriented dipole moments.

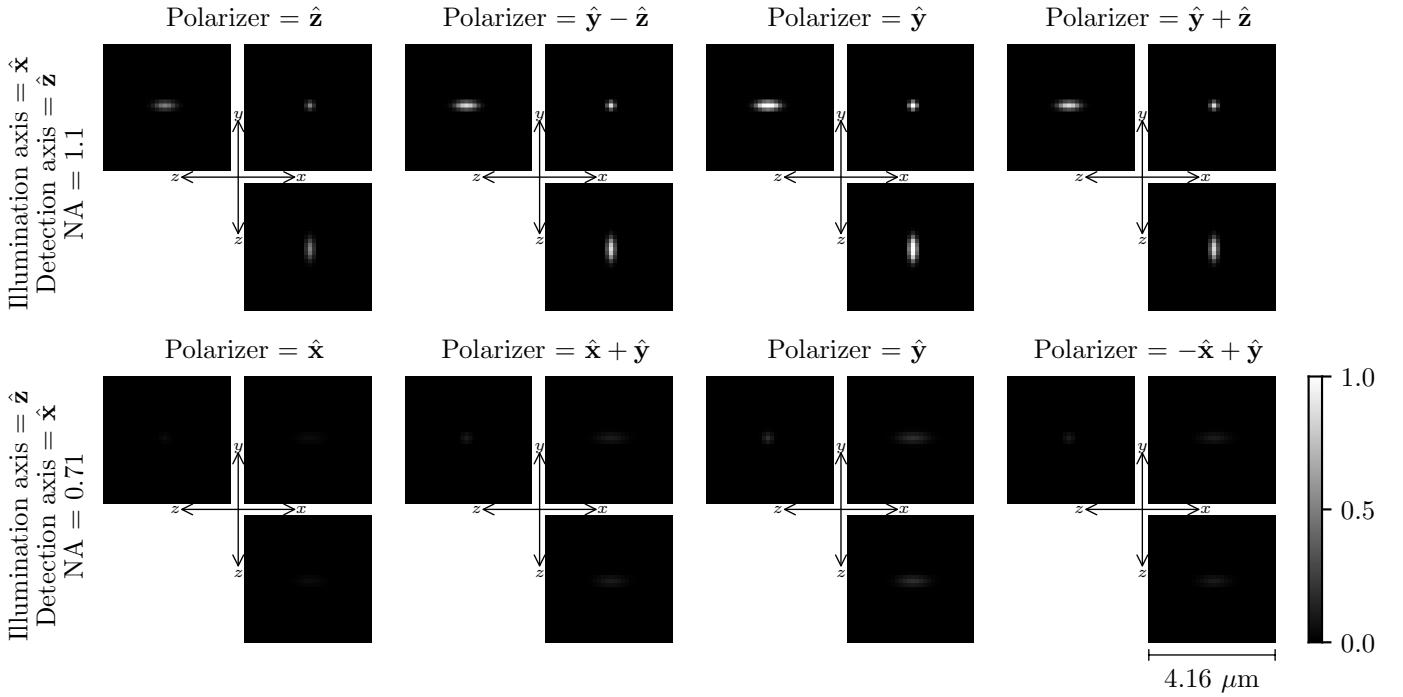


Figure 4: Simulated data from an isotropic bead.

We have made several assumptions about the axial part of the kernel. First, we assume that the axial part can be described by a Gaussian. The axial part would be more accurately described by a widefield fluorescence kernel convolved with the Gaussian width of the light sheet, but we expect that the width of the light sheet dominates the kernel so a pure Gaussian is appropriate. Second, we have assumed that the width of the light sheet is constant which allows us to describe the system as shift-invariant. This assumption has been used in previous diSPIM deconvolutions, and it allows us to keep the computations tractable. Finally, we have chosen the width of the axial Gaussian to be  $\sim 3\times$  broader than the transverse width of the transverse kernel of an isotropic bead. This width was chosen to match the axial FWHM of the kernels supplied by Min in PSFA\_1p1NA.tif and PSFB\_0p71NA.tif. **Do you think these assumptions are appropriate? Do the bead PSFs look reasonable?**

## 4.2 Helix studies

We created a more complex spatio-angular phantom to test our forward model and inversion scheme. The helix phantom is shown in Figure 5—it consists of three helices with different coil directions and dipoles that are aligned along the tangent of the helix at each point. On one end of the helix the dipoles are almost completely aligned along the tangent, and at the other end of the helix the dipoles are isotropic.

We would like to express the helix phantom as a continuous spatio-angular density. First, we specify a single helix with a parameterized line

$$c(t) = (r_h \cos t, r_h \sin t, p_h t) \quad \text{for } t \in [-2\pi, 2\pi]. \quad (68)$$

Next, for each point in space  $\mathbf{r}_o$  we find the closest point on the helix then find the corresponding parameter  $t$  and name it  $t_{\mathbf{r}_o}^*$

$$t_{\mathbf{r}_o}^* = \min_t \|c(t) - \mathbf{r}_o\|^2. \quad (69)$$

Next, we build the complete helix using

$$f_{\text{helix}}(\mathbf{r}_o, \hat{\mathbf{s}}_o) = \begin{cases} w(\mathbf{r}_o, \hat{\mathbf{s}}_o, c'(t_{\mathbf{r}_o}^*), \kappa(t_{\mathbf{r}_o}^*)), & t_{\mathbf{r}_o}^* < r_{\text{cyl}} \\ 0 & t_{\mathbf{r}_o}^* > r_{\text{cyl}} \end{cases} \quad (70)$$

where

$$w(\mathbf{r}_o, \hat{\mathbf{s}}_o, \hat{\mathbf{n}}, \kappa) = \frac{1}{4\pi_1 F_1(1/2, 3/2, \kappa)} e^{\kappa(\hat{\mathbf{n}} \cdot \hat{\mathbf{s}}_o)}, \quad (71)$$

$$\kappa(t) = kt + k_o, \quad (72)$$

and  $F_1(1/2, 3/2, \kappa)$  is the Kummer confluent hypergeometric function. Eq. 70 restricts the phantom to within  $r_{\text{cyl}}$  of the helix, Eq. 71 is a normalized Watson distribution with spread parameter  $\kappa$ , and Eq. 72 linearly increases the spread parameter as we move along the helix.

Finally, we assemble three helices

$$f_{\text{phantom}}(\mathbf{r}_o, \hat{\mathbf{s}}_o) = f_{\text{helix}}(\mathbf{r}_o - \mathbf{r}_0, \hat{\mathbf{s}}_o) + \mathcal{R}_x \{f_{\text{helix}}(\mathbf{r}_o - \mathbf{r}_1, \hat{\mathbf{s}}_o)\} + \mathcal{R}_y \{f_{\text{helix}}(\mathbf{r}_o - \mathbf{r}_2, \hat{\mathbf{s}}_o)\}, \quad (73)$$

where  $\mathcal{R}_x$  is a rotation operator that rotates the helix direction to along the  $\hat{\mathbf{x}}$  direction. For our simulations we chose the following set of parameters

$$r_h = 700 \text{ nm}, \quad (74)$$

$$p_h = 1000 \text{ nm}, \quad (75)$$

$$\mathbf{r}_0 = (-2080, -2080, 2080) \text{ nm}, \quad (76)$$

$$\mathbf{r}_1 = (0, 0, 0) \text{ nm}, \quad (77)$$

$$\mathbf{r}_2 = (2080, 2080, 2080) \text{ nm}, \quad (78)$$

$$r_{\text{cyl}} = 250 \text{ nm}, \quad (79)$$

$$k = \frac{5}{4\pi}, \quad (80)$$

$$k_0 = 2\pi. \quad (81)$$

We calculated the coefficients of the object using Eq. 63, then simulated the noise-free data with the same imaging parameters as the previous section. The data we generated is shown in Figure 6.

We applied our spatio-angular restoration algorithm to the noise-free data and recovered the object in Figure 7. Alternatively, we can interpret Figure 7 as the measurement-space component of the helix phantom—we applied the forward model then reconstructed with the pseudoinverse solution which is identical to projection onto the measurement space (Barrett Table 1.2).

Finally, we simulated data using the helix phantom with Poisson noise at  $\text{SNR} = 10$  (the maximum number of photons is 100), then attempted to reconstruct the object with several different Tikhonov regularization parameters. Figure 8 shows that the mean square error (MSE) between the phantom and the reconstructed object has a minimum at a specific regularization parameter as we would expect. The optimal regularization parameter will change as we vary the SNR, but we can use Figure 8 as evidence that the inversion algorithm is working well.

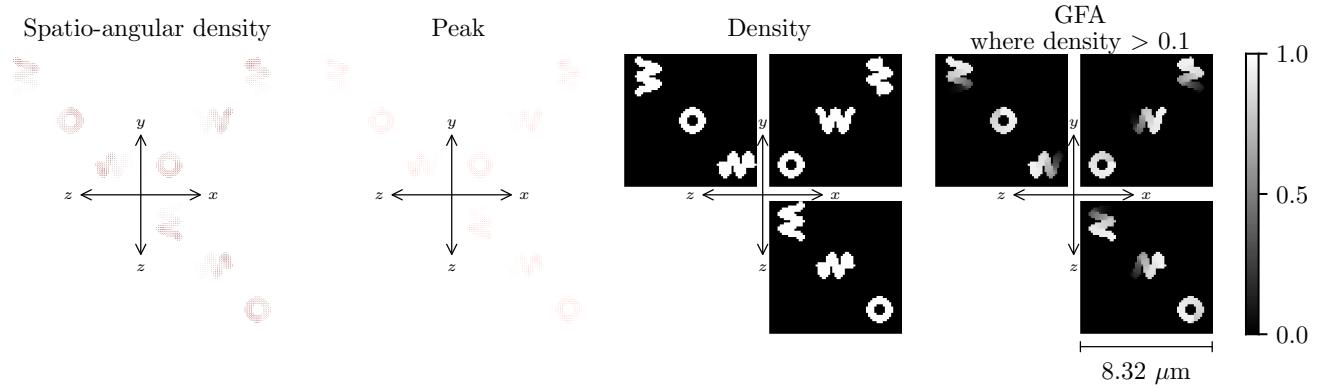


Figure 5: Helix phantom summary.

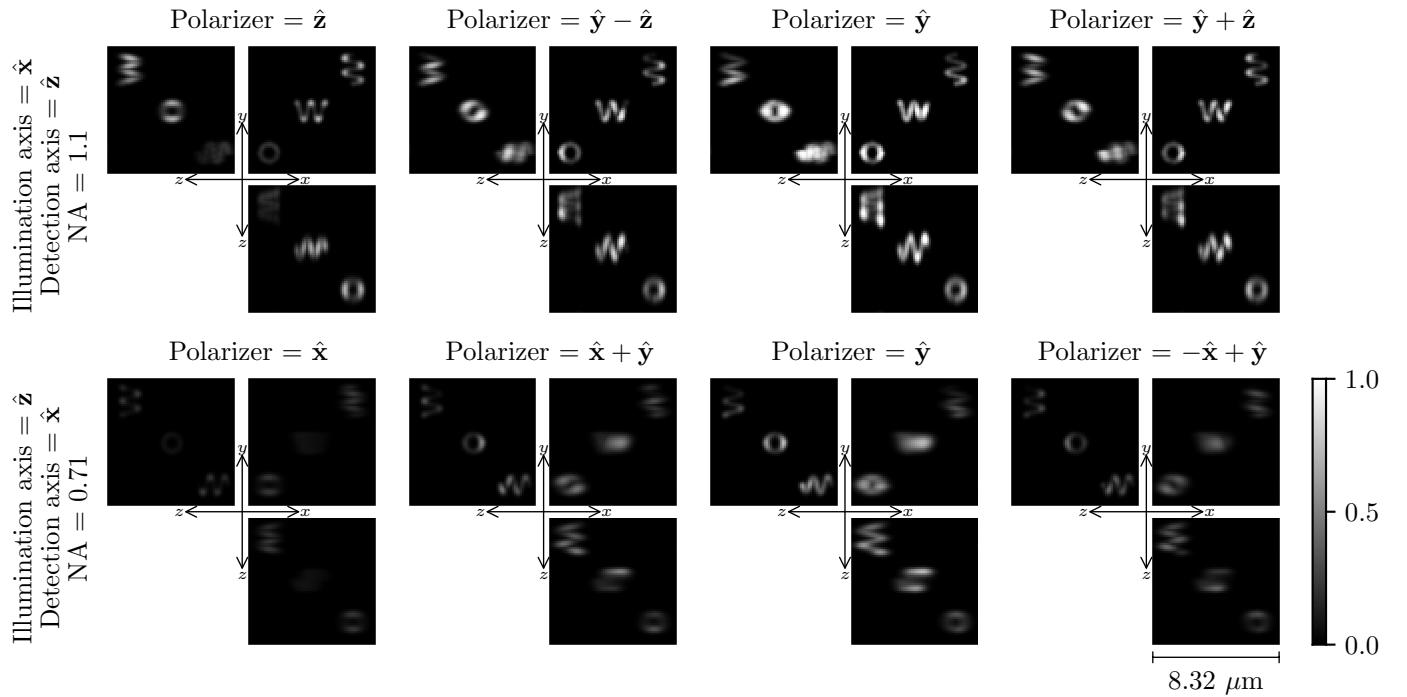


Figure 6: Data generated by the helix phantom.

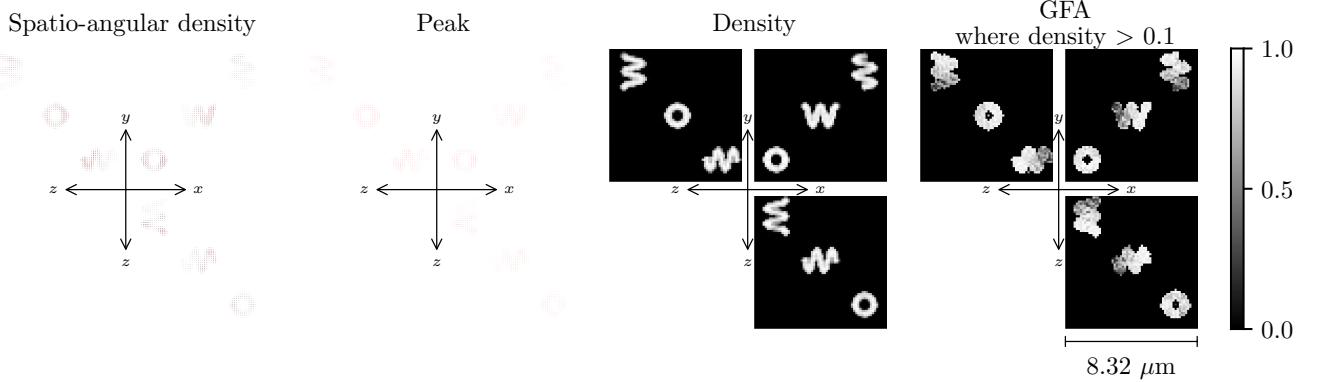


Figure 7: Helix phantom reconstruction.

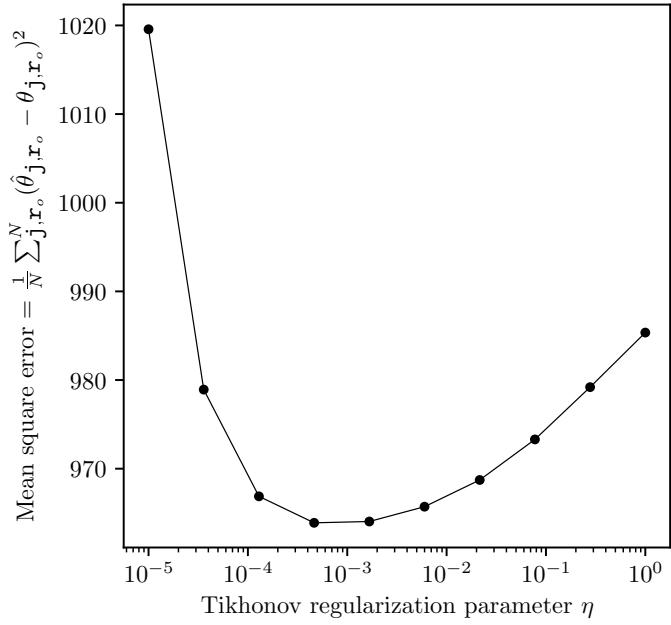


Figure 8: Mean square error of helix phantom reconstruction with SNR = 10.

## 5 Experimental results

We collected several data sets with the polarized light asymmetric diSPIM (1.1 and 0.71 NA). In this section we will review the preprocessing, calibration, and reconstructed results.

### 5.1 Min's preprocessing

Min applies the following preprocessing steps to the raw data

1. “Deshearing” the data to correct for stage scanning.
2. ROI selection to reduce the size of the data.
3. “Isometrization” to convert all of the data to isometric voxels.
4. Registration of the two views so that they are in the same coordinate system.

### 5.2 Talon's preprocessing

I applied the following preprocessing steps to prepare Min's preprocessed data for spatio-angular restoration.

1. Background correction.
2. Correcting the views for power differences.
3. Correcting the views for voxel size differences.
4. Correcting the volumes (each view and polarization) with polarization calibration data.
5. TODO: I will apply a bleaching correction based on the first and last polarizer setting. The data in these notes had negligible bleaching, so I haven't needed this correction yet.

For the background correction I subtracted the mean of a background ROI from all of the data. I applied the same background correction to the volumes from each polarizer setting, but I applied different background corrections to the data from each view.

To correct the views for power differences I multiplied the data from the high-NA view by a factor of  $(1.1/0.71)^2 = 2.4$ . This correction models the fact we balanced the input power of the two views so that the pixels from each view would collect approximately the same number of photons.

Next, I applied a correction to account for the different voxel sizes of the raw data. Min's preprocessing outputs isometric voxels, but it does not preserve the total number of counts before and after the correction. For example, if we put a single  $(100, 100, 200)$  nm voxel with a value of 50 into Min's preprocessing algorithm, the output would be two  $(100, 100, 100)$  nm voxels each with a value of 50. I am applying a correction factor so that the output would be two  $(100, 100, 100)$  nm voxels each with a value of 25.

In the GUV dataset the raw voxel size in view A is  $(130, 130, 549)$  nm, and the raw voxel size in view B is  $(227, 227, 835.8)$  nm. To correct for this difference I multiply the data in view A by a factor of  $(227/130)^2(835.8/549) = 4.64$ . [Min, Hari—does this make sense?](#)

Finally, I applied a correction factor to each volume to account for imperfections in the polarized illumination due to misalignment or polarization errors introduced during reflection by the dichroics. To collect calibration data Min imaged the center of a fluorescent lake of uniformly oriented fluorophores under each polarizer setting. He also collected data from the epi path and the imaging path for a total of 16 volumes (epi/imaging, 4 polarizer settings, 2 views). I manually selected an ROI in each of these 16 volumes and stored the mean value of these ROIs. Next, I applied correction factors to correct for errors on the illumination path. We would expect that the intensity measured from the epi path should be constant as the polarization changes, so I applied correction factors to account for any deviation from a constant—the largest correction factor was approximately 5%. Finally, I applied correction factors to correct for errors on the detection path. We have a model that predicts the data we would expect to collect from a uniform lake of uniformly oriented fluorophore—namely, the transfer function evaluated at zero spatio-angular frequency  $H_{0,v}^0(0, \hat{\mathbf{p}})$ . I compared the calibration data to our model and applied correction factor to account for any differences—the largest correction factor was approximately 10%. [Min, Hari—does this make sense?](#)

I applied each of the correction factors to the data, but I could have applied the correction factors to the model. I have chosen to correct the data to make the comparison with the simulated data in the previous section easier.

### 5.3 GUV study

Mai prepared a sample of giant unilamellar vesicles (GUVs) labelled with Alexa Fluor 488. From two-dimensional fluorescence orientation studies we expect to observe the dipole moments oriented normal to the surface of the vesicles.

Figure 9 shows the corrected data and Figure 10 shows the reconstructed spatio-angular density. The reconstructed peak orientations are approximately normal to the sphere's surface, although some orientations are not reconstructed well—the “degeneracy” between  $\hat{\mathbf{x}} + \hat{\mathbf{z}}$  and  $\hat{\mathbf{x}} - \hat{\mathbf{z}}$  oriented dipoles is evident in the  $xz$  projection of the peaks. In particular, it's clear that the density is not recovered well for dipoles oriented along  $\hat{\mathbf{x}} - \hat{\mathbf{z}}$  at the points farthest from the cover slip. I suspect that these portions of the object are in the null space of the microscope, but I am still investigating. A GUV phantom will help us understand this issue.

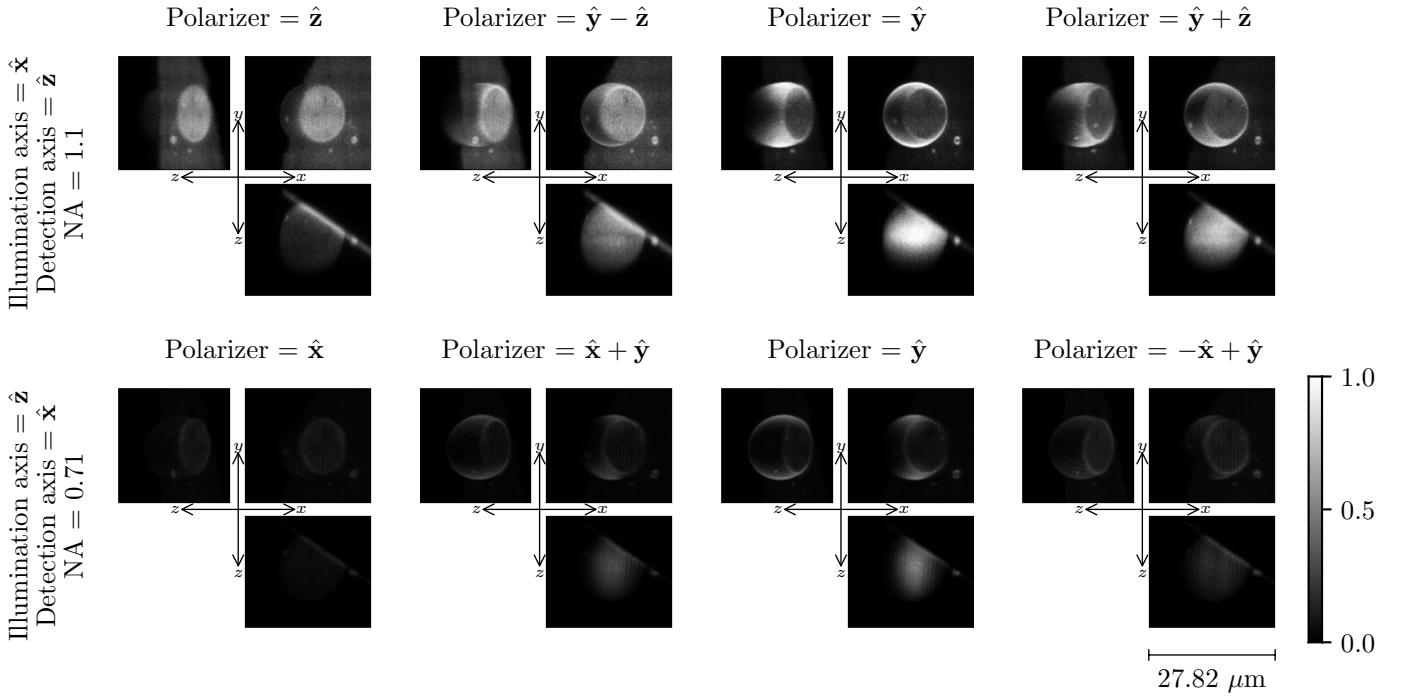


Figure 9: GUV data after corrections.

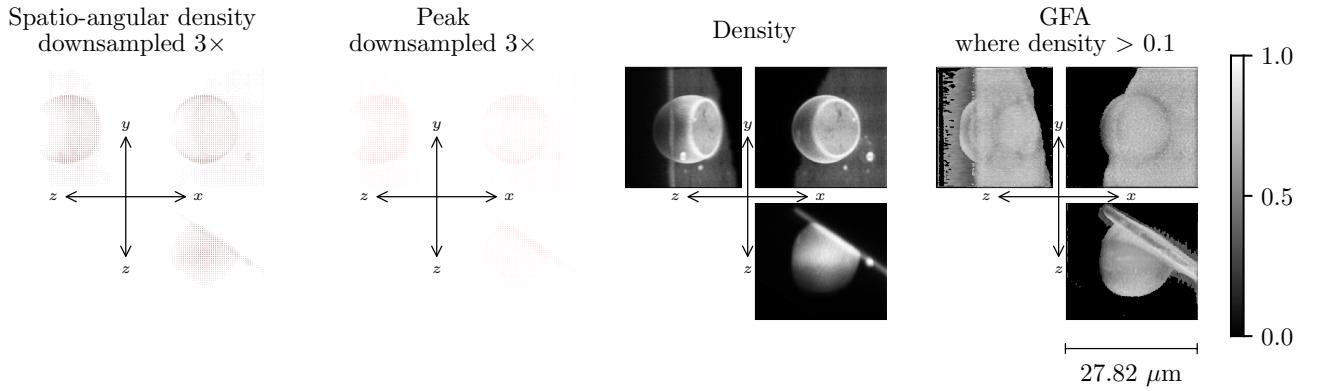


Figure 10: GUV reconstruction with  $\eta = 1$ .

#### 5.4 Xylem study

Tobias prepared a sample of xylem labeled with mScarlet. From two-dimensional fluorescence orientation studies we expect to observe the dipole moments oriented parallel to the helices.

Figure 11 shows the corrected data and Figure 12 shows the reconstructed spatio-angular density. The reconstructed peak orientations are approximately parallel to the helix surface, although once again the “degeneracy” between  $\hat{x} + \hat{z}$  and  $\hat{x} - \hat{z}$  oriented dipoles is evident in the  $xz$  projection of the peaks. The  $yz$  projection of the peak and ODFs is particularly promising—the reconstructed orientations are clearly parallel to the helix around the whole circle of orientations.

One part of the helix near the center of the  $xy$  projection is nearly absent in the data and the reconstruction. It is not clear to me why this region did not give a larger signal in the  $-\hat{x} + \hat{y}$  polarizer data since the helix is parallel to the  $-\hat{x} + \hat{y}$  direction in this region. In the numerical phantom we successfully recovered all of the orientations in the phantom, so it’s not clear to me what is causing this problem. Reconstructing larger volumes may point us in the right direction.

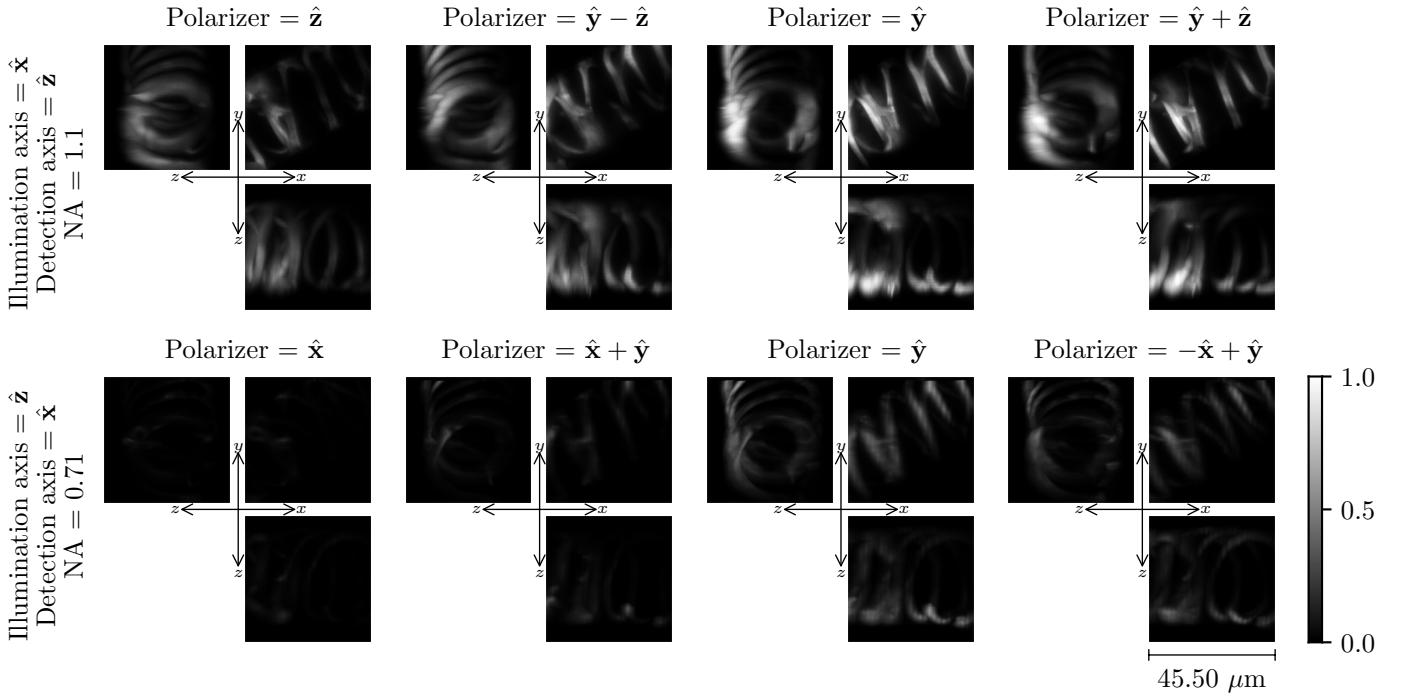


Figure 11: Xylem data after corrections.

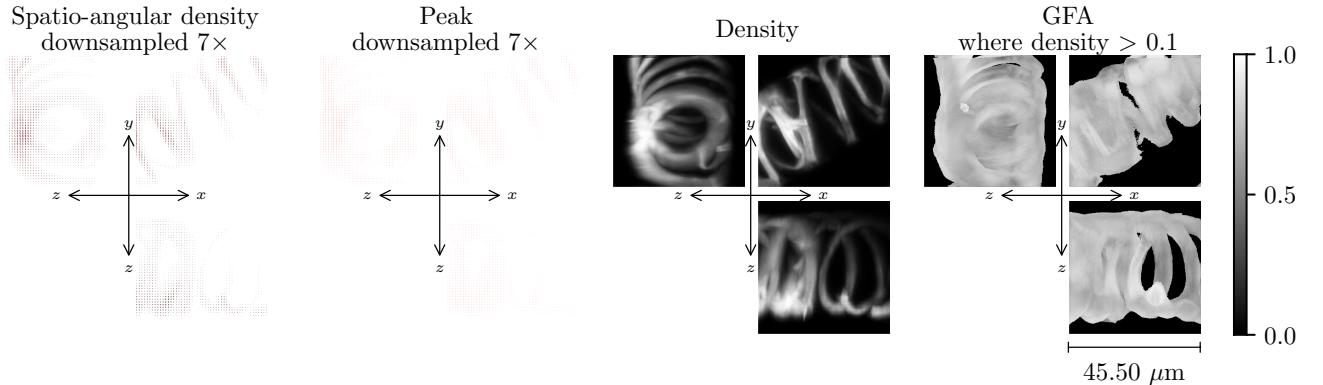


Figure 12: Xylem reconstruction with  $\eta = 1$ .

## 6 Next priorities

Here are some of my next tasks in approximate order of priority. Feedback is always welcome!

- Optimize the summary statistics calculations. These optimizations will make the full  $1000 \times 1000 \times 1000$  voxel reconstructions feasible.
- Directly compare the spatial resolution of the (1) raw data (2) density after spatio-angular reconstruction (3) spatial deconvolution of data averaged over polarizations.
- Continue reconstructing our existing data—the two color data in Gantt’s cells, Bob’s cells from last year, and the remainder of the helix data.
- Draft a theory article for JOSA A or Optics Express. Working title: “Spatio-angular transfer functions for fluorescence microscopes I. Basic theory”.