

# **Coded Illumination Techniques for Phase Imaging and Motion Blur**

Copyright 2016  
by  
Zachary F. Phillips

**Abstract**

Coded Illumination Techniques for Phase Imaging and Motion Blur

by

Zachary F. Phillips

Master of Science in Applied Science and Technology

University of California, Berkeley

Associate Professor Laura Waller, Chair

## 0.1 High-Throughput Imaging

Throughput is a primary concern for many imaging applications - for example, the clinical laboratory a large hospital may need to scan hundreds of histology slides per hour. Often, slide imaging or scanning large samples at high resolution requires the registration and tiling of many images acquired using a high-magnification objective. The acquisition of these datasets is often very long, owing to the precise positioning required as well as a necessary autofocus step prior to each acquisition. Optimization of this process would clearly benefit hospitals and research centers studying datasets which necessitate a large field of view with high resolution.

Most modern high-throughput imaging systems use a stop-and-stare imaging style, where images are acquired by moving the sample to many positions, stopping, autofocusing, and finally acquiring an image. The full-field image is then stitched together using a registration algorithm. As mentioned previously, this method is often prohibitively slow due to the time necessary to stop and start the motion as well as autofocusing. A promising alternative to this scheme is strobed illumination, where the sample is moved constantly while the illumination is strobed, or flashed once each exposure using a very short, high-intensity pulse width. In this framework the image is still blurred, but the blurring is designed to be smaller than the system PSF causing no image degradation.

## 0.2 Motion Deblur

Strobed illumination is an ideal implementation in many applications. However, producing a very bright, very short pulse can often be difficult, particularly when a sample is moving fast, or in a low-resource setting such as a portable device where illumination power and intensity are restricted. A promising alternative to strobed illumination is motion compensation using a deblurring algorithm which incorporates hardware coding techniques. Non-blind motion deblurring was previously explored in the context of computational photography, including several hardware implementations which exploit multiple cameras [1,2] or coded exposures [3]. In addition, blind deblurring for photography has been extensively studied in the context of computational photography, where priors such as object statistics [4–7] and wavelet sparsity [8] to solve for the blur kernel and latent image simultaneously. In general, in-plane motion blur can be modeled as the convolution of a static intensity image ( $I_s$ ) with a blur kernel  $B$ , which defines the smearing of a single point as it is moved across the field of view during a single exposure:

$$I_{blurred}(\vec{r}) = I_0(\vec{r}) * B(\vec{r}) \quad (1)$$

Here  $\vec{r}$  and  $\vec{k}$  represent the spatial coordinates and the Fourier conjugate variables for k-space respectively, and  $I_0$  is the unblurred image of the sample or scene. In this simple case, we assume complete knowledge of the blur kernel  $B$ . The inverse problem can be formulated as a simple deconvolution of the static intensity image:

$$\tilde{I}_0(\vec{r}) = \mathcal{F}_{2D}^{-1} \left\{ \frac{I_{\text{blurred}}}{\tilde{B}(\vec{k})} \right\} \quad (2)$$

Where  $\tilde{\cdot}$  denotes the 2D Fourier Transform. This inverse problem is a standard deconvolution problem and can be solved using a single least squares operation. Like most deconvolution problems, this problem is ill-posed and requires regularization to solve due to small values or zeros in the spectrum of  $B$ . This regularizer can be tailored to the problem using a-priori information about the sample, but in general a  $\ell_2$  regularizer provides adequate results, and allows a single-step solution for a general object.

In many cases we not only know  $B$  a priori, but have complete control over the shape of  $B$  subject to practical hardware constraints. We can, therefore, choose an optimal motion pathway as well as illumination or exposure magnitude before acquiring to enhance the quality of our reconstructions. In the case where exposure is not coded,  $B$  is simply a 1D *rect* function placed into a 2D image as shown in Fig. 0.1. In the Fourier domain, this transforms to a *sinc* function, which has many zero crossings, making it hard to invert even if  $B$  is perfectly known. The problem of designing an optimal  $B$  motivated the design of the flutter-shutter camera [3]. In this design, a ferrite shutter was placed in front of a camera lens and was opened and closed repeatedly during a single camera exposure. As a sample moves across the field it is convolved with this shutter time series, mapped to the spatial dimension by the velocity of the object, which is assumed to be known. This allows the user to optimize the blur kernel based on the object's velocity and exposure time.

A common metric for the fitness of a particular choice of  $B$  is the condition number  $\kappa$ . Generally, the condition number is the ratio of the largest to smallest singular value of the filter  $H$ . For linear shift-invariant (LSI) systems these are equal to the absolute value of the maximum and minimum Fourier coefficients respectively:

$$\kappa\{B\} = \frac{\sigma_{\max}\{B\}}{\sigma_{\min}\{B\}} = \|B\|_p \|B^{-1}\|_p = \frac{\max |\tilde{B}(\vec{k})|}{\min |\tilde{B}(\vec{k})|} \quad (3)$$

This metric can be evaluated quickly which makes optimizing  $B$  directly numerically feasible. Essentially, this metric promotes a flat Fourier spectrum of  $B$  and penalizes values close to zero, which makes the problem well-posed. It is important to note that the optimum solution of  $B$  is clearly a delta function  $\delta(\vec{r})$ , and this optimization will converge to this solution if left unconstrained. However, a delta function corresponds to strobed illumination, which provides much less overall exposure, and therefore a much noisier image compared to constant exposure. To counter this, we place a box constraint on the maximum intensity of the illumination, which has practical motivations based on the maximum illumination intensity. In practice we bound  $B \in [0, 1]$ . In addition we place a constraint on the total illumination throughput of the system during an exposure time,  $\sum_{\vec{r}} B(\vec{r}) \geq \gamma$ , where  $\gamma$  is typically set to 50% of the maximum intensity to allow the algorithm reasonable freedom as in [3]. The complete problem is expressed as:

$$\begin{aligned}
& \underset{B}{\text{minimize}} \quad \kappa\{B\} = \frac{\max |\tilde{B}(\vec{k})|}{\min |\tilde{B}(\vec{k})|} \\
& \text{subject to} \quad \sum_{n=1}^N B[n] \geq \gamma N, \quad 0 \leq B[n] \leq 1 \quad \forall n
\end{aligned} \tag{4}$$

Figure 0.1 shows example blur kernels with their representations in real space and Fourier space. Noise also plays an important role, as noise will corrupt spatial frequencies which have poor transmission, further motivating the design of kernels with a flat Fourier spectrum. This method was also applied to microscopy using programmable LED array illumination which allows motion deblurring by varying the intensities of the LED array during exposure times [9]. Unlike the ferro-electric shutter used in [3], the LED array has no moving parts, which allows faster update speeds, and can provide coded multi-color illumination if tri-color LEDs are used.

### 0.3 Phase Imaging with Motion Deblur

Phase Imaging using the Weak Object Transfer Function (WOTF) is highly compatible with motion deblur since both are modeled as linear convolutions on the same object. Chronologically, the results presented in Chapter 2 originated from the motion deblur problem, since we desired a method of recovering a blurred phase object from a single image. Recovering phase from multiple exposures is not considered in this chapter, but could be incorporated in future work. Our forward model in the frequency domain is a combination of the existing motion deblur model with the WOA. In the case of a single image, we model the blurred Intensity image as two separate convolutions, applied sequentially:

$$\tilde{I} = B * [H_\mu * A + H_\phi * \phi] \tag{5}$$

We can express the above equation as a block-wise matrix product in the Fourier domain, letting  $\tilde{B}$ ,  $\tilde{H}_\mu$ , and  $\tilde{H}_\phi$  be the diagonalized Fourier Transforms of the transform functions, and  $\tilde{I}$ ,  $\tilde{\mu}$ ,  $\tilde{\phi}$  be the vectorized image and object components respectively:

$$\tilde{I} = \tilde{B} \cdot [\tilde{H}_\mu \quad \tilde{H}_\phi] \begin{bmatrix} \tilde{\mu} \\ \tilde{\phi} \end{bmatrix} \tag{6}$$

The WOTF equations, which were defined in Chapter 2, are provided here again for a given illumination wavelength  $\lambda$ :

$$\tilde{H}_\mu(\vec{f}, \lambda) = \left[ P(\vec{f}, \lambda) \star (P(\vec{f}, \lambda) \cdot S(-\vec{f}, \lambda)) + (P(\vec{f}, \lambda) \cdot S(-\vec{f}, \lambda)) \star P(\vec{f}, \lambda) \right] \tag{7}$$

$$\tilde{H}_\phi(\vec{f}, \lambda) = \frac{\lambda_0}{\lambda} \cdot \left[ P(\vec{f}, \lambda) \star (P(\vec{f}, \lambda) \cdot S(-\vec{f}, \lambda)) - (P(\vec{f}, \lambda) \cdot S(-\vec{f}, \lambda)) \star P(\vec{f}, \lambda) \right], \tag{8}$$

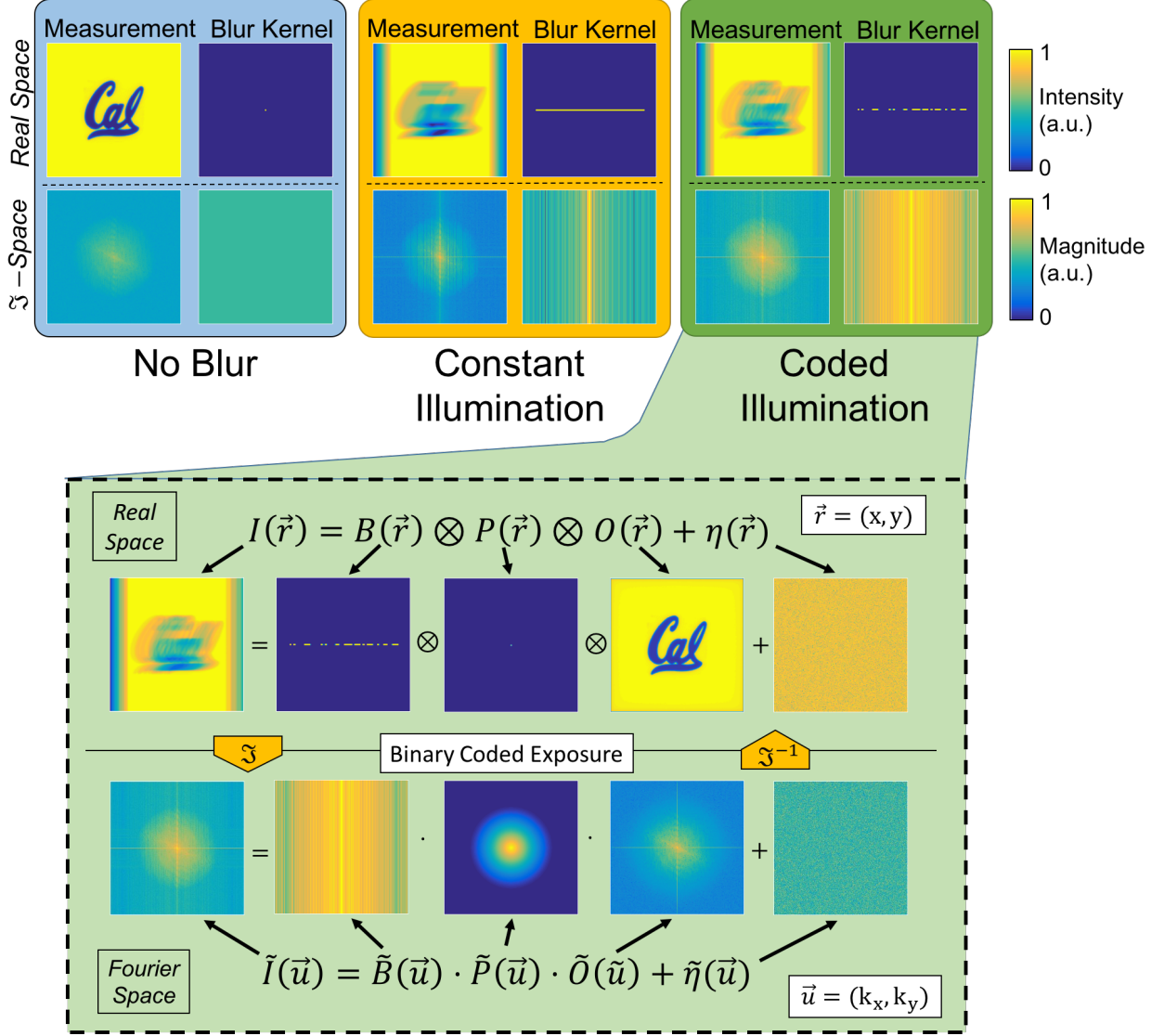


Figure 0.1: Motion blur real-space and Fourier space representations. In the case of a static object, the object is filtered by the optical transfer function (OTF), assuming incoherent illumination. With motion blur, an additional convolution with a blurring function  $B(\vec{r})$  causes attenuation in the frequency domain which acts as a cascaded linear filter on the object. In the case of coded blur, the transfer function is attenuated much less than in the conventional blur case. The colorbars are constant for real and fourier domain images respectively.

Combining measurements from the three color channels, we model the full over-determined system as:

$$\begin{bmatrix} \tilde{I}_R \\ \tilde{I}_G \\ \tilde{I}_B \end{bmatrix} = \begin{bmatrix} \tilde{B}_R & 0 & 0 \\ 0 & \tilde{B}_G & 0 \\ 0 & 0 & \tilde{B}_B \end{bmatrix} \times \begin{bmatrix} \tilde{H}_{\mu,R} & \tilde{H}_{\phi,R} \\ \tilde{H}_{\mu,G} & \tilde{H}_{\phi,G} \\ \tilde{H}_{\mu,B} & \tilde{H}_{\phi,B} \end{bmatrix} \times \begin{bmatrix} \tilde{\mu} \\ \tilde{\phi} \end{bmatrix} \quad (9)$$

As in [3] and [9], we design our patterns such that the condition number of the blur kernel is minimized. By combining both motion deblurring and the linearized phase retrieval technique described in Chapter 2 (Eq. ??, ??), we can use knowledge of our WOTF as derived previously to influence our choice of  $B$  to improve our overall phase and amplitude reconstruction from blurred data. The motion deblur problem as presented here will always degrade the result even with an ideal  $B$  due to the constraints placed on the optimization problem. In the previous case, degradation due to the blurring operation was minimized by solving for a blur kernel with an optimally flat Fourier spectrum. This method, however, did not take into account the additional attenuation due to the OTF of the optical system.

Our improvement to the methods described in [3] and [9] is to consider the spectrums of cascaded filters in our system when designing our blur kernel. We can think of our sequential deconvolution problem as a single-step deconvolution which inverts the element-wise product of the blur kernel and WOTF's in the Fourier Domain. Therefore, the relative attenuation produced by the blur kernel at each frequency can be matched to reduce the degradation to highly attenuated frequencies in the WOTF, such as high frequencies in both amplitude and phase WOTF's, as well as low frequencies in the phase WOTF. The exact structure of this transfer function depends on the pupil function of the optical system and design of the source. In practice, we note that the phase transfer function is of higher order than the amplitude transfer function (OTF), which generally means there are more zero crossings and values close to zero in the phase transfer function. Therefore, we will use the phase transfer function for optimizing the blur kernel.

To solve for the optimal blur kernel considering the WOTF, we modify Eq. 4 to consider a 1D spectral reference  $q$ , which provides a measure of the attenuation imposed by the optical system at each spatial frequency in the blur kernel.

$$\begin{aligned} & \underset{B}{\text{minimize}} \quad \frac{\max |\tilde{B} \cdot \tilde{q}|}{\min |\tilde{B} \cdot \tilde{q}|} \\ & \text{subject to} \quad \sum_{n=1}^N B[n] \geq \gamma N, \quad 0 \leq B[n] \leq 1 \quad \forall n \end{aligned} \quad (10)$$

For linear kernels, we chose  $q$  to be the sum of the magnitude of the phase transfer function along the direction orthogonal to the blur direction as shown in Fig. 0.2. This quantity can be thought of as a penalty function for attenuating each spatial frequency during the blurring process - if the OTF is already very low at a given frequency, the blur kernel should try not to attenuate this frequency significantly. Since the blurring filter is applied in hardware,

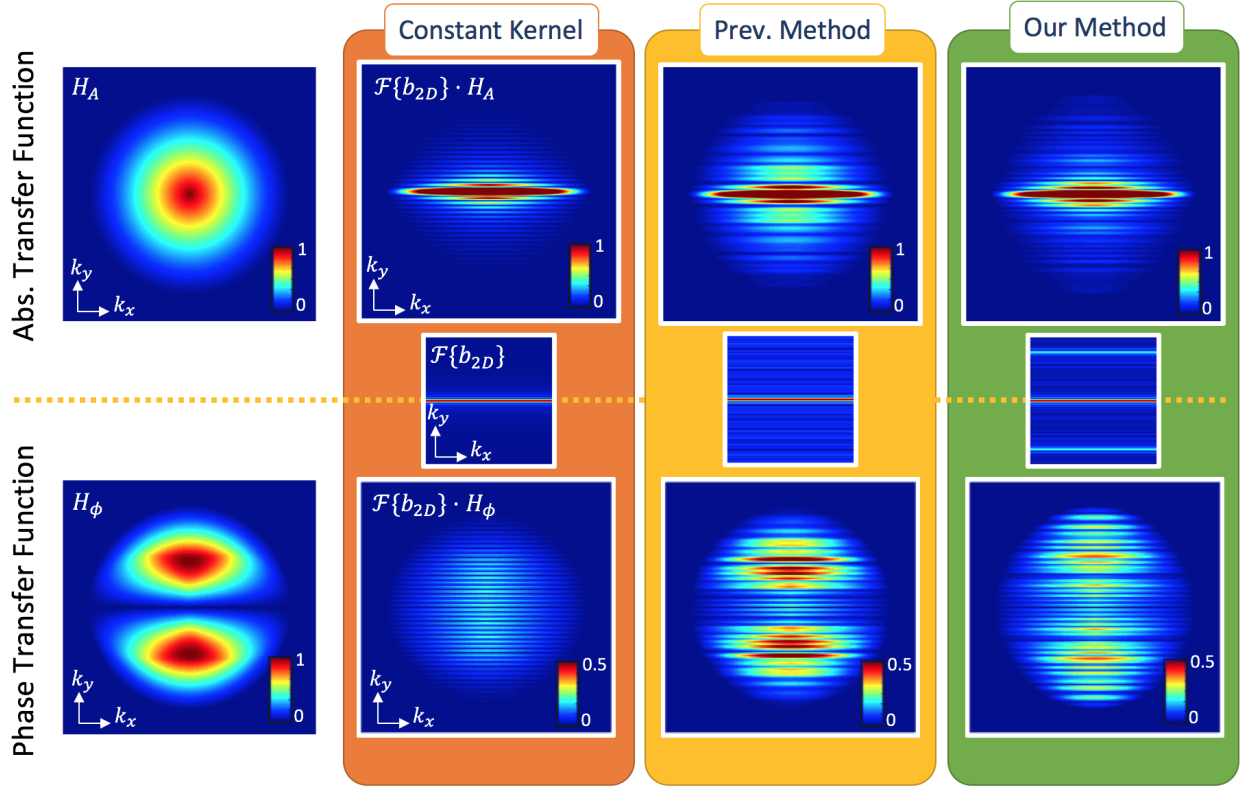


Figure 0.2: Motion blurring kernels for 1D motion in the Fourier Domain. **Left Column:** Unblurred absorption (top) and phase (bottom) transfer functions. **2<sup>nd</sup> Column:** Middle: Fourier transform of constant (non-coded) blur kernel. The top and bottom spectra in this column are the product of the unblurred transfer function and this center spectrum. **3<sup>rd</sup> Column:** Blurred transfer function spectra for temporal coding which does not consider WOTF structure. **4<sup>th</sup> Column:** Blurred transfer function spectra for temporal coding which emphasizes low and high frequencies based on WOTF structure.

this process improves noise performance for frequencies which are heavily attenuated by the optical system. It is important to note that this method will never improve resolution or noise performance beyond the static solution - however, it can greatly reduce the degradation due to the blurring process, making deblurring practical for high-speed quantitative imaging applications.



## 0.4 Validation

### Simulation

Simulation results are shown in Fig. 0.3. Here we show a static sample, deblurring without coded illumination, deblurring with the previous method (no spectral reference), and deblurring using our method, which accounts for the optical system transfer function. In this simulation we note that the normalized sum-squared error (N-SSE) in phase is reduced significantly in our method. The amplitude N-SSE did increase slightly using our method, which is likely due to the fact that we used the phase WOTF for generating our blur kernels instead of amplitude. The choice of which WOTF to use could be application-dependent.

### Experimental Results

Our system consists of a commercial Nikon AZ100 microscope using a  $1 \times 0.10$  NA objective, a Bayer-patterned SCMOS camera (PCO.edge 5.5), an XY stage with linear encoders, (Prior H117), and illumination from 23 multi-channel LEDs (Jameco 2128500) arranged in a hexagonal pattern using a laser-cut holder for positioning. The LEDs were placed approximately 160mm from the sample to match the spacing such that the outer LEDs illuminate the sample from angles just inside of the NA of the microscope. This is done to ensure maximum phase contrast and bandwidth (resolution) of the system. The LEDs are controlled using a Teensy 3.2 microcontroller, which can be dynamically programmed. Camera exposures, stage movement, and illumination modulation are controlled using open-looped feedback with 5ms synchronization resolution, limited by the update speed of the LED array. This system is shown in Fig. 0.4.

Our forward model considers the case where our LED illumination is incoherent and discrete, both spatially and temporally. We assume each emitter has three coincident emitters for Red ( $\bar{\lambda} = 625$ ), Green ( $\bar{\lambda} = 525$ ), and Blue ( $\bar{\lambda} = 470$ ), wavelengths, which propagate through the optical system independently of each other and are detected separately by the bayer filter of our color camera. We assume a sample which is non-dispersive and unstained. A velocity of 25mm per second was used for sample movement, but this could be increased by improving hardware synchronization. Blur kernels were calculated using the calibrated phase WOTFs for each color channel separately, considering the spacing of k-space due to wavelength.

Experimental reconstructions are shown in Fig. 0.5. To test our method, we used a micro-lens array (Fresnel-Tech 605) as our sample due to its well-defined geometry. Fig. 0.5 shows reconstructions for the static case, previous method, and our method. While the sample amplitude is relatively unchanged, the phase reconstructions clearly show that accounting for the spectral reference provides better results than optimizing the blur kernel alone. This supports our claim that image degradation from blurring can be reduced significantly, but not eliminated, using our method.

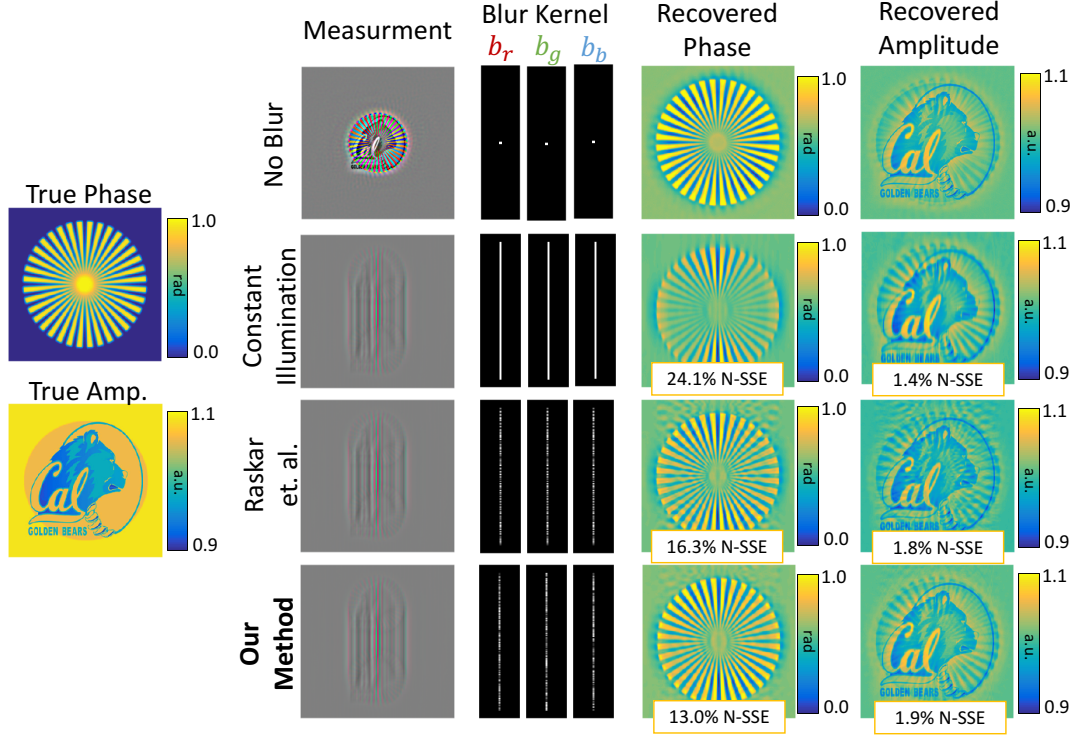


Figure 0.3: Simulation results for phase retrieval from blurred images. **Top Row:** Phase and amplitude retrieval of a stationary sample. This serves as our best-case reconstruction for the blurred images. **Second Row:** Phase and amplitude retrieval of a blurred sample using constant illumination. This is the degenerate solution. **Third Row:** Deblur using a coded blur kernel as in the previous methods. The result improves significantly over the constant illumination case, but is still missing features in phase. **Fourth Row:** Deblur using our coded kernels

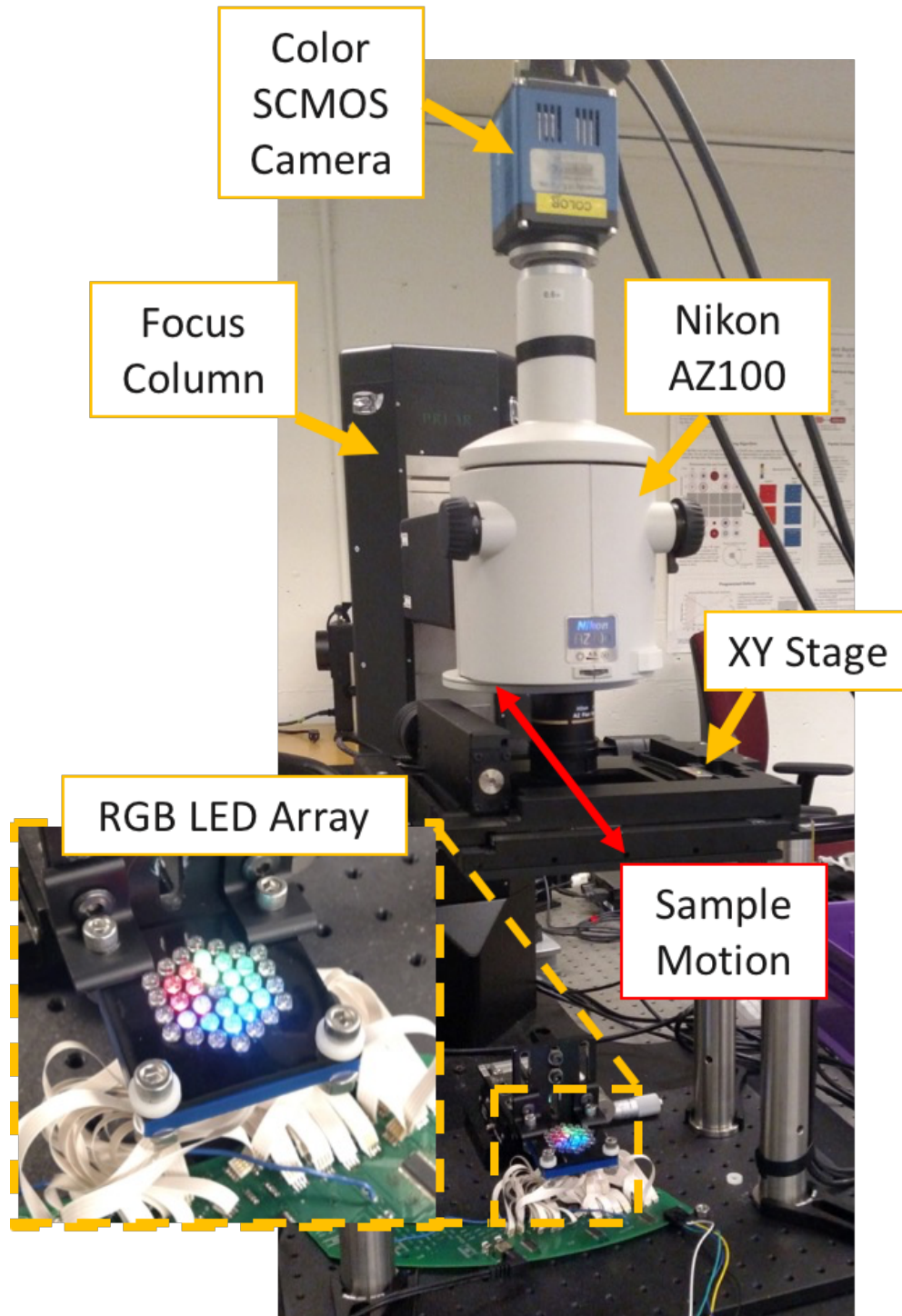


Figure 0.4: Experimental setup for motion deblurring. The tri-color LED array provides fast illumination coding in time, angle, and color. The LED array can be programmed dynamically using a Teensy micro-controller and has a maximum update speed of approximately 250 Hz. The LED array is synchronized with the linear motion stage and camera in hardware to provide precise system synchronization.

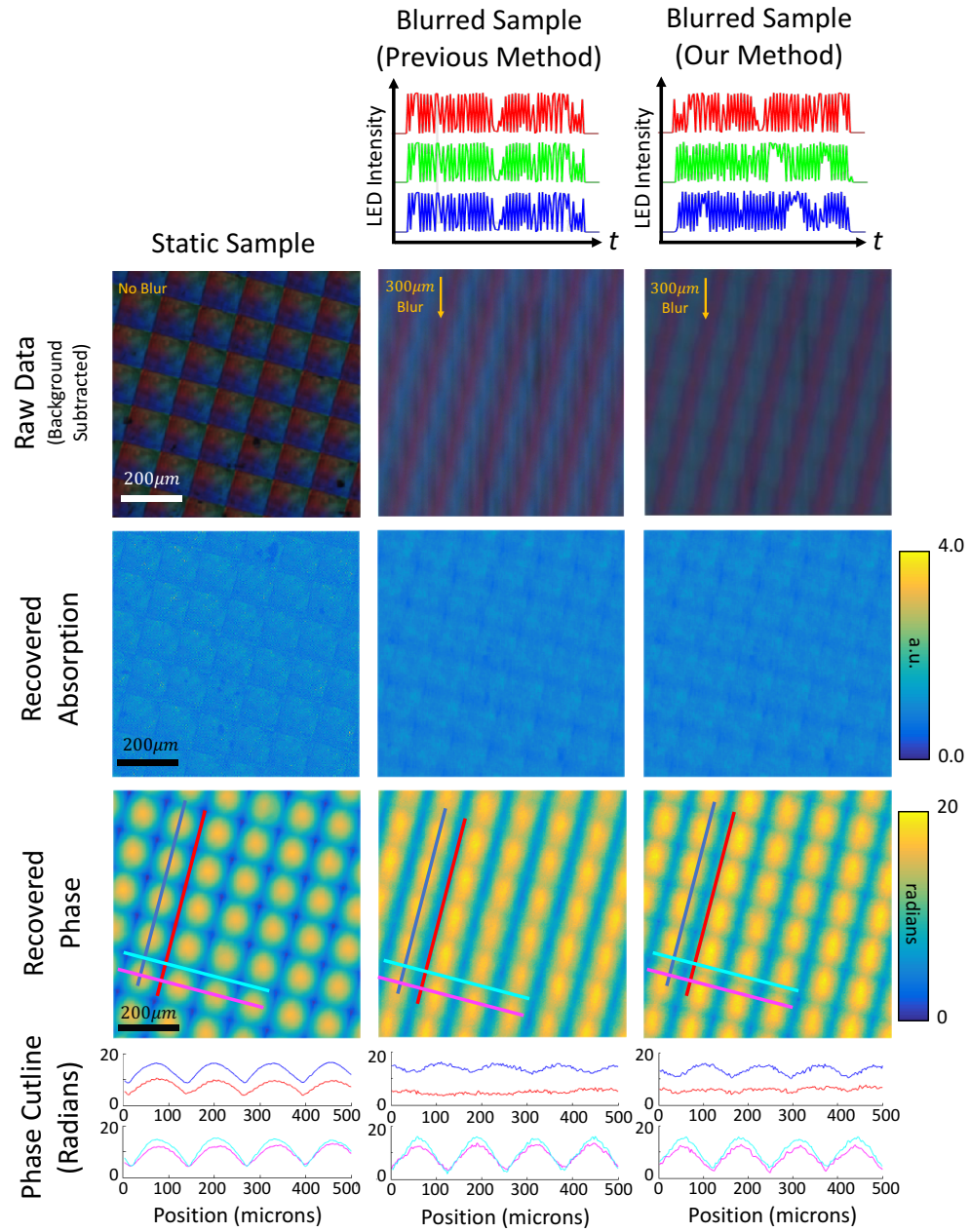


Figure 0.5: Experimental results for motion deblur of a moving sample (Fresnel Technologies 605 micro-lens array).

# Bibliography

- [1] Nayar SK, Ben-Ezra M (2004) Motion-based motion deblurring. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26: 689-698.
- [2] Li F, Yu J, Chai J (2008) A hybrid camera for motion deblurring and depth map super-resolution. In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. pp. 1-8. doi:10.1109/CVPR.2008.4587574.
- [3] Raskar R, Agrawal A, Tumblin J (2006) Coded exposure photography: motion deblurring using fluttered shutter. *ACM Transactions on Graphics (TOG)* 25: 795–804.
- [4] Fergus R, Singh B, Hertzmann A, Roweis ST, Freeman WT (2006) Removing camera shake from a single photograph. In: *ACM Transactions on Graphics (TOG)*. ACM, volume 25, pp. 787–794.
- [5] Shan Q, Jia J, Agarwala A (2008) High-quality motion deblurring from a single image. *ACM Trans Graph* 27: 73:1–73:10.
- [6] Cho S, Lee S (2009) Fast motion deblurring. *ACM Trans Graph* 28: 145:1–145:8.
- [7] Levin A (2006) Blind motion deblurring using image statistics. In: *Advances in Neural Information Processing Systems*. pp. 841–848.
- [8] Cai JF, Ji H, Liu C, Shen Z (2012) Framelet-based blind motion deblurring from a single image. *IEEE Transactions on Image Processing* 21: 562-572.
- [9] Ma C, Liu Z, Tian L, Dai Q, Waller L (2015) Motion deblurring with temporally coded illumination in an led array microscope. *Opt Lett* 40: 2281–2284.