Axial super resolution via multiangle TIRF microscopy with sequential imaging and photobleaching

Yan Fu[a], Peter W. Winter[b], Raul Rojas[c], Victor Wang[d], Matthew McAuliffe[d], and George H. Patterson[a,1]

[a] Section on Biophotonics, National Institute of Biomedical Imaging and Bioengineering, National Institutes of Health, Bethesda, MD 20892

[b] Section on High Resolution Optical Imaging, National Institute of Biomedical Imaging and Bioengineering, National Institutes of Health, Bethesda, MD 20892

[c] Section on Biological Chemistry, National Institute of Dental and Craniofacial Research, National Institutes of Health, Bethesda, MD 20892

[d] Biomedical Imaging Research Services Section, Division of Computational Bioscience, Center for Information Technology, National Institutes of Health, Bethesda, MD 20892

[1] Corresponding author
George H. Patterson
9000 Rockville Pike
Building 13, Room 3E33
National Institutes of Health
Bethesda, MD 20892
Phone: 301-443-0241
Email: pattersg@mail.nih.gov

SUPPLEMENTARY INFORMATION

The multi-direction illumination pattern is produced by moving the incident beam in a circular pattern in the rear aperture of the objective lens using a two-axis galvanometer (Fig. S1). The incident beam angle is adjusted by changing the diameter of the circular pattern at the back focal plane by altering the amplitudes of two sine wave outputs from a waveform generator driving the 2D galvanometer. To test and calibrate the microscope, we determined the relationship between the voltages applied to the 2D galvanometer and the angle produced by measuring the radius of the circular pattern out of the objective (See **Calculation of incident beam angles** and **Fig. S2*A***).

1. Calculation of incident beam angles

In the MA-TIRF system, the angles of the incident beam were adjusted by controlling the amplitudes of the two sine wave outputs from the waveform generator. The relationship between the angle of the beam out of the objective and the amplitudes of the sine wave output can be estimated by the following equations according to the configuration of the setup in **Fig. S1** and **Fig. S2*A***.

The radius, $\mathbf{r_1}$, of the circular pattern located at the scan lens (focal length $\mathbf{f_1}$ = 50mm) immediately after the galvanometer is determined by the angle, $\boldsymbol{\theta_{galvo}}$, and $\mathbf{f_1}$.

$$\mathbf{r_1 = f_1 \tan\theta_{galvo}} \quad (1)$$

The angles in the 2X expander, $\boldsymbol{\theta_{ex1}}$ and $\boldsymbol{\theta_{ex2}}$, are equivalent and dependent on $\mathbf{r_1}$ and the focal lengths, $\mathbf{f_2}$ and $\mathbf{f_3}$.

$$\mathbf{\tan\theta_{ex1} = r_1 / f_2} \quad (2)$$

$$\mathbf{\tan\theta_{ex2} = r_2 / f_3} \quad (3)$$

$$\mathbf{\theta_{ex2} = \theta_{ex1}} \quad (4)$$

$$\mathbf{r_2 = r_1 f_3 / f_2} \quad (5)$$

$$\mathbf{r_2 = \tan\theta_{galvo} f_1 f_3 / f_2} \quad (6)$$

The variable **V** is the amplitude of the sine wave output from the waveform generator. The galvanometer system was set to 1 **V** = 1 degree of movement of the mirror and 2 degrees movement of the laser beam to give the following relationship

$$\mathbf{r_2 = \tan(2V)\, f_1 f_3 / f_2} \quad (7)$$

The angle of incidence ($\boldsymbol{\theta_{Obj}}$) is dependent on the index of refraction ($\mathbf{n_{Oil}}$=1.5168, $\mathbf{n_{Air}}$=1.000277), the objective focal length ($\mathbf{f_{Obj\ Oil}}$), the numerical aperture of the

objective ($\mathbf{NA_{Obj\ Oil}}$), and is proportional to the position of the beam at the rear aperture of the objective ($\mathbf{r_2}$).

$$\mathbf{\sin\theta_{Obj\ Oil} \propto f_1 f_3 \tan(2V)/n_{Oil} f_{Obj} f_2} \quad (8)$$

A cell culture dish with radial marks to indicate the distance off center was placed on the microscope stage. The radius of the circulating beam, $\mathbf{r_3}$, is the off center distance of the beam measured on the marked cell culture dish above the objective. The distance between the top of the dish and the objective is represented by **h** and the incident beam angle ($\mathbf{\theta_{Obj\ Air}}$) exiting the objective was determined. Since $\mathbf{f_{Obj\ Air}}$ <<< **h**, the following approximation was used.

$$\mathbf{\tan\theta_{Obj\ Air} = r_3/h} \quad (9)$$

Using Snell's law, $\mathbf{\theta_{Obj\ Air}}$ was converted into $\mathbf{\theta_{Obj\ Oil}}$.

$$\mathbf{n_{oil} \sin\theta_{Obj\ Oil} = n_{Air} \sin\theta_{Obj\ Air}} \quad (10)$$

and used to determine the angle with oil.

$$\mathbf{n_{oil} \sin\theta_{Obj\ Oil} = n_{Air} \sin\left(\tan^{-1}(r_3/h)\right)} \quad (11)$$

The determined $\mathbf{\sin\theta_{Obj\ Oil}}$ was found to be linearly proportional to **tan2V** and fit well with the equation:

$$\mathbf{\sin\theta_{oil} = -0.00153 + 35.5\ (\tan 2V)} \quad (12)$$

Since $\mathbf{n_{glass} \sin\theta_c = n_{water}}$, the critical angle $\mathbf{\theta_c}$ was calculated to be 61.5 degree (n_{glass}=1.5168, n_{water}=1.3333). Extending the curve in **Fig. S2*B***, the calculated critical angle is obtained at 0.7099 volts sine wave amplitude.

As described above, the slope of the fitting curve depends on the focal length of each lens and also the distance between the mirror and the first lens. Because the distances between the two galvanometer mirrors (x axis and y axis) and the first lens were slightly different (~5-10 mm), the beam exiting from the objective was not perfectly circular, especially at large incident angles (≥ critical angle). Therefore, the amplitudes of the outputs or phase difference between the sine waves can be adjusted to compensate for this difference. On our machine, this was accomplished simply by changing the phase difference between the sine wave outputs of the waveform generator from 90° to 89.2°.

2. Experimental determination of the critical angle

A drawback to our approach is that the calibration was extrapolated to angles above the critical angle (**Fig. S2*B***). Since the relationship between incident beam angles and sine wave parameters was critical for the use of this machine, further tests were performed by experimental determination of the critical angle.

The critical angle was calculated to be 61.5° according to Snell's law and using the equation $\mathbf{\sin\theta_{oil} = -0.00153 + 35.5(\tan 2V)}$, the amplitude of the sine wave corresponding to the critical angle was calculated to be 0.7099 V. Critical angle determination by experimentation relied on a phenomenon described previously (1) in which the intensity at the glass-water interface of the incident light can be 4-5 times greater at the critical angle than the transmitted light below the critical angle. A monolayer of His-tagged EGFP was prepared by binding to a monoclonal His tag antibody coated coverslip and then imaged at increasing angles to use the fluorescence intensity as the readout of the excitation light intensity. As the angles increased, the fluorescence intensity of the image increased to a maximum and then decreased (**Fig. S2*C***). Given that the signal is all likely derived within the first 10-20 nm in axial dimension from the interface with the coverslip, the axial dependence on collection of emission due to nearfield effects previously discussed (1, 2) is unlikely to play a major role in the level of the measured fluorescence intensity. Therefore, fluorescence intensity should reflect the transmitted light intensity when the incident beam is below the critical angle and the evanescent field intensity when the incident beam is equal to or above the critical angle (1). The fluorescence intensity of monolayer EGFP was normalized by the fluorescence of bulk FITC solution for transmitted light intensity and is plotted versus incident beam angle in **Fig. S2*C***. The peak found at 0.7120 V is in good agreement with the calculated value of 0.7099 V and indicates adequate calibration of multiangle TIRF control, but the equation discussed earlier was modified to account for this small difference $\mathbf{\sin\theta_{oil} = -0.00153 + 35.5\ (\tan 2(V-0.0021))}$. It is noteworthy that the emission intensity-angle profile found here was similar to the theoretical profiles for the excitation intensity-angle profile previously published (1).

3. Experimental determination of penetration depth

To build the relationship between incident beam angles ($\boldsymbol{\theta}$) and penetration depths (**d**), silica beads either fluorescently labeled or bathed in a fluorescent medium were imaged. Using the radius of the bead signal, the bead dimensions, and simple geometry calculations, the imaging depths (**z**) at multiple incident beam angles were determined (**Fig. S3**). The penetration depth (**d**) was calculated by fitting the fluorescence intensity profile along a line drawn from the center of the bead to the periphery with a single exponential decay equation.

$$\mathbf{F_\theta = I_{0,\theta}\, e^{-z/d}} \qquad (13)$$

We note here that we did not observe the slow exponential due to scattering when we performed our calibrations as reported previously(3). At the onset of our studies, this inconsistency with published reports led us to a second strategy utilizing unlabeled silica beads in a solution of fluorescein isothiocyanate (FITC). In these images, beads are observed as dark objects surrounded by fluorescence signal (**Fig. S3D-E**). To determine TIRF depth by imaging the beads in FITC solution, we assumed a homogeneous distribution of fluorescence and used the following equation.

$$F_\theta = \int_0^Z I_{0,\theta}\, C\, e^{-z/d}\, dz = I_{0,\theta}\, C\, d\, (1 - e^{-z/d}) \quad (14)$$

$I_{0,\theta}$ is the intensity on the coverslip surface and F_θ is the total intensity detected in the camera at the incident beam angle θ. **C** is the concentration of the FITC in the solution. The penetration depth (**d**) was calculated by fitting the fluorescence intensity profile along the line across the center of the bead using the equation above. The penetration depths determined by the two methods were similar and consistent with theoretical calculation. With the exception of the 300 nm depth measurements, all results differed by less than 10% from the theoretically calculated values based on equation

$$d=\lambda_0/4\pi(n_1^2\sin^2\theta_1 - n_2^2)^{-1/2} \quad (15)$$

Furthermore, measurements made in a 60% glycerol solution with a refractive index similar to that of the silica beads were also consistent with theoretical values (**Fig. S3F**). Therefore, the calculated penetration depths using n_1=1.5168 and n_2=1.3333 were used to report the incident beam angles used in the imaging experiments in this work.

4. Image collection and analysis

The intensity of the TIRF evanescent field decays exponentially with distance **z** away from the glass-water interface

$$I_z = I_0\, e^{-z/d} \quad (16)$$

where

$$d = \frac{\lambda}{4\pi\, n_1} (\sin^2\theta - \sin^2\theta_c)^{-1/2} \quad (15)$$

I_0 is the intensity at the interface, n_1 is the refractive index of the coverslip and θ_c is the critical angle. While the exponential decay of the evanescent wave provides the key advantage for TIRF microscopy and makes possible the technique described here, the lack of a discrete boundary of irradiation requires some effort in determining the proper level of photobleaching as well as incident excitation power at each TIRF depth. Although some arbitrary trial and error experiments were initially performed to optimize this parameter, the following describes our rationale using the calculated TIRF depth at each layer to guide the photobleaching protocol.

In this example, the sample is treated as a uniform structure in the axial direction which has been arbitrarily divided into different layers. Each layer results from the difference between two penetration depths d_1 and d_2 that correspond to two incident angles θ_1 and θ_2, respectively. Two images are collected in each layer with the first image collected immediately after movement to a new incident beam angle designated as the pre-bleach image. A second image is collected after irradiation of the sample to decrease the fluorescence through photobleaching and

is designated as the post-bleach image. The final image of each layer is obtained by subtracting the post-bleach image from the pre-bleach image at each incident angle. For these estimations, the fluorophore concentration is assumed to be constant **C** and uniformly distributed in all layers. The fluorescence $\mathbf{F}_\theta$ generated from the sample from interface to the axial position **z** as a function of the depth **d**:

$$\mathbf{F}_\theta = \int_0^Z \mathbf{I}_{0,\theta}\, \mathbf{C}\, \mathbf{e}^{-z/d}\, dz = \mathbf{I}_{0,\theta}\, \mathbf{C}\, \mathbf{d}\, (1 - \mathbf{e}^{-z/d}) \qquad (14)$$

$\mathbf{I}_{0,\theta}$ is the incident beam intensity at the interface which is dependent on the incident angle and polarization of the incident beam (see ref. (4)). For the **s**-polarized incident beam,

$$\mathbf{I}^{S}_{0,\theta} = \frac{|A_S|^2\, 4\cos^2\theta}{1-(n_2^2/n_1^2)} \qquad (17)$$

and for the **p**-polarized incident beam,

$$\mathbf{I}^{P}_{0,\theta} = |A_P|^2 \frac{4\cos^2\theta(2\sin^2\theta-(n_2^2/n_1^2))}{(n_2/n_1)^4\cos^2\theta+\sin^2\theta-(n_2^2/n_1^2)} \qquad (18)$$

where $\mathbf{n}_1$ and $\mathbf{n}_2$ are refractive indices of coverslip and sample medium, respectively and $\mathbf{A}_s$ and $\mathbf{A}_p$ are the electric field amplitudes of the **s**-polarized and **p**-polarized incident beams, respectively. In this example, the incident light and $\mathbf{I}_0$ are not discussed separately in terms of ***s*** and ***p*** polarizations since the time-averaged images were collected from the sample being excited from 360 degrees. While the polarization of the laser beam remains constant, the polarization of the incident beam with respect to the direction of light propagation changes repeatedly between ***s*** and ***p*** with a gradient of mixed ***s*** and ***p*** polarizations as the beam is moved around the periphery of the objective rear aperture. Thus, the incident light has a range of polarizations for each image.

For a specific angle that corresponds to a penetration depth $\mathbf{d}_1$ and assuming infinite axial depth **z**, the fluorescence intensity detected by the camera from all layers of the sample is

$$\mathbf{F}_{total,\theta 1} = \mathbf{I}_{0,\theta 1}\, \mathbf{C}\, \mathbf{d}_1 \qquad (19)$$

and represents the pre-bleach image in the first layer. With a penetration depth $\mathbf{d}_1$, the fluorescence generated from the coverslip interface **0** to axial position $\mathbf{z}_1$, where $\mathbf{z}_1 = \mathbf{d}_1$,

$$F_{d1,\theta1} = I_{0,\theta1}\, C\, d_1\, (1 - e^{-1}) \tag{20}$$

Therefore, the fluorescence to penetration depth d_1 compared to total is

$$\frac{F_{d1,\theta1}}{F_{total,\theta1}} = 1 - e^{-1} = 63.2\% \tag{21}$$

Although the image contains the fluorescence from the entire TIRF illumination zone, photobleaching 63.2% of the signal will preferentially reduce the signal below penetration depth d_1. Here, it is assumed that photobleaching is dependent on the irradiation level of incident beam and the irradiation time, which produces a photobleaching profile at angle θ_1 resembling the intensity profile of the evanescent wave. As a consequence, photobleaching occurs preferentially in the layers of the specimen closer to the coverslip and the post-bleach image has approximately 36.8% of the original fluorescence with most derived from higher axial positions. Subtracting this image from the pre-bleach image provides an image in which signal from higher axial positions is preferentially removed since molecules at these positions are photobleached to a lesser extent than positions closer to the coverslip.

Next the incident beam angle is changed from θ_1 to θ_2 to image the second layer. This layer is defined as z_1 to z_2 where z_1 equals the penetration depth d_1 and z_2 equals the penetration depth d_2. The fluorescence generated at axial depth z_1 (equivalent to d_1) when the penetration depth is d_2 is

$$F_{d1,\theta2} = I_{0,\theta2}\, C\, d_2\, (1 - e^{-d1/d2}) \tag{22}$$

However, since the first layer has been photobleached, the fluorescence detected by the camera in the first image of the second layer will be

$$F_{total\text{-}d1,\theta2} = F_{total,\theta2} - F_{d1,\theta2} \tag{23}$$

$$F_{total\text{-}d1,\theta2} = I_{0,\theta2}\, C\, d_2 - (I_{0,\theta2}\, C\, d_2\, (1 - e^{-d1/d2})) = I_{0,\theta2}\, C\, d_2\, e^{-d1/d2} \tag{24}$$

and the second layer of fluorescence will be

$$F_{d2\text{-}d1,\theta2} = I_{0,\theta2}\, C\, d_2\, (e^{-d1/d2} - e^{-1}) \tag{25}$$

and

$$\frac{F_{d2\text{-}d1,\theta2}}{F_{total\text{-}d1,\theta2}} = 1 - e^{(d1/d2\, -\, 1)} \tag{26}$$

If d_1=100 nm and d_2=150 nm, the ratio is equal to 28.3 %, which means that we photobleach 28.3% of fluorescence in the first image of the second layer. The same protocol and analysis is followed for subsequent layers.

When we decrease the incident beam angle from $\boldsymbol{\theta_1}$ to $\boldsymbol{\theta_2}$ toward the critical angle ($\boldsymbol{\theta_1} > \boldsymbol{\theta_2}$), the generated evanescent wave intensity at the surface will increase (see the Table S1). Because most of the fluorophores in the previous layer are photobleached, the fluorescence when switching to the subsequent angle, and thus greater TIRF depth, will be actually substantially decreased. Therefore, the incident laser beam power is increased to offset the loss in fluorescence. Table S1 gives guidelines for the percentage of fluorophores to be photobleached in each layer and the relative excitation power increase due to the incident beam angle changes.

5. Verification of 20 nm axial resolution

To verify the 20 nm axial resolution of multiangle TIRF with sequential imaging and photobleaching, we examined images of 20-nm and 40-nm fluorescent beads using this method. Commercialized uniform 40-nm and 20-nm fluorescent beads were embedded in 0.4%(w/vol) agarose gel so that beads can be distributed in all layers. This gel has a refractive index (n=1.3329) close to water (5). In the imaging result shown in **Fig. S5**, the axial depth is calculated to be approximately 20-nm between each layer. In the 40-nm fluorescent bead image in **Fig. S5*A***, the beads labeled 1, 3, 5, and 6 appeared in two adjacent layers, while other beads such as those labeled 2 and 4 appeared in three adjacent layers. None of the 40-nm beads were observed in more than 3 layers. In the 20-nm fluorescent bead images in **Fig. S5*B***, most beads were observed in only a single layer, although some beads such as those labeled in **Fig. S5B** were observed in two adjacent layers. Importantly, no 20-nm bead signal was observed in more than two layers. This result confirmed the 20 nm axial resolution of this multiangle TIRF with sequential imaging and photobleaching method.

6. Calibration using two-color beads

Since multi-color imaging was also of interest in these studies, the penetration depths for different angles of incidence were determined with two excitation wavelengths using the image-photobleach-image protocol. Silica beads were labeled with both EGFP and mCherry and depth determinations were made sequentially starting with 561-nm excitation of mCherry (**Fig. S6 *A-D***) to avoid photobleaching of mCherry with the 488-nm laser. Images of beads labeled with EGFP were then taken by using 488-nm laser with the image-photobleach-image protocol (**Fig. S6 *E-H***) and overlaid with the red images at each incidence angle (**Fig. S6 *I-L***). Intensity profiles along the lines across the center of beads (**Fig. S6 *M-P***) were used to estimate the TIRF depth (**Fig. S3*C***). The results showed that the intensity peaks from two channels are very close to each other and the differences of penetration depths measured from two channels are less than 20%.

7. Illumination field vs. imaging field

With our TIRF microscope, the illumination field shows a Gaussian-like shape across the 512 x 512 pixel image (**Fig. S10**). To provide the necessary evenly

illuminated field for photobleaching and image acquisition, we restrict our analysis to center part of the field which corresponds to an imaging field of approximately 150 x 150 pixels (18 μm x 18 μm). Within this restricted area (**Fig. S10*B***), the illumination is nearly uniform.

8. Demonstration of uni-directional multiangle TIRF

Although we rotated the illumination beam azimuthally in the back focal plane to reduce background and more uniformly illuminate the sample, we think the sequential imaging and photobleaching method could also be performed on commercially available uni-directional TIRF microscopes. Of course, this requires characterization of the relationship between the incident beam angles and penetration depth of the evanescent waves. To demonstrate the feasibility of using the sequential imaging and photobleaching method on uni-directional TIRF microscope, we fixed the incident beam illumination in one direction and only moved one galvomirror. The multiangle TIRF images after sequential imaging and photobleaching are shown in **Figure S12**. Similar to our images collected using azimuthal scanning, the color-coded 3D reconstructed image (**Fig. S12*E***) also shows actin filament bundles intertwined at different depths within the TIRF excitation zone.

9. Notes on the limitations of the multiangle TIRF with sequential imaging and photobleaching technique.

A. Signal-to-noise

Photobleaching at each layer reduces the number of available fluorescing molecules and hence the number of photons from each subsequent layer. Even when we decrease the incident angle which also results in an increase in the incident beam power, the fluorescence in the subsequent layer is still relatively weak. Because we still need enough photons to reconstruct the image, the decrement of the incident angle cannot be too small and so far we only been able to obtain reliable results when thickness of each layer is ≥20nm.

We performed a simple subtraction analysis on a series of multiangle TIRF images after normalization for intensity to see how shot noise may limit these images. Based on the pixel values of the difference images, the estimation for the photon counts ranged from ~20 photons per pixel in the dimmer areas of the image up to ~170 per pixel to the brighter structures depending on the section. Assuming the shot noise is proportional to the square root of the number of photons in the original image, the photons estimated in the difference images for some of the dimmer structures are approaching expected shot noise values. Data from the same structures collected under the photobleaching protocol indicate that despite the loss in fluorescent molecules, it does not suffer from shot noise limitations more than simply subtracting images in a multiangle TIRF series. Our simple arithmetical analysis deals with noise in the most simplistic manner. One of the methods of 3D TIRF image reconstruction through acquiring a stack of TIRF images at a range of incidence angles may do better.

B. Imaging rate and irradiation power

Since we take two images at each incident angle and also need time to perform the

bleaching event between two images, the time to collect a single image is extended. The photobleaching step generally takes longer than acquiring images and the photobleaching time depends on the power of the laser, the incident angle, and the fluorophore. We used the same laser power to perform both imaging and photobleaching within the complete cycle requiring up to approximately 2 minutes per layer. This could presumably be improved with more laser power, but we do not think this method is currently appropriate for the study of fast dynamics and we have restricted our studies to fixed cells.

C. Photobleaching control

Controlling the extent of photobleaching is key for this protocol and imaging a laterally heterogeneous biological cell using this method presents multiple challenges. Using the rationale outlined above (Section 4. Image Collection and Analysis), we rely on the whole imaging field intensity reduction rather than specific regions. This is intended to help avoid any bias in our image collection. Photoactivation, dequenching, or photoswitching to an "on" state during the photobleaching step would be detrimental behavior and we could not use such a fluorophore in these experiments. Depending on their extent, multi-exponential and reversible components during the photobleaching step could result in errors with this process. For the probes used in these studies, we find their photobleaching to be consistently dependent on the illumination power and angle. It should be noted that this finding is based on our experience with the calibration beads where we have known TIRF depth dependent fluorescence profiles. Thus, we are relying on the photobleaching behavior of the fluorophores observed in our samples to be similar to the behaviors observed when the fluorophores are attached to the calibration beads.

D. Potential artifacts affecting the penetration depth calibration

As discussed earlier, the slow exponential decay due to scattering reported previously (3) was not evident when we performed our calibrations. Our results fitted well with a single exponential and were consistent with the expected theoretical values for two calibration methods under different refractive index conditions. Thus, we have been unable to reconcile the differences, but we certainly do not discount those previous findings and suggest users of this method to be advised of those studies.

We did note the "comet" artifact resulting from the calibration bead refractive index mismatch with the surrounding medium described previously (3) (**Fig. S11A**). Azimuthally spinning the incident beam makes the comet effect to be azimuthally isotropic, but this also reduces background from this effect by averaging (**Fig. S11B**). In fact, we noted faster spinning speeds produce a lower comet induced background due to averaging. Given the presence of this extra background, we further tested for its effects on the calibration by comparing measurements of penetration depth in a refractive index matched 60% glycerol solution (n=1.42) and a normal water based solution. Images of unlabeled beads with fluorescein in a 60% glycerol solution did not exhibit the comet effect due to the refractive index match. However, penetration depth measurements using unlabeled silica beads in the water and in 60% glycerol/water-FITC were both consistent with theoretical results (**Fig. S3F**). These data suggested to us that while the "comet" or "ghosting" due to

bead-solution refractive index mismatch may be observed during calibration, the increased background associated with this seems to have a negligible effect on our calibration. Nevertheless, users should be aware of this potential artifact.

Supplementary Methods

Even illumination multiangle TIRF imaging system

The laser sources, 100-mW 405-nm Cube laser (Coherent, Inc., Santa Clara, CA, USA), 50-mW 488-nm laser (Oxxius, Lannion, France, product number LBX-488-50-CIR-PP), and 50-mW 561-nm Sapphire laser (Coherent, Inc.) were linearly combined using dichroic mirrors (Semrock, Inc., Rochester, NY, product number Di02-R405-25x36 and Di02-R488-25x36) and directed into an acousto-optic tunable filter (AA Opto-electronic Inc., Orsay, France, product number AOTFnC-400.650) to switch between the laser wavelengths and control the laser intensities. The azimuthal-rotation laser beam was generated by a 2D galvanometer system (Thorlabs, Inc. Newton, NJ, USA, product number GVSM002), in which two mirrors were controlled by two sine waves with a phase difference of approximately 90 degrees output from a waveform generator (Agilent Technologies, Inc., Santa Clara, CA, USA, product number 33522A). The laser beam was expanded 2X with two achromatic lenses (Edmund Optics, Inc., Barrington, NJ, USA, f=50 mm, product number 49-665; Thorlabs, Inc., f=100 mm, product number AC254-100-A) before passing through a tube lens (Thorlabs, Inc., f=200 mm, product number AC254-200-A) which was focused at the back focal plane of an Olympus 100X/1.49NA oil objective (Olympus America, Inc., Center Valley, PA, USA). The laser beam was directed into the rear port of an Olympus IX70 microscope and reflected from a multi-band dichroic mirror (Semrock, Inc., product number Di01-R488/561-25x36) to the objective. A C-Focus system (Mad City Labs Inc., Madison, WI, USA) was used to compensate for focus drift over long time periods. The fluorescence emission passed through the multi-band dichroic mirror to the standard IX-70 tube lens, a beam expander consisting of two achromatic lenses (Thorlabs, Inc., f=75 mm, product number AC254-75-A and f=100 mm, product number AC254-100-A) and an emission filter wheel (Lambda 10-2 filter wheel, Sutter Instrument, Novato, CA, USA) containing bandpass filters for green and red fluorescence, respectively (Semrock, Inc., product number FF01-525/45-25; FF01-600/37-25). The emission was detected with an Andor iXon$^+$ EMCCD (Andor Technology USA, South Windsor, CT, USA).

Fluorescent beads and fluorescence monolayer preparation.

Fluorescently labeled beads were produced by incubating Histidine$_6$-tagged EGFP (His-EGFP) with uniform non-functionalized silica beads (mean diameter 7.27μm and refractive index 1.42, Bangs Laboratories, Inc., Fishers, IN, USA). They were subsequently washed with phosphate buffered saline (PBS) 5 times prior to imaging. His-EGFP was produced and purified using Ni-NTA agarose in our lab as previously described (6). The monolayer of His-EGFP for critical angle determination was prepared by sequentially coating a coverslip with monoclonal Histidine$_6$-Tag antibody (GenScript USA Inc., Piscataway, NJ, USA, product number A00186-100) and His-EGFP. The coverslip was washed with PBS three times prior to imaging.

Cell culture, transfection and immunofluorescence labeling

U2OS cells were cultivated on Nanostrip cleaned (Cyantek, Fremont, CA,

USA) #1.5 25-mm-diameter coverslips (Warner Instruments, Hamden, CT, USA, product number 64-0715) in standard DMEM-HG medium (Invitrogen, Life Technologies, Grand Island, NY, USA, product number 11960) with 2 mM Glutamax (Invitrogen, product number 35050), 1 mM sodium pyruvate (Invitrogen, product number 11360) and 10%(vol/vol) heat-inactivated FBS (Invitrogen, product number 10082). Cells were grown overnight on glass cover slides at 37 °C in a 5% CO_2 incubator until they reached approximately 75% confluence.

For clathrin-mediated endocytosis, A431 cells were cultured in DMEM medium supplemented with 10%(vol/vol) FCS, 2 mM glutamax, and 1 mM sodium pyruvate. Cells were serum starved in DMEM supplemented with 2 mM glutamax and 1 mM sodium pyruvate for ~10 h prior to experiment. Cells were incubated on ice cold HEPES-buffered DMEM medium for 1hour followed by incubation on ice containing 2 µg/mL Alexa-Fluor555 labeled EGF ligand (Life Technologies, Cat#:E-35350) in HEPES-buffered DMEM. Cells were washed with ice cold PBS and incubated at 37°C in HEPES-buffered DMEM for the indicated period of time to allow receptor-ligand internalization and transport into the cell. Cells were subsequently fixed by 4%(w/vol) paraformaldehyde and immunolabeled with anti-clathrin heavy chain antibody (Thermo Fisher Scientific, Inc., Cat#: MA1-065) followed by incubation with Alexa Fluor 488 goat anti-mouse IgG (Life technologies, A-11001) as indicated below.

For immunofluorescence labeling, cells were fixed with 4%(w/vol) paraformaldehyde (Electron Microscopy Services, Hatfield, PA, USA, product number 15710) for 15 min at 37°C. After fixation, the cells were washed in PBS twice, quenched with 100 mM glycine in PBS and blocked in antibody-dilution buffer (AbDil buffer; 150 mM NaCl, 20 mM Tris, 0.1%(vol/vol) Triton X-100, 0.1%(w/vol) NaN_3 and 2% bovine serum albumin, pH 7.4). Actin filaments were stained for 1 hour with phalloidin-Alexa Fluor® 488 (Invitrogen, product number A12379) and/or phalloidin-Alexa Fluor® 568 (Invitrogen, product number A12380) stocked in methanol and diluted in PBS. Microtubules were stained using a primary antibody: alpha-tubulin monoclonal antibody, mouse (Invitrogen, product number 322500) and secondary antibodies: Goat anti-mouse IgG (H+L)-Alexa Fluor® 568 (Invitrogen, product number A11004) and/or goat anti-mouse IgG (H+L)-Alexa Fluor® 488 (Invitrogen, product number A11001). Both antibodies were diluted in AbDil buffer and incubated for 1 h each followed by three times washing steps in PBS.

U2OS cells were transfected with plasmid DNA expressing Zyxin-Emerald via X-tremeGENE HP DNA transfection reagent. 100 µl pre-warmed serum-free DMEM transfection medium and 1 µL 0.5 µg/µl DNA were added into a 1.5 ml microfuge tube. Three µl X-tremeGENE HP transfection reagent was added into the tube after warming to room temperature. The mixture was vortexed and incubated at the room temperature for 15 min before adding to the cell culture dish. The transfected cells were incubated at 37 °C in a 5% CO_2 incubator overnight prior to fixation and/or imaging.

Molecule localization analysis

Bayesian analysis of blinking and bleaching (3B) (7) was performed on images of fixed U2OS cells labeled with phalloidin-Alexa Fluor 488 (Invitrogen, product number A12379). All 3B analyses were performed using custom software designed for use on the NIH Biowulf cluster parallel processing system (https://hpc.nih.gov). All custom software scripts needed to perform parallelized 3B analysis and instructions on their use are available as supplemental information. The extreme length of time required to complete 3B analysis on a large field of view (FOV) of 300x300 pixels in a stack of approximately 200 images is prohibitive. Therefore, the full FOV image stacks were divided into arrays of equally sized, non-overlapping sub-regions of interest (ROI) and parallelized analysis was performed on the separate sub-ROIs. The custom software subdivides the full FOV images, dilates the sub-ROIs by a set amount to prevent edge effects and performs 3B analysis separately on each of the dilated sub-ROIs. After completing the 3B analysis, each dilated sub-ROI is cropped to its original size and reassembled into the full FOV image. For data presented in this work the size of each image stack was 206 frames acquired during the photobleaching steps, the size of each sub-ROI was 15x15 pixels, total dilation for each sub-ROI was 4 pixels in each dimension (two pixels per side), the number of 3B iterations run was 200 and mu was set to 0.252147. Mu was calculated according to the formula presented previously (8) using a pixel size of 120 nm and a lateral FWHM of 358.5 nm.

Supplementary Figures:

Figure S1. Schematic diagram of the multiple-angle TIRF.

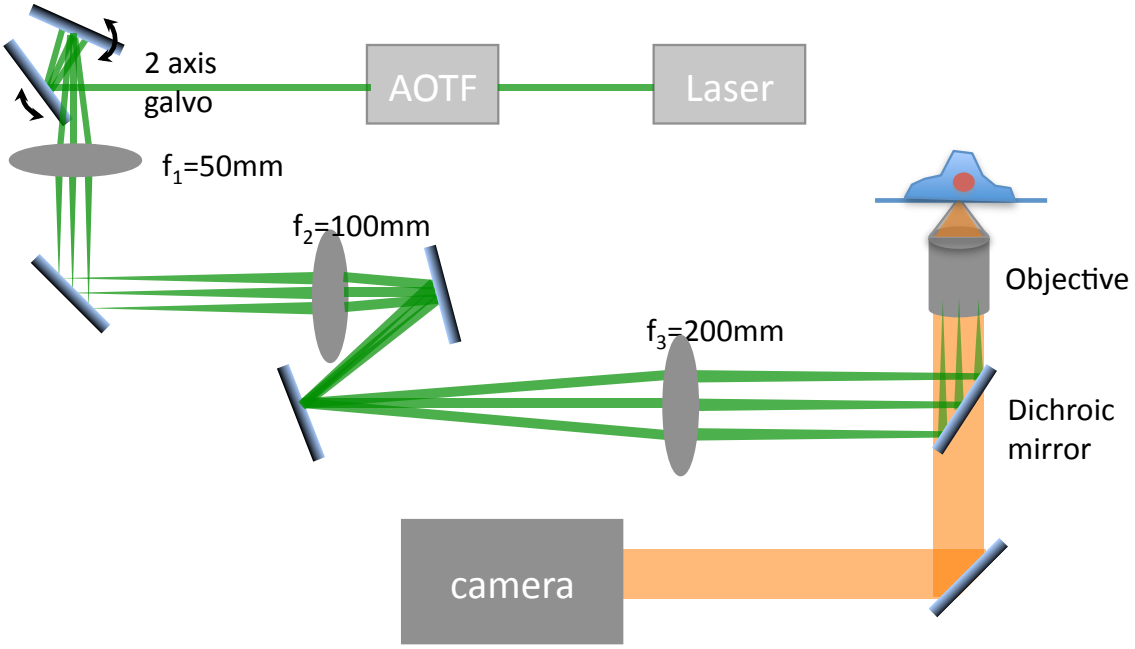

Figure S1. Schematic diagram of the multiple-angle TIRF. Laser sources, 100-mW 405-nm Cube laser (Coherent, Inc.), 50-mW 488-nm laser (Oxxius, LBX-488-50-CIR-PP), 50-mW 561-nm Sapphire laser (Coherent, Inc.) were combined using dichroic mirrors (Di02-R405-25x36 and Di02-R488-25x36, Semrock) and directed into an acousto-optic tunable filter (AA Opto-electronic Inc., AOTFnC-400.650). The azimuthal-rotation of the laser beam was generated by a 2D galvanometer system (Thorlabs, GVSM002), in which two mirrors were controlled by two sine waves with a phase difference of 89.2 degrees output from a waveform generator (Agilent Technologies, 33522A). The laser beam was expanded 2X with two achromatic lenses (Edmund Optics, f=50 mm, 49-665; Thorlabs, f=100 mm, AC254-100-A) before passing through a tube lens (f=200 mm, AC254-200-A) which was focused at the back focal plane of an Olympus 100X/1.49NA oil objective. The laser beam was directed into the rear port of an Olympus IX70 microscope and reflected from a multi-band dichroic mirror (Di01-R488/561-25x36, Semrock) to the objective. The fluorescence emission was detected with an Andor iXon^{+} EMCCD (Andor Technology).

Figure S2. Calculation and measurement of the incident beam angle.

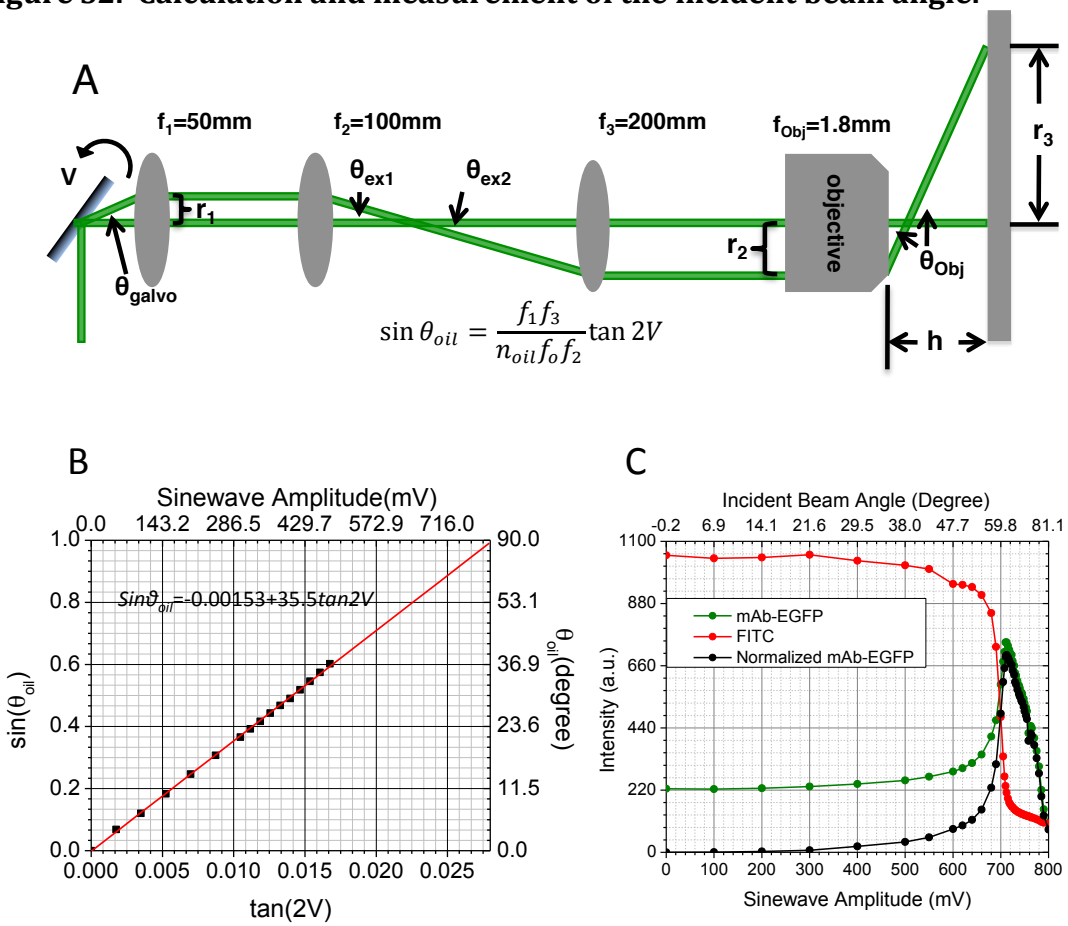

Figure S2. Calculation and measurement of the incident beam angle. (***A***) Diagram describing the relationship between the incident beam angle (θ_{Obj}) and the positions of the galvanometer mirrors (V). (***B***) The beam angle output from the objective in air was determined as a function of the sine wave amplitudes (black squares). These values were converted to angles in immersion oil (n_{oil}=1.5155) using Snell's law and extrapolated to 90° (solid red line). Using n_{water}=1.3333 and n_{glass}=1.5168 for the interface, the critical angle was calculated to be 61.5 degrees which corresponds to an amplitude of 0.7099 volts. (***C***) A thin layer of His-tagged EGFP (mAb-EGFP) was bound to anti-His antibody coated coverslips and imaged at increasing angles of incidence (solid green circles). The fluorescence of a FITC solution was also imaged at increasing angles of incidence (solid red circles). The values from the FITC solution were used to normalize the bound His-EGFP values (solid black circles). Relying on the fluorescence intensity of the EGFP monolayer as the readout of illumination intensity, the critical angle was estimated to correspond to an amplitude of 0.7120 volts.

Figure S3. Penetration depth calculation.

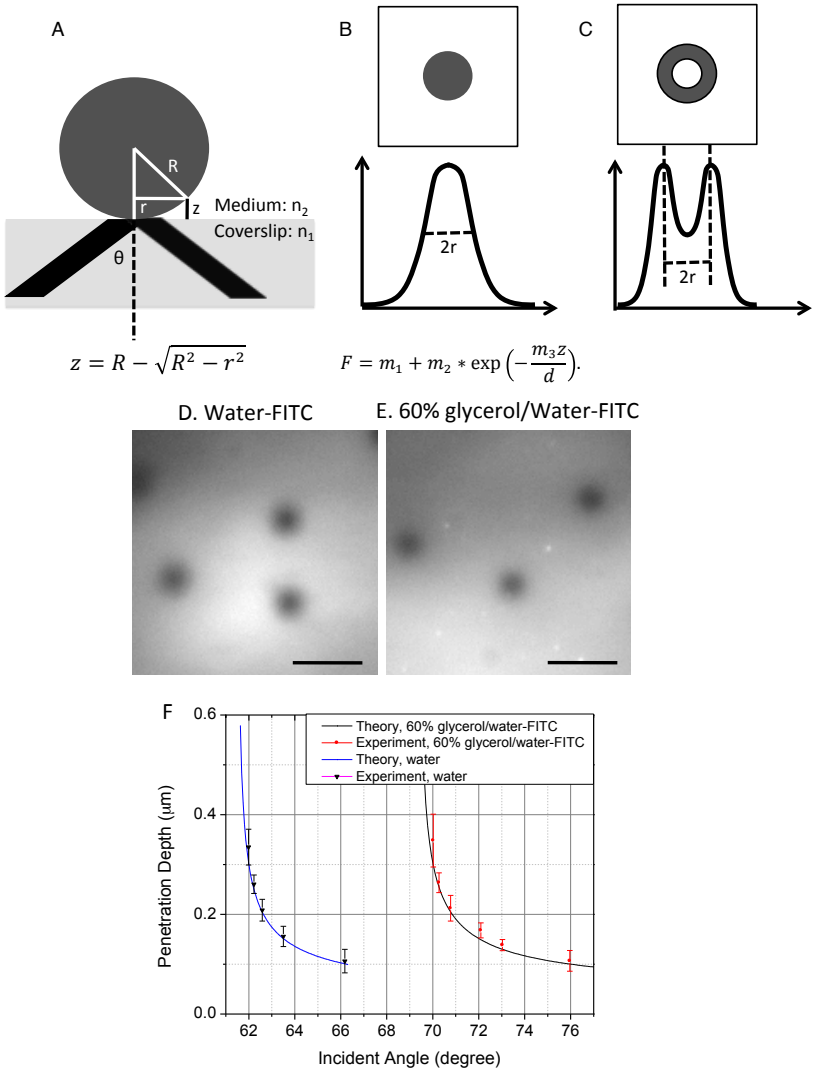

Figure S3. Penetration depth calculation. (***A***) A silica bead was labeled with a fluorophore and imaged at increasing angles of incidence. The imaging depth (z) can be estimated based on the geometry of the bead, where R is the radius of the bead and r is the radius of the imaged bead on the camera. (***B***) The radius of imaged bead (r) was determined by the full width of half maxima from the intensity profile along the line across the center of the bead or by the fitting the plot values from the maximum to the minimum values with a single exponential decay equation. (***C***) With the exception of the first depth image, r is estimated by the distance between two peaks of an intensity profile along the line across the middle of the bead in the image-photobleach-image protocol. Unlabeled silica beads were placed in a fluorescein water based solution (**D**) or 60% glycerol/water based solution (**E**) and imaged by multiangle TIRF with sequential imaging and photobleaching. (**F**) The penetration depth is shown for the theoretical values and experimental results in water-FITC and 60% glycerol/water-FITC solution. Scale bar = 5 μm in **D** and **E**.

Figure S4. Image collection and processing.

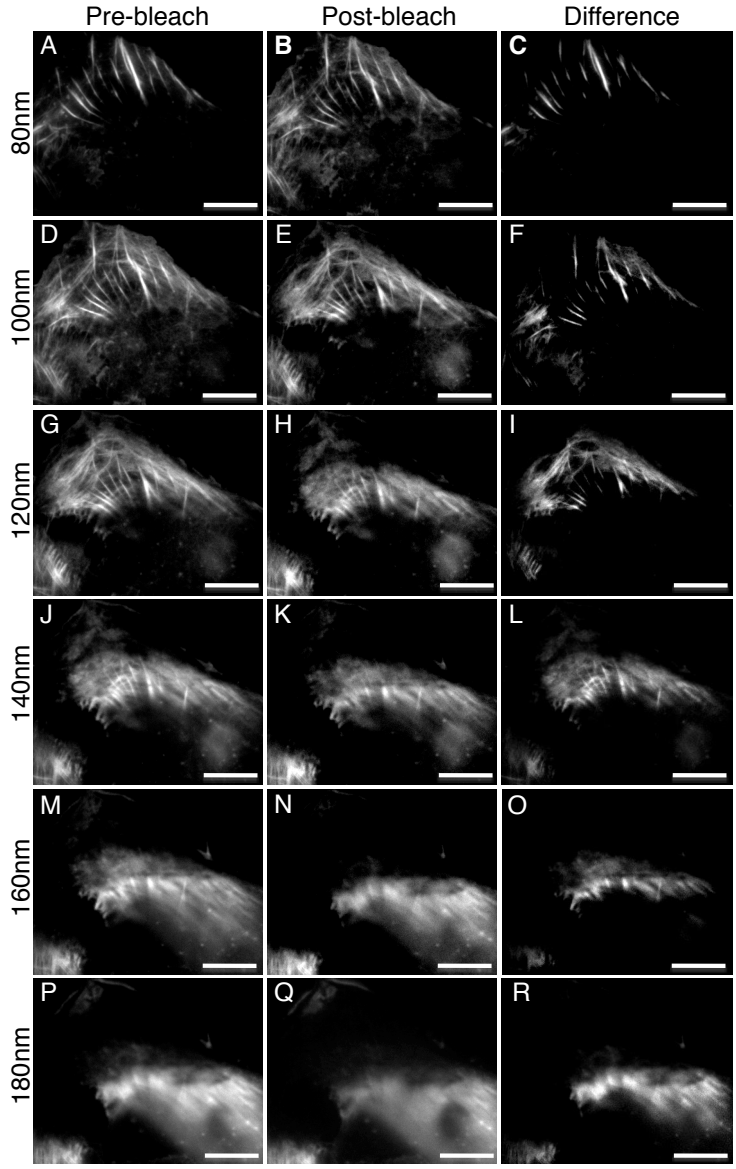

Figure S4. Image collection and processing. Images were collected at a starting TIRF depth, 80 nm in this example. (***A***) This image is referred to as a "pre-bleach" image. The sample was then subjected to photobleaching to a pre-determined level and imaged again to obtain the "post-bleach" image shown in (***B***). (***C***) The difference image when subtracting the post-bleach image from the pre-bleach image shows the fluorescence in this layer or section. This process is repeated at subsequent TIRF depths (***D-F***) 100 nm, (***G-I***) 120 nm, (***J-L***) 140 nm, (***M-O***) 160 nm, and (***P-R***) 180 nm with increased laser power (see **Table S1**). Scale bars = 10 μm.

Figure S5. Multiangle TIRF images of 20-nm and 40-nm fluorescent beads in agarose gel with sequential imaging and photobleaching.

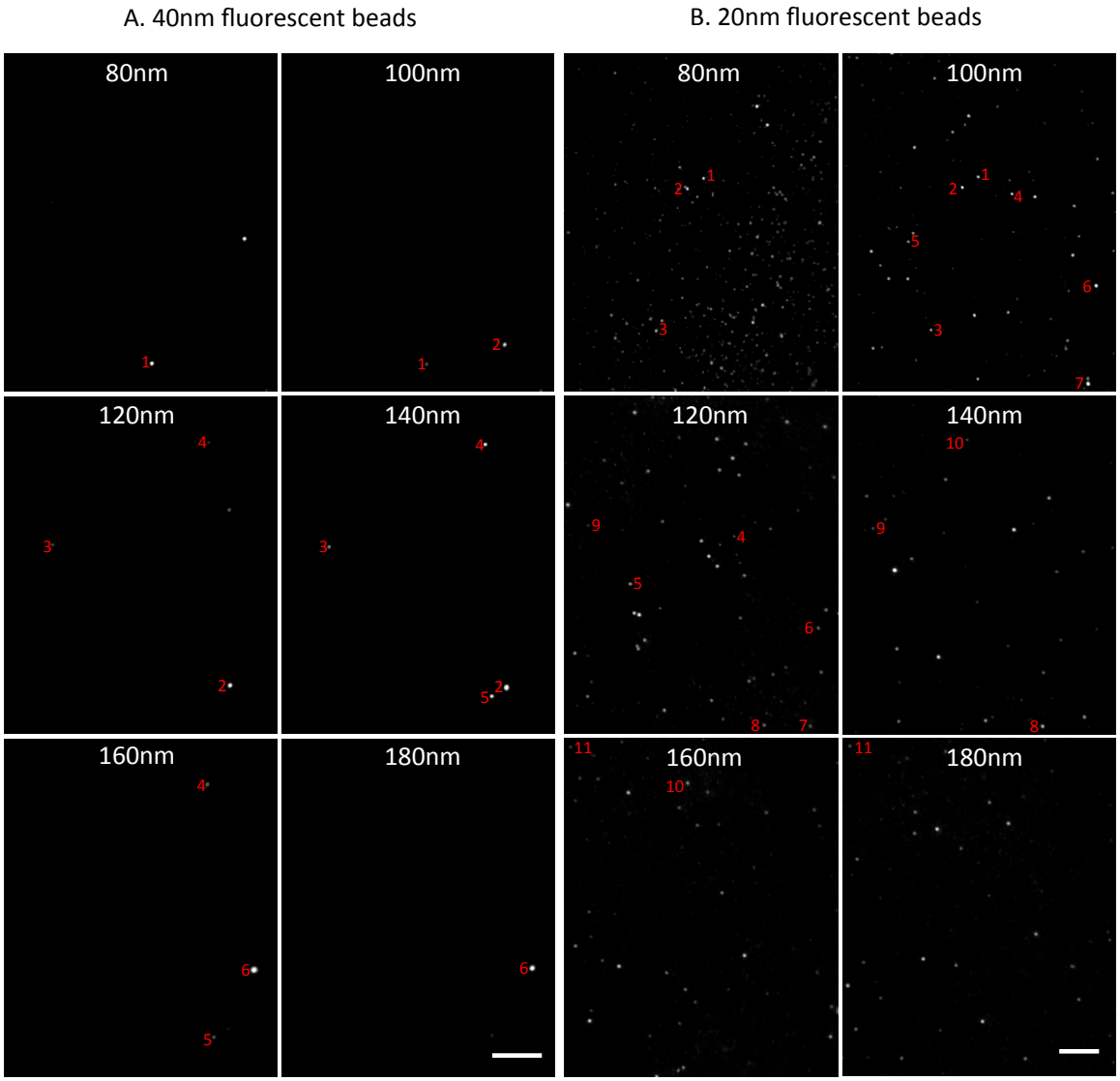

Figure S5. Multiangle TIRF images of 20-nm and 40-nm fluorescent beads in agarose gel with sequential imaging and photobleaching. The penetration depth difference between each layer was calculated to be approximately 20 nm. (***A***) Images of 40-nm fluorescent beads were collected at penetration depths ranging from 80 nm to 180 nm. Fluorescence signals from beads 1, 3, 5 and 6 were observed in two adjacent layers. Signals from beads 2 and 4 were observed in three adjacent layers. (***B***) Images of 20-nm fluorescent beads were collected at penetration depths ranging from 80 nm to 180 nm. Signals from most beads were found only in one layer. Beads labeled 1 to 11 were all observed in two adjacent layers but no bead fluorescence signals were found in more than 2 layers. Bar scale = 5 μm.

Figure S6. Calibration of two-color multiangle TIRF images using silica beads.

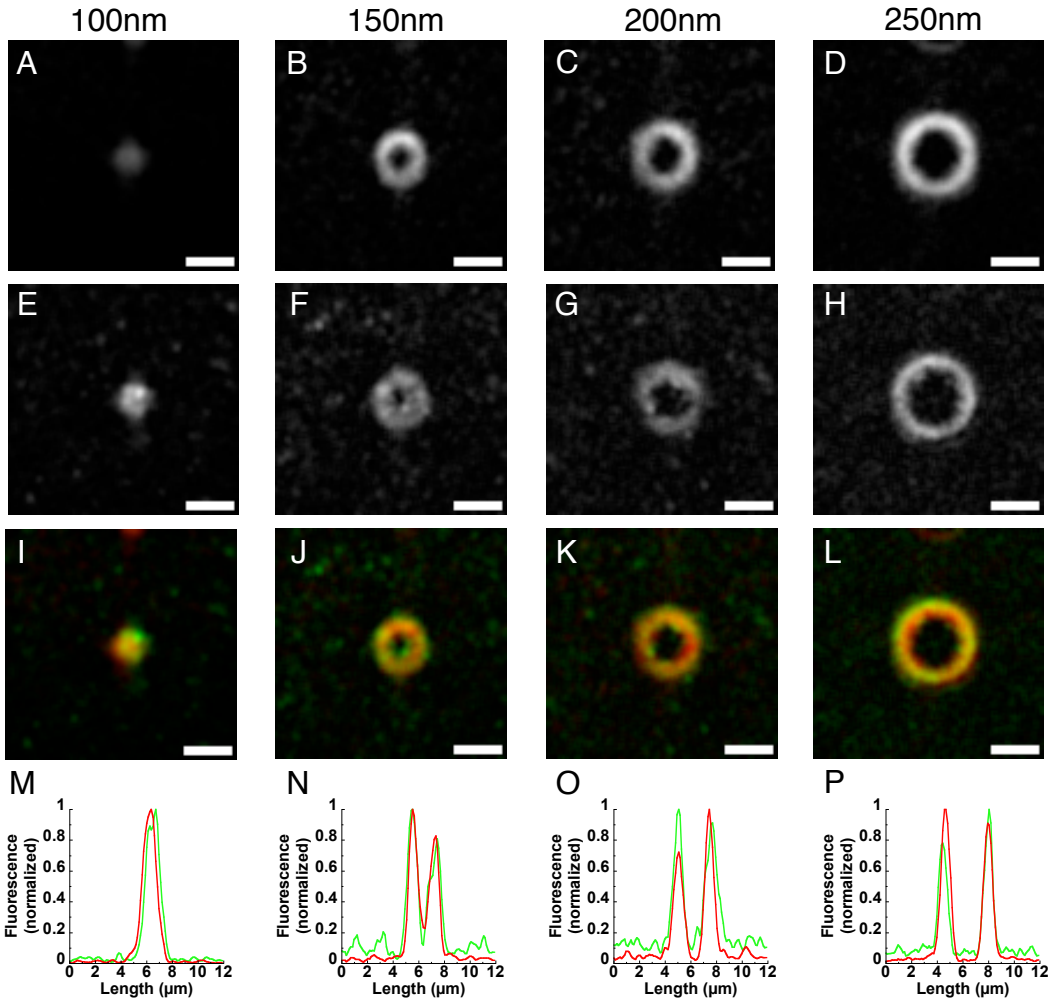

Figure S6. Calibration of two-color multiangle TIRF images using silica beads. Silica beads were labeled with both EGFP and mCherry. Images of mCherry labeled beads were collected using 561-nm excitation under even illumination multiangle TIRF at depths of (***A***) 100 nm, (***B***) 150 nm, (***C***) 200 nm, and (***D***) 250 nm. Images of EGFP labeled beads were collected using 488-nm excitation under multi-direction multiangle TIRF at depths of (***E***) 100 nm, (***F***) 150 nm, (***G***) 200 nm, and (***H***) 250 nm. Images collected in (***A-D,*** red) were merged with their respective TIRF depth images collected in (***E-H,*** green). Images were processed by background subtraction and smoothing. Lines (5 pixel width) were drawn across the midpoint of the beads and plot profiles are shown for mCherry signal (red) and EGFP signal (green) for depths of (***M***) 100 nm, (***N***) 150 nm, (***O***) 200 nm, and (***P***) 250 nm. Scale bars = 2.5 μm.

Figure S7. Two-color multiangle TIRF imaging of actin-filaments with sequential imaging and photobleaching.

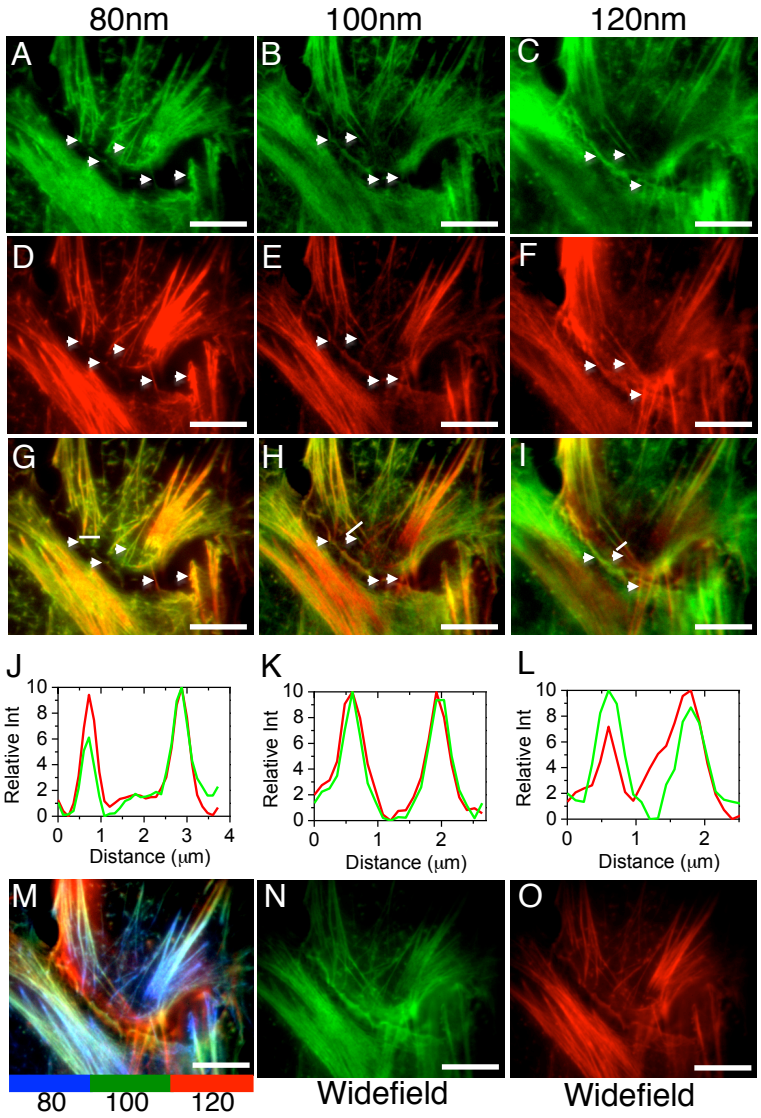

Figure S7. Two-color multiangle TIRF imaging of actin-filaments with sequential imaging and photobleaching. Actin-filaments were co-labeled with phalloidin-Alexa Fluor 488 and phalloidin-Alexa Fluor 568. (***A-C***) Green images and (***D-F***) red images were collected using 488-nm and 561-nm excitation, respectively, at penetration depths of 80 nm, 100 nm and 120 nm. (***G-I***) The red and green images were merged and (***J-L***) plot profiles were obtained in the areas indicated by the white lines in (***G-I***). Arrows indicate the structures in each layer which were observed in both green and red images. (***M***) A color-coded 3D image at penetration depths from 80 to 120 nm was produced for the red channel 561-nm excitation. Widefield images of the (***N***) green and (***O***) red channels are shown for reference. Scale bars = 10 μm.

Figure S8. Two-color multiangle TIRF imaging of actin-filaments, microtubules, and zyxin with sequential imaging and photobleaching.

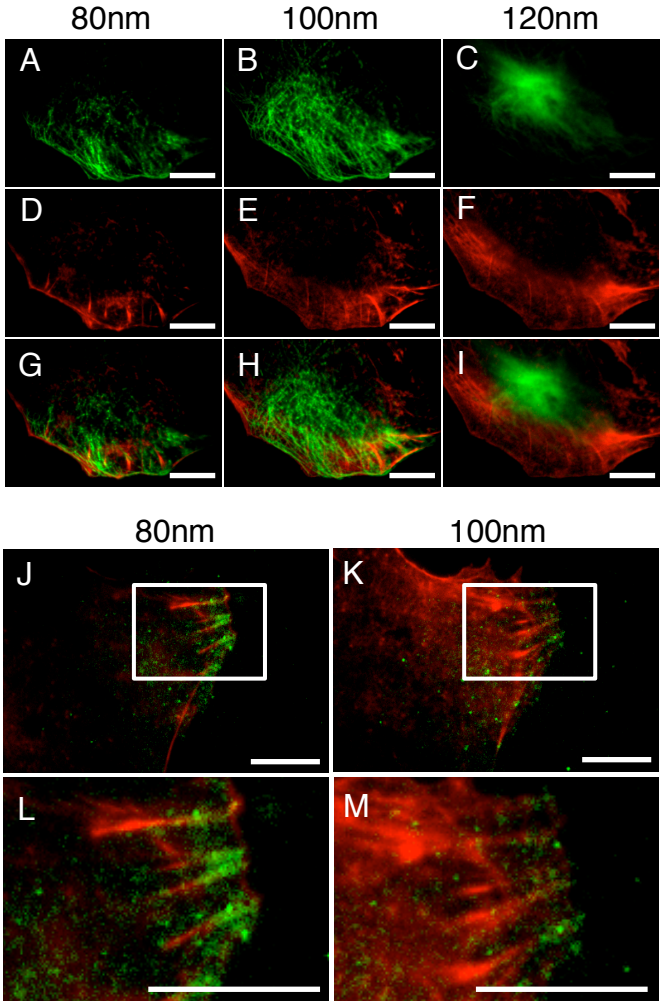

Figure S8. Two-color multiangle TIRF imaging of actin-filaments, microtubules, and zyxin with sequential imaging and photobleaching. A U2OS cell was fixed and labeled with phalloidin-Alexa Fluor 568 to image filamentous actin and immuno-labeled with an anti-tubulin antibody followed by an Alexa Fluor 488 secondary antibody. (***A***-***C***) Green images and (***D***-***F***) red images were collected using 488-nm and 561-nm excitation, respectively, at penetration depths of (***A***,***D***,***G***) 80 nm, (***B***,***E***,***H***) 100 nm and (***C***,***F***,***I***) 120 nm. (***G***-***I***) The red and green images were merged to show the relative distributions of microtubules and filamentous actin. (***J***-***M***) A U2OS cell expressing zyxin-Emerald (**green**) was fixed and labeled with phalloidin-Alexa Fluor 568 (**red**) to image filamentous actin. Green and red images were collected using 488-nm and 561-nm excitation, respectively, at penetration depths at (***J*** and ***L***) 80 nm and (***K*** and ***M***) 100 nm. The two channels were merged to show the relative distributions of zyxin and filamentous actin. Images (***L***) and (***M***) are magnified regions outlined in the white boxes in (***J***) and (***K***), respectively. Scale bars = 10 μm in all images.

Figure. S9 Determination of lateral resolution after 3B analysis.

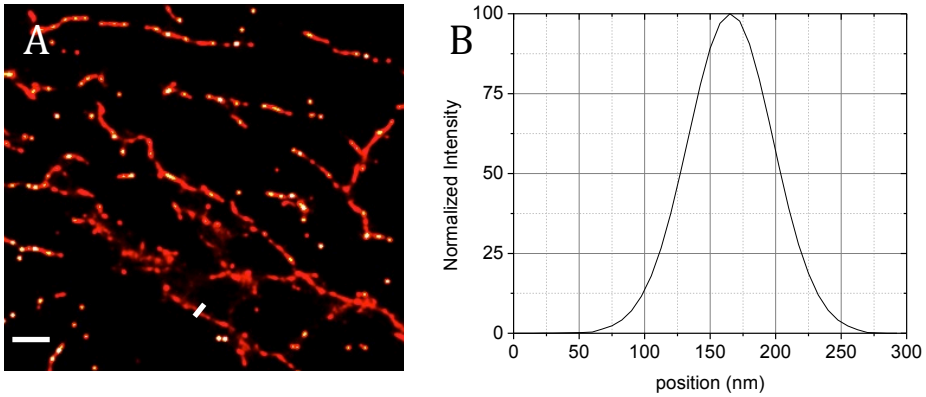

Figure S9. Determination of lateral resolution after 3B analysis. A U2OS cell was fixed and immune-labeled with anti-tubulin antibody followed by an Alexa Fluor 488 secondary antibody. 3B analysis followed multiangle TIRF imaging with sequential imaging and photobleaching. (**A**) The image of microtubule after 3B analysis taken at the TIRF illumination beam angle corresponding to the penetration depth 100nm. (**B**) Intensity profile along the line across one fiber indicated in (**A**). The FWHM of the line profile was referred as lateral resolution of 75 nm. Scale Bar = 1 μm.

Figure. S10 Intensity profile of the illumination field.

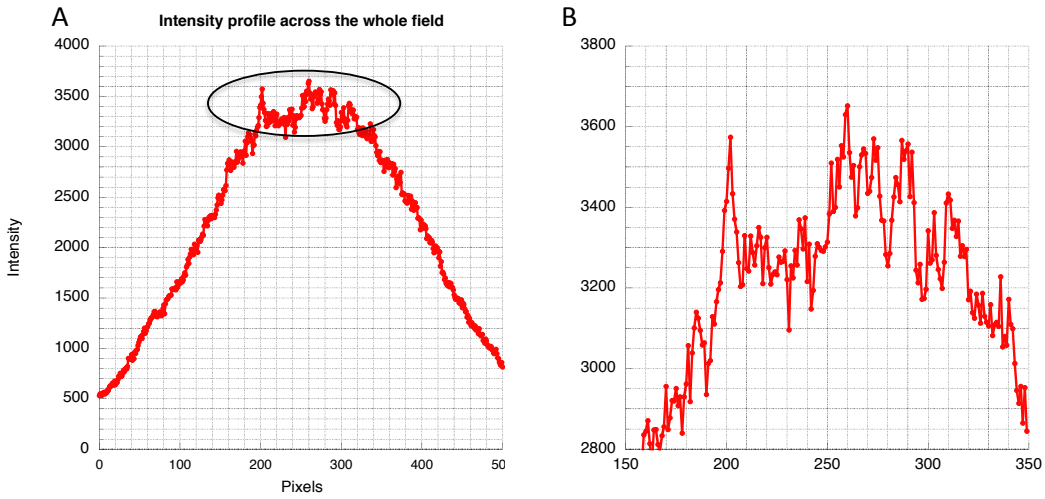

Figure S10. Intensity profile of the illumination field. (**A**) Intensity profile of a typical TIRF illumination across the whole imaging field. (**B**) The center of intensity profile indicated by the circle in (**A**) includes 200 pixels.

Figure S11. Comparison of images of a silica bead using fixed TIRF beam in the fixed direction and spinning TIRF beam azimuthally.

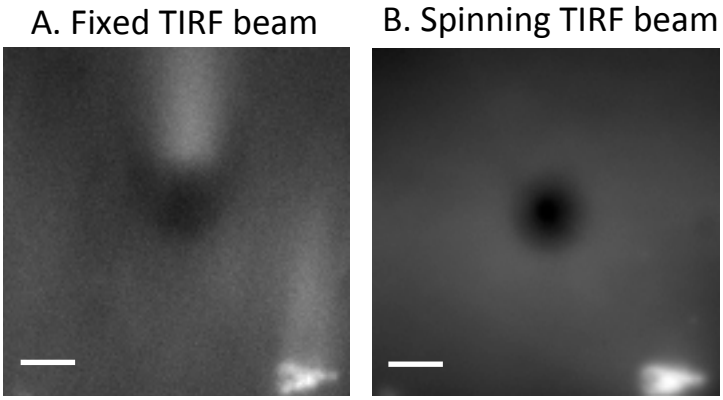

Figure S11. Comparison of images of a silica bead using fixed TIRF beam in the fixed direction and spinning TIRF beam azimuthally. Unlabeled silica beads were imaged in a fluorescein solution. The solution was illuminated by an evanescent wave from a fixed incident beam in one direction (**A**). The solution was also illuminated with an evanescent wave produced by a spinning incident beam (**B**). The image was obtained using the same exposure time as in (**A**) during which time the beam made 82 rounds. Scale bar = 5 μm.

Figure. S12 Unidirectional multiangle TIRF images of actin filaments with sequential imaging and photobleaching method.

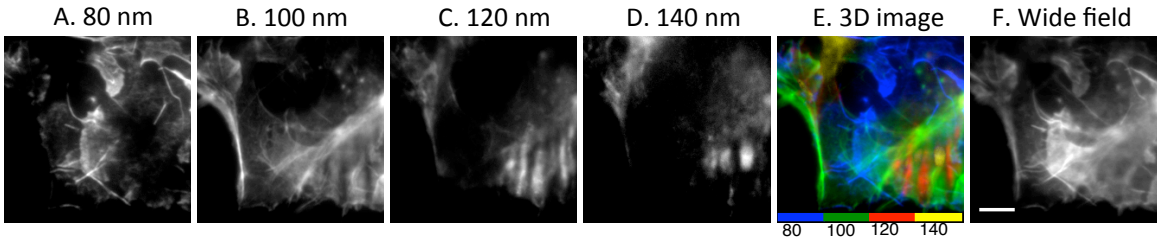

Figure S12. Unidirectional multiangle TIRF images of actin filaments with sequential imaging and photobleaching method. A COS7 cell was fixed and labeled with phalloidin-Alexa Fluor 488 to image filamentous actin. (**A**-**D**) Multiangle TIRF images with sequential imaging and photobleaching were collected at penetration depths ranging from 80 nm to 140 nm. (**E**) The color-coded 3D image was reconstructed from images (**A**-**D**). (**F**) The wide-field image was collected when the incident beam angle was below the critical angle. Scale bar = 5 μm.

Table S1: Photobleaching parameters used in this study.

	Angle (n_1=1.333; n_2=1.5168)	Fluorescence intensity to photobleach (%)
1st layer: d_1=100nm	66.246	63.2
2nd layer: d_2=150nm	63.564	28.3
3rd layer: d_3=200nm	62.644	22.1
4th layer: d_4=250nm	62.224	18.1
5th layer: d_5=300nm	62.002	15.4

1st layer: d_1=80nm	69.290	63.2
2nd layer: d_2=100nm	66.246	18.1
3rd layer: d_3=120nm	64.740	15.4
4th layer: d_4=140nm	63.846	13.3
5th layer: d_5=160nm	63.285	11.8
6th layer: d_6=180nm	62.920	10.5
7th layer: d_7=200nm	62.644	9.5

Table S1. Photobleaching parameters used in this study. The sample is treated as a uniform structure in the axial direction which has been arbitrarily divided into different layers. Each layer results from the difference between two penetration depths (d) that correspond to two incident angles, respectively. Two images are collected in each layer with the first image collected immediately after movement to a new incident beam angle designated as pre-bleach image. A second image is collected after irradiation of the sample to decrease the fluorescence through photobleaching and is designated as the post-bleach image. The level of photobleaching used at each angle is indicated in "Fluorescence intensity to photobleach" column. The percentages indicated are relative to fluorescence emission intensity in the pre-bleach image collected at each angle. The final image of each layer is obtained by subtracting the post-bleach image from the pre-bleach image at each incident angle. As the incident angle was altered, the relative illumination intensity was adjusted based on equations (17) and (18) to compensate.

References:

1. Axelrod D (2001) Total internal reflection fluorescence microscopy in cell biology. *Traffic* 2:764-774.
2. Axelrod D (2008) Chapter 7: Total internal reflection fluorescence microscopy. *Methods Cell Biol* 89:169-221.
3. Mattheyses AL & Axelrod D (2006) Direct measurement of the evanescent field profile produced by objective-based total internal reflection fluorescence. *J Biomed Opt* 11(1):014006.
4. Axelrod D, Burghardt TP, & Thompson NL (1984) Total internal reflection fluorescence. *Annu Rev Biophys Bioeng* 13:247-268.
5. Byron ML & Variano EA (2013) Refractive-index-matched hydrogel materials for measuring flow-structure interactions. *Exp Fluids* 54:1456-1461.
6. Patterson GH & Lippincott-Schwartz J (2002) A photoactivatable GFP for selective photolabeling of proteins and cells. *Science* 297(5588):1873-1877.
7. Cox S*, et al.* (2012) Bayesian localization microscopy reveals nanoscale podosome dynamics. *Nat Methods* 9(2):195-200.
8. Rosten E, Jones GE, & Cox S (2013) ImageJ plug-in for Bayesian analysis of blinking and bleaching. *Nat Methods* 10(2):97-98.

Using the 3B Scripts in Biowulf

This document is courtesy of Victor Wang (CIT) and Peter Winter (NIBIB).

The following scripts have been provided below:

- autorunMulti.bat
- email.bat
- genScript.bat
- makeImage.bat
- makeImage.m
- makeMask.bat
- makeMask.m
- makeStacks.bat
- makeStacks.m
- nGenScript.bat
- parseResultsMulti.bat
- reservoir.bat
- showq.bat

User interaction is fairly minimal with the scripts used to run the 3B analysis. However, there are a couple of important steps to take into account. The first is properly setting the parameters describing the 3B analysis. These are stored in nGenScript.bat, so open the script in your chosen text editor. Most of the variables should be changed, except potentially the last few. The ones to change are:

- MASKSIZE: how large you want your sub-regions in the image to be
- XSIZE and YSIZE: the image dimensions
- ITERATIONS: the number of iterations to run, which is then multiplied by 4 for the actual number run
- BLUR_MU and BLUR_SIGMA: imaging parameters of your system (see the Instruction Manual for calculation info)
- BASEDIR: the location of the scripts as well as the data folders
- DATADIR: the location where data is stored. Data should be stored in subdirectories of this location.
- IMTEMPLATE: the pattern which input images follow. This should be followed by the wildcard character.
- MASKNAME: the name of the mask to determine which portions of the image to run.

A note on the input masks; there are 3 different modes for this to work. If no mask is provided, then the program will automatically generate regions to calculate based on the input images. However, this does not necessarily best represent the data. The second mode is by providing a mask and placing it in the directory specified by DATADIR. This will act as a mask for ALL the data sets in the data directory. The third mode is that each data set has its own mask, denoted by Mask_$DATASET.tif,

which resides in the data?s subdirectory. If this is not present, the second mode will be used on the data set. If that also does not work, the first mode is used.

NOTE: With the 3B parameters set, the last variable that needs to be changed is the email recipients. Open email.bat and change the EMAIL variable to whoever you want emailed once the analysis has completed.

Once these are changed to match the user specifications, save the script and quit out. You may then run the autorunMulti.bat script (by sourcing it like so: ./autorunMulti.bat). Upon execution, the script will prompt you for several answers. The first asks which mask mode to use. Answering yes uses mode one, while answering no uses either mode two or three depending on file availability. The second prompt asks for whether the data sets are of SPIM nature of not. The third and final prompt is just a last-chance check to make sure that you have set all of the parameters correctly, both in nGenScript and in the prompts.

If you answered yes to the final prompt, the scripts will automatically run the 3B analysis, from submission to the cluster to actually visualizing the data in image form. An email will be sent to the set recipients (in the email script) once the entire process has been completed. At this point, the images should be in your data directory under the folder Image_files/images.

The set of scripts developed by Victor Wang and Peter Winter for the Biowulf cluster follow. The text in the following pages can be used to make the necessary batch and Matlab files.

The text below can be used to make the "**autorunMulti.bat**" file.

```
#!/bin/bash

while true; do
   read -p "Generate mask files without input? (y/n): " yn
   case $yn in
      [Yy]* ) SWITCH="y"; break;;
      [Nn]* ) SWITCH="n"; break;;
      * ) echo "Please answer y or n.";;
   esac
done

while true; do
   read -p "Are data SPIM stacks? (y/n): " yn
   case $yn in
      [Yy]* ) DSPIM="y"; break;;
      [Nn]* ) DSPIM="n"; break;;
      * ) echo "Please answer y or n.";;
   esac
done

while true; do
   read -p "Are the settings in nGenScript.bat correct? (y/n): " yn
   case $yn in
      [Yy]* ) qsub -l nodes=1 -k n -v run=1,SWITCH=$SWITCH,DSPIM=$DSPIM
nGenScript.bat; break;;
      [Nn]* ) break;;
      * ) echo "Please answer y or n.";;
   esac
done
```

The text below can be used to make the "**email.bat**" file.

```
#!/bin/bash
# script to send simple email
# email subject
SUBJECT="Swarm job has completed"
# Email To ?
EMAIL="your.email@here.com, your.otheremail@here.com"
# Email text/message
EMAILMESSAGE="emailmessage.txt"
cd /data/HROI
shopt -s nullglob
echo -e "A swarm job has been completed\n"> $EMAILMESSAGE
if [ -n "$JOBDIR" ]
then
        echo -e "Data is in ${JOBDIR}\n">>$EMAILMESSAGE
fi
#if [ -n "$JOBID" ]
#then
#       echo -e "Job ID: ${JOBID}\n" >>$EMAILMESSAGE
#fi
#if [ -n "$SWARMID" ]
#then
        #LOOP HERE OVER $SWARMID.o* files
#       for f in ${SWARMID}.o*
#       do
#               echo -e "\nRuntime information for swarm job: ${f}
\n">>$EMAILMESSAGE
#               grep -A6 "Show some" ${f} | grep ".biobos" >> $EMAILMESSAGE
#       done
#fi
echo -e "\nCheck shared data folder" >>$EMAILMESSAGE
# send an email using /bin/mail
/bin/mail -s "$SUBJECT" "$EMAIL" < $EMAILMESSAGE
sleep 3
```

The text below can be used to make the "**genscript.bat**" file.

```
#!/bin/bash
#
#genScript.bat
#
#Written by Victor Wang (NIH/CIT)
#
#SET THESE VARIABLES HERE######################
#Generated masks will be of this size
MASKSIZE=15
#Number of iterations to run (divided by 4)
ITERATIONS=50
#Mask dilation size
DILATION=2.2
#Spot shape priors based on microscope specs
BLUR_MU=-0.079209
BLUR_SIGMA=0.1
#Base directory, where the config file is
BASEDIR=/data/your_base_directory
#Directory within BASEDIR that contains the data
DATADIR=your_data_directory
#General template for the images in the stack
IMTEMPLATE=Cell2_*
#Name of the mask for the make_grid_markup program
MASKNAME=largemaskcell.tif #Cell2_SPIMB_33-10000.tif #largemaskcell.tif
#Name of the command file
SWARMNAME=swarmfile.txt
#Template for results file. Will be modified for iterations
OUTNAME=results
#Name for the outputted mask, not that important
FILTERNAME=filtered
#Image extension for the mask, not that important
IMEXT=.tiff
#END OF VARIABLES##############################

SWITCHMASK=${SWITCH:-"n"}
EXESTR=multispot5_headless
INPUTDIR=${BASEDIR}/${DATADIR}
OLDOUT=${BASEDIR}/old_out
SCRIPTFILE=${INPUTDIR}/${SWARMNAME}
RESULTSDIR=${INPUTDIR}/results
INPUTMASK=${INPUTDIR}/${MASKNAME}
INPUTFILE=${INPUTDIR}/${IMTEMPLATE}
MASKSDIR=${INPUTDIR}/masks
```

```
INCR=1
TEMPSTR=${RESULTSDIR}_${INCR}
shopt -s nullglob
module load 3B

if [ $SWITCHMASK == "y" ] || [ -z "$MASKNAME" ] || [ ! -e "$INPUMASK" ]
then
        TEMP=( $INPUTFILE )
        INPUTMASK=${TEMP[0]}
fi

#Move any old standard out/error folders to storage
#if [ ! -d $OLDOUT ]
#then
#       mkdir $OLDOUT
#fi

#mv ${BASEDIR}/genScript.bat.* $OLDOUT
#mv ${BASEDIR}/email.bat.* $OLDOUT
OLDMASKDIR=${z}old_masks
OLDRESDIR=${z}old_results
INCR=1
TEMPSTR=${OLDRESDIR}_${INCR}


#Change new result directory name if present
if [ -d $RESULTSDIR ]
then
  while [ -d $TEMPSTR ]
  do
    let "INCR+=1"
    TEMPSTR=${OLDRESDIR}_${INCR}
  done
  OLDRESDIR=${TEMPSTR}
  mv  ${RESULTSDIR} ${OLDRESDIR}
fi

INCR=1
TEMPSTR=${OLDMASKDIR}_${INCR}

#Change new mask directory name if present
if [ -d $MASKSDIR ]
then
  while [ -d $TEMPSTR ]
  do
    let "INCR+=1"
```

```
    TEMPSTR=${OLDMASKDIR}_${INCR}
  done
  OLDMASKDIR=${TEMPSTR}
  mv  ${MASKSDIR} ${OLDMASKDIR}
fi

#Remove old command file if present
if [ -f "$SCRIPTFILE" ]
then
        rm $SCRIPTFILE
fi

mkdir ${MASKSDIR}
mkdir ${RESULTSDIR}
FILTERFILE=${MASKSDIR}/${FILTERNAME}-%04i${IMEXT}

#Create subregions from large, original mask
#This portions must be run on the cluster, which is why
#the entire script must be submitted
make_grid_markup --image ${INPUTMASK} \
        --size $MASKSIZE \
        --radius $DILATION \
        --mask /dev/null \
        --filter $FILTERFILE

#Create the commands to run each subregion of the
#image as determined by the masks created in the
#previous step. Also sets up the results files so
#that each result file will correspond to a submask
#file by name.
for f in ${MASKSDIR}/*
do
        EXTLEN=${#IMEXT}
        NAMELEN=${#f}
        let "NUM=${NAMELEN}-${EXTLEN}-4"
        NUMSTR=${f:NUM:4}
        echo "$EXESTR --save_spots ${RESULTSDIR}/${OUTNAME}_${NUMSTR}.txt
\
        --log_ratios $f \
        --dilate_mask_as_filter_radius ${DILATION} \
        --main.total_iterations ${ITERATIONS} \
        --blur.mu ${BLUR_MU} \
        --blur.sigma ${BLUR_SIGMA} \
         ${INPUTFILE}" >> $SCRIPTFILE
done
```

```
shopt -u nullglob
#autorun command, so will run the swarm commands automatically
#as well as submit the email scripts properly.
if [ "$run" == "1" ]
then
        cd ${BASEDIR}
#       if [ -z "$batch" ]
#       then #no batching, just submit the commands as a jobarray
#               SWARMID=`swarm -f $SCRIPTFILE --module 3B --jobarray`
#       else #want to batch commands together
#               SWARMID=`swarm -f $SCRIPTFILE -b $batch --module 3B --jobarray`
#       fi
        SWARMID=`swarm -f $SCRIPTFILE --module 3B --jobarray`

        BIOIND=`expr index $SWARMID ".bio"`
        let "BIOIND-=1"
  JOBNUM=${SWARMID:0:BIOIND}
  JOBINFO=`qstat -r $JOBNUM | grep ".bio"`
  SWARMINFO=( $JOBINFO )
  SWARMJOB=${SWARMINFO[3]} #looks only for the swarm job name
        echo $SWARMID
        echo $BIOIND
        echo $JOBNUM
        echo $JOBINFO
        echo $SWARMINFO
        echo $SWARMJOB

        #Runs the script that parses all the result files into
        #something useful
        QSUBID=`qsub -l nodes=1 -v
BDIR=${BASEDIR},RDIR=${RESULTSDIR},RNAME="results_*",INAME=${DATADIR}
\
        -W depend=afterany:${SWARMID} ${BASEDIR}/parseResults.bat`
        #submit email script so that you recieve an
        #email once the analysis has completed
        qsub -l nodes=1 -v
JOBID=${SWARMID},JOBDIR=${RESULTSDIR},SWARMID=${SWARMJOB}  -W
depend=afterany:${QSUBID} ${BASEDIR}/email.bat
fi
```

The text below can be used to make the "**makeImage.bat**" file.

```
#!/bin/bash
#
#makeImage.bat
#
#Written by Victor Wang (NIH/CIT)
#
#Wrapper for makeImage.m
#Creates an image from the results files
#Generic version, will make images but
#no stacks
#
#The methods is based on the same
#reconstruction methods as seen in the 3B
#analysis ImageJ plugin

#SET USER VARIABLES HERE##################################
#Should be consistent with previous scripts
IMAGEDIR=/data/your_base_directory/your_data_directory/Image_files
XSIZE=512
YSIZE=512
PIXELNM=162.5
PIXELREC=10
FWHM=100
BASEDIR=/data/your_base_directory
DATADIR=your_data_directory
IMDIR=Image_files
#END OF USER VARIABLES

BASEDIR=${BDIR:-$BASEDIR}
DATADIR=${DDIR:-$DATADIR}
IMDIR=${IDIR:-$IMDIR}
XSIZE=${WIDTH:-$XSIZE}
YSIZE=${HEIGHT:-$YSIZE}

IMAGEDIR=${BASEDIR}/${DATADIR}/${IMDIR}

cd $BASEDIR
if [ ! -d "${IMAGEDIR}/images" ]
then
        mkdir ${IMAGEDIR}/images
fi
module load matlab
```

```
matlab -nodisplay -r "makeImage('$IMAGEDIR', $XSIZE, $YSIZE, $PIXELNM,
$PIXELREC, $FWHM);quit"
```

The text below can be used to make the "**makeImage.m**" file.

```
%This function is based on the 3B Image reconstruction as
%seen in the ImageJ plugin.
function makeImage(txt, xSize, ySize, pixelnm, pixelrec, fwhm)
	%location=txt;
	zoom=pixelnm/pixelrec;
	trueX=floor(xSize*zoom);
	trueY=floor(ySize*zoom);
	blur=fwhm/(2*sqrt(2*log(2)))*zoom/pixelnm;
	klength=2*ceil(3*blur)+1;
	h = fspecial('gaussian', klength, blur);
	files = dir([txt '/*_image.txt']);
	for location = files'
		location
		file=fopen([txt '/' location.name]);
		fscanf(file, 'PIXELS %*d');
		fscanf(file, '%*d');
		fscanf(file, '\n');
		lines=fscanf(file, 'PASS0: 1 1 %f %f\n', [2 inf]);
		fclose(file);
		image=zeros(trueY, trueX);
		len=length(lines);
		for i=1:len
			x=floor(lines(1,i)*zoom + 0.5);
			y=floor(lines(2,i)*zoom + 0.5);
			image(y,x)=image(y,x)+1;
		end
		h
		blurred = imfilter(image, h, 'replicate', 'conv');
		immax = max(max(blurred));
		image = uint16(round(65535 * blurred / immax));
		ext=strfind(location.name, '.');
		lenext=length(ext);
		ind=ext(lenext);
		imout=[txt '/images/' location.name(1:ind-1) '.tif'];
		cmap = colormap('gray');
		imwrite(image,cmap,imout,'tif');
	end
end
```

The text below can be used to make the "**makeMask.bat**" file.

```
#!/bin/bash
#
#makeMask.bat
#
#Written by Victor Wang (NIH/CIT)
#
#bash wrapper to make the masks based on inputs

module load matlab
cd /data/your_base_directory

XSIZE=128
YSIZE=127
XSTART=45
YSTART=45
XEND=90
YEND=90
DEST=/data/your_base_directory/Victor_Test/mask.tif

matlab -nodisplay -r "makeMask($XSIZE , $YSIZE , $XSTART , $YSTART , $XEND ,
$YEND, '$DEST' ); quit"
```

The text below can be used to make the "**makeMask.m**" file.

```
function makeMask(xSize, ySize, xStart, yStart, xEnd, yEnd, location)

image=zeros(ySize,xSize);
for(i=[xStart:yEnd])
        for(j=[yStart:yEnd])
                image(j,i)=255;
        end
end

imwrite(image, location);

end
```

The text below can be used to make the "**makeStacks.bat**" file.

```
#!/bin/bash
#
#makeStacks.bat
#
#Written by Victor Wang (NIH/CIT)
#
#Wrapper for makeImage.m
#Creates an image from the results files

#SET USER VARIABLES HERE###############################
#Should be consistent with previous scripts
IMAGEDIR=/data/your_base_directory/your_data_directory/Image_files
XSIZE=512
YSIZE=512
PIXELNM=162.5
PIXELREC=10
FWHM=100
BASEDIR=/data/your_base_directory
DATADIR=your_data_directory
IMDIR=Image_files
#END OF USER VARIABLES

BASEDIR=${BDIR:-$BASEDIR}
DATADIR=${DDIR:-$DATADIR}
IMDIR=${IDIR:-$IMDIR}
XSIZE=${WIDTH:-$XSIZE}
YSIZE=${HEIGHT:-$YSIZE}

IMAGEDIR=${BASEDIR}/${DATADIR}/${IMDIR}
IMAGES=${IMAGEDIR}/images
echo $IMAGEDIR

cd $BASEDIR
if [ ! -d "$IMAGES" ]
then
        mkdir $IMAGES
fi

module load matlab

matlab -nodisplay -r "makeStacks('$IMAGEDIR', $XSIZE, $YSIZE, $PIXELNM,
$PIXELREC, $FWHM);quit"
```

The text below can be used to make the "**makeStacks.m**" file.

```
function makeStacks(txt, xSize, ySize, pixelnm, pixelrec, fwhm)
	%location=txt;
	zoom=pixelnm/pixelrec;
	trueX=floor(xSize*zoom);
	trueY=floor(ySize*zoom);
	blur=fwhm/(2*sqrt(2*log(2)))*zoom/pixelnm;
	klength=2*ceil(3*blur)+1;
	h = fspecial('gaussian', klength, blur);
	files = dir([txt '/*_image.txt']);
	height=0;
	cmap = colormap('gray');
	imoutA=[txt '/images/SPIMA_stack.tif'];
	imoutB=[txt '/images/SPIMB_stack.tif'];
	for location = files'
		name=location.name;
		numarr=regexp(name, '\d');
		numstr='';
		for i=1:length(numarr)
			numstr=[numstr name(numarr(i))];
		end
		plane=str2num(numstr);
		if plane > height
			height = plane;
		end
	end
	for plane=1:height
		filenameA=[txt '/SPIMA_' num2str(plane) '_image.txt'];
		filenameB=[txt '/SPIMB_' num2str(plane) '_image.txt'];
		image=zeros(trueY,trueX);
		if exist(filenameA, 'file') ~= 0
			file=fopen(filenameA);
			fscanf(file, 'PIXELS %*d');
			fscanf(file, '%*d');
			fscanf(file, '\n');
			lines=fscanf(file, 'PASS0: 1 1 %f %f\n', [2 inf]);
			fclose(file);
			len=length(lines);
			for i=1:len
				x=floor(lines(1,i)*zoom + 0.5);
				y=floor(lines(2,i)*zoom + 0.5);
				image(y,x)=image(y,x)+1;
			end
			blurred = imfilter(image, h, 'replicate', 'conv');
```

```
                immmax = max(max(blurred));
                image = uint16(round(65535 * blurred / immax));
                imout=[txt '/images/SPIMA_' num2str(plane) '.tif'];
                imwrite(image,cmap,imout,'tif');
            end
            imwrite(uint16(image),cmap,imoutA, 'tif', 'WriteMode', 'append');
            image=zeros(trueY, trueX);
            if exist(filenameB, 'file') ~= 0
                file=fopen(filenameB);
                fscanf(file, 'PIXELS %*d');
                fscanf(file, '%*d');
                fscanf(file, '\n');
                lines=fscanf(file, 'PASS0: 1 1 %f %f\n', [2 inf]);
                fclose(file);
                len=length(lines);
                for i=1:len
                    x=floor(lines(1,i)*zoom + 0.5);
                    y=floor(lines(2,i)*zoom + 0.5);
                    image(y,x)=image(y,x)+1;
                end
                blurred = imfilter(image, h, 'replicate', 'conv');
                immax = max(max(blurred));
                image = uint16(round(65535 * blurred / immax));
                imout=[txt '/images/SPIMB_' num2str(plane) '.tif'];
                imwrite(image,cmap,imout,'tif');
            end
            imwrite(uint16(image),cmap,imoutB, 'tif', 'WriteMode', 'append');

        end
end
```

The text below can be used to make the "**nGenScript.bat**" file.

```
#!/bin/bash
#
#nGenScript.bat
#
#Written by Victor Wang (NIH/CIT)
#
#SET THESE VARIABLES HERE#####################
#Generated masks will be of this size
MASKSIZE=15
#Image Size
XSIZE=300
YSIZE=210
#Number of iterations to run (divided by 4)
ITERATIONS=50
#Mask dilation size
DILATION=2.2
#Spot shape priors based on microscope specs
#Formula for mu is in instruction manual
BLUR_MU=0.252147
BLUR_SIGMA=0.1
#Base directory, where the config file is
BASEDIR=/data/your_base_directory
#Directory within BASEDIR that contains the data
DATADIR= your_data_directory
#General template for the images in the stack ('*' denotes wildcard)
IMTEMPLATE=Image_*
IMEXTENSION=.tif
#Name of the mask for the make_grid_markup program
MASKNAME=Mask.tif
#Name of the command file
SWARMNAME=swarmfile.txt
#Template for results file. Will be modified for iterations
OUTNAME=results
#END OF VARIABLES#############################

#If the mask should be autogenerated. Default is
#no. Set externally, but can also be tripped by
#setting MASKNAME to an empty string.
SWITCHMASK=${SWITCH:-"n"}

EXESTR=multispot5_headless
INPUTDIR=${BASEDIR}/${DATADIR}
SCRIPTFILE=${INPUTDIR}/${SWARMNAME}
```

```
INPUTMASK=${INPUTDIR}/${MASKNAME}
shopt -s nullglob
module load 3B

#Remove old command file if present
if [ -f "$SCRIPTFILE" ]
then
        rm $SCRIPTFILE
fi

for z in $INPUTDIR/*/
do

RESULTSDIR=${z}results
INPUTFILE=${z}${IMTEMPLATE}${IMEXTENSION}
MASKSDIR=${z}masks
OLDMASKDIR=${z}old_masks
OLDRESDIR=${z}old_results
INCR=1
TEMPSTR=${OLDRESDIR}_${INCR}

#Check if there are any image files in there
#before doing anything, so that it doesn't
#try to do stuff in Image_files
FILES=( $INPUTFILE )
FILELEN=${#FILES[@]}
if [ $FILELEN == "0" ]
then
        continue
fi

#Change new result directory name if present
#and move any old results to a folder called
#old_resultsN
if [ -d $RESULTSDIR ]
then
  while [ -d $TEMPSTR ]
  do
    let "INCR+=1"
    TEMPSTR=${OLDRESDIR}_${INCR}
  done
  OLDRESDIR=${TEMPSTR}
  mv  ${RESULTSDIR} ${OLDRESDIR}
fi

INCR=1
```

```
TEMPSTR=${OLDMASKDIR}_${INCR}

#Change new mask directory name if present
#and move any old masks in the same way as
#results
if [ -d $MASKSDIR ]
then
   while [ -d $TEMPSTR ]
   do
      let "INCR+=1"
      TEMPSTR=${OLDMASKDIR}_${INCR}
   done
   OLDMASKDIR=${TEMPSTR}
   mv  ${MASKSDIR} ${OLDMASKDIR}
fi

mkdir ${MASKSDIR}
mkdir ${RESULTSDIR}
FILTERFILE=${MASKSDIR}/filtered-%04i.tiff

#Determines whether to use the large input mask
#or to simply pass the first image in the stack
#to determine mask. SWITCHMASK is usually set
#externally, but leaving MASKNAME blank will also
#trigger this case
#
#Also, in the case that an actual input mask file
#cannot be found, the first image on the stack
#will be used.

DIRNAME=${z#"${INPUTDIR}/"}
DIRNAME=${DIRNAME%"/"}
INDMASK=${z}Mask_${DIRNAME}.tif
if [ -e $INDMASK ] && [ "$SWITCHMASK" == "n" ]
then
        INPUTMASK=$INDMASK
elif [ "$SWITCHMASK" == "y" ] || [ -z "$MASKNAME" ] || [ ! -e "$INPUTMASK" ]
then
        TEMP=( $INPUTFILE )
        INPUTMASK=${TEMP[0]}
fi
#Create subregions from large, original mask
#This portions must be run on the cluster, which is why
#the entire script must be submitted
make_grid_markup --image $INPUTMASK \
        --size $MASKSIZE \
```

```
        --radius $DILATION \
        --mask /dev/null \
        --filter $FILTERFILE

#Create the commands to run each subregion of the
#image as determined by the masks created in the
#previous step. Also sets up the results files so
#that each result file will correspond to a submask
#file by name.
for f in ${MASKSDIR}/*
do
        #EXTLEN=${#IMEXT}
        #NAMELEN=${#f}
        #let "NUM=${NAMELEN}-${EXTLEN}-4"
        #NUMSTR=${f:NUM:4}
        NUMSTREXT=${f#"${MASKSDIR}/filtered-"}
        NUMSTR=${NUMSTREXT%".tiff"}
        echo "$EXESTR --save_spots ${RESULTSDIR}/${OUTNAME}_${NUMSTR}.txt
\
        --log_ratios $f \
        --dilate_mask_as_filter_radius ${DILATION} \
        --main.total_iterations ${ITERATIONS} \
        --blur.mu ${BLUR_MU} \
        --blur.sigma ${BLUR_SIGMA} \
         ${INPUTFILE}" >> $SCRIPTFILE
done
done #for z in inputdir

#Need to turn this off so that when the swarmjob
#returns an array, it doesn't return null instead
shopt -u nullglob

#autorun command, so will run the swarm commands automatically
#as well as submit the email scripts properly.
if [ "$run" == "1" ]
then
        cd ${BASEDIR}
        #SWARMID=`swarm -f $SCRIPTFILE --module 3B --jobarray --autobundle`
        #echo $SWARMID
        #BIOIND=`expr index $SWARMID ".bio"`
        #let "BIOIND-=1"
        #JOBNUM=${SWARMID:0:BIOIND}
        #JOBINFO=`qstat -r $JOBNUM | grep ".bio"`
  #SWARMINFO=( $JOBINFO )
  #SWARMJOB=${SWARMINFO[3]} #looks only for the swarm job name
```

```
	SWARMID=`qsub -l nodes=1 -v BDIR=${BASEDIR},SDIR=${SCRIPTFILE}
reservoir.bat`

	#Runs the script that parses all the result files into
	#something useful (for the next script)
	QSUBID=`qsub -l nodes=1 -v
BDIR=${BASEDIR},RDIR=${DATADIR},RNAME="${OUTNAME}_*" \
	-W depend=afterany:${SWARMID} parseResultsMulti.bat`

	#Runs the script that generates the tiff images from
	#the parsed results files
	if [ "$DSPIM" == "y" ]
	then
	MAKEID=`qsub -l nodes=1,matlab-image=1 -v
BDIR=${BASEDIR},DDIR=${DATADIR},IMDIR="Image_files",WIDTH=${XSIZE},HEIG
HT=${YSIZE} \
	-W depend=afterany:${QSUBID} makeStacks.bat`
	else
	MAKEID=`qsub -l nodes=1,matlab-image=1 -v
BDIR=${BASEDIR},DDIR=${DATADIR},IMDIR="Image_files",WIDTH=${XSIZE},HEIG
HT=${YSIZE} \
	-W depend=afterany:${QSUBID} makeImage.bat`
	fi
	#submit email script so that you recieve an
	#email once the analysis has completed
	qsub -l nodes=1 -v JOBDIR=${RESULTSDIR}  -W
depend=afterany:${MAKEID} email.bat
fi
```

The text below can be used to make the "**parseResultsMulti.bat**" file.

```
#!/bin/bash
#
#parseResultsMulti.bat
#
#Filter out the points outside of dilation zone
#and prepares the data for display in ImageJ
#
#SET VARIABLES HERE##############################
#Your base data directory
BASEDIR=/data/your_base_directory
#Results directory
RESULTS=your_data_directory
#Name of the output file for ImageJ
OUTNAME=image
#Template of the results files
RESULTSNAME=results_*
#END OF VARIABLES#################################

shopt -s nullglob

#Variables that will change based on input from
#the autorun version of the nGenScript
#Will only change if provided externally otherwise
#defaults to variables set earlier
BASEDIR=${BDIR:-$BASEDIR}
RESULTS=${RDIR:-$RESULTS}
OUTNAME=${INAME:-$OUTNAME}
RESULTSNAME=${RNAME:-$RESULTSNAME}

#Set directories based on base directories from above
RESULTSDIR=${BASEDIR}/${RESULTS}
FINALFILE=${RESULTSDIR}/${OUTNAME}.txt
IMDIR=${RESULTSDIR}/Image_files

if [ -d "$IMDIR" ]
then
        rm -r $IMDIR
fi

mkdir $IMDIR

#Setup the file that contains ROI info
```

```
CORNERS=${IMDIR}/corners.txt
CSV=${IMDIR}/corners.csv
printf "Plane\tMinX\tMinY\tMaxX\tMaxY\n" >> $CORNERS
printf "Plane,MinX,MinY,MaxX,MaxY\n" >> $CSV

#Iterate through all the folders in the directory
for z in $RESULTSDIR/*/
do
SUBRESULTS=${z}results
RESULTSFILE=${SUBRESULTS}/${RESULTSNAME}

#IMDIR won't contain a results folder, so skip it
#if it exists
if [[ ! -d "$SUBRESULTS" ]]
then
        echo "$z no results"
        continue
fi

#Check if there are any files in the results folder
#to see if you need to skip it
FILES=( $RESULTSFILE )
FILELEN=${#FILES[@]}
if [ $FILELEN == "0" ]
then
        echo "$z contains no results files"
        continue
fi

#Extract the plane name from the directory
#to add to txt files
PLANENAME=${z#${RESULTSDIR}}
NAMELEN=${#PLANENAME}
let "NAMELEN-=2"
PLANECLEAN=${PLANENAME:1:${NAMELEN}}
FINALFILE=${IMDIR}/${PLANECLEAN}_image.txt
OUTFILE=${SUBRESULTS}/coordinates.txt
PIXELFILE=${SUBRESULTS}/pixels.txt

if [ -e "$OUTFILE" ]
then rm $OUTFILE
fi

if [ -e "$PIXELFILE" ]
then rm $PIXELFILE
fi
```

```
BOUNDMINX=9999
BOUNDMINY=9999
BOUNDMAXX=0
BOUNDMAXY=0

CNT=0
for f in $RESULTSFILE
do

        #Search through pixels line to take only
        #ROI bounds (first and last pairs)
        PIXELS=`grep PIXELS ${f}`
        SPLIT=( $PIXELS )
        PIXLEN=${#SPLIT[@]}
        MINX=${SPLIT[1]}
        MINY=${SPLIT[2]}
        MAXX=${SPLIT[PIXLEN-2]}
        MAXY=${SPLIT[PIXLEN-1]}

        #Keep track of this plane's bounds for
        #later reconstruction
        if [ $MINX -lt $BOUNDMINX ]
        then
                BOUNDMINX=$MINX
        fi
        if [ $MINY -lt $BOUNDMINY ]
  then
    BOUNDMINY=$MINY
  fi
        if [ $MAXX -gt $BOUNDMAXX ]
  then
    BOUNDMAXX=$MAXX
  fi
        if [ $MAXY -gt $BOUNDMAXY ]
  then
    BOUNDMAXY=$MAXY
  fi

        #Parse through results file to get spot
        #coordinates. Slightly modified from Edward
        #Rosten's code so that points outside of the
        #ROI are removed before stitching everything
        #back together
        awk -v minX=$MINX \
        -v minY=$MINY \
```

```
        -v maxX=$MAXX \
        -v maxY=$MAXY \
        '/PASS/{for(i=2; i <=NF; i+=4){
                x=$(i+2)
                y=$(i+3)
                if( x>=minX-0.5 && x<=maxX+0.5 && y>=minY-0.5 && y<=maxY+0.5 )
                print "PASS0: 1 1", x, y
                }
        }' \
        ${f} >> ${OUTFILE}

        let "CNT+=1"

done #for f in resultsfile

#fail safe, in case something didn't work earlier
if [ "$CNT" -eq "0" ]
then
        echo "$z fail-safe kicked in"
        continue
fi

#Write the corner coordinates to text files
printf "${PLANECLEAN}\t%d\t%d\t%d\t%d\n" $BOUNDMINX $BOUNDMINY
$BOUNDMAXX $BOUNDMAXY >> $CORNERS
printf "${PLANECLEAN},%d,%d,%d,%d\n" $BOUNDMINX $BOUNDMINY
$BOUNDMAXX $BOUNDMAXY >> $CSV

#List of all pixels in ROI for the ImageJ reconstruction
printf "PIXELS " > ${PIXELFILE}

for ((y=BOUNDMINY ; y <= BOUNDMAXY ; y++))
do
        for ((x=BOUNDMINX; x <= BOUNDMAXX; x++))
        do
                printf "%d %d " $x $y >> ${PIXELFILE}
        done
done

printf "\n" >> ${PIXELFILE}

#Combine pixels with the spot coordinates
cat $PIXELFILE > $FINALFILE
cat $OUTFILE >> $FINALFILE

done #for z in resultsdir
```

The text below can be used to make the "**reservoir.bat**" file.

```
#!/bin/bash
#BDIR=/data/HROI
#SDIR=/data/HROI/Victor_Test/swarmfile.txt

cd $BDIR

CORESLINE=`./showq.bat`
SPLIT=( ${CORESLINE} )
TOTAL=${SPLIT[3]}

let "RESSIZE=3980-TOTAL"
INJECT=1000
THRESH=2000
WAITTIME=1h

LINES=`cat $SDIR | wc -l`
CURRENT=RESSIZE
if [ "$LINES" -lt "$RESSIZE" ]
then
        SWARMID=`swarm -f $SDIR --module 3B --jobarray`
else
        let "RESONE=RESSIZE+1"
        sed -n "1,${RESSIZE}p;${RESONE}q" $SDIR > swarmpart.txt
        swarm -f swarmpart.txt --module 3B --jobarray
        let "REMAIN=LINES-CURRENT"
        let "CURRENT+=1"
        while [ "$REMAIN" -gt "0" ]
        do
                sleep $WAITTIME
                CORESLINE=`./showq.bat`
                SPLIT=( ${CORESLINE} )
                TOTAL=${SPLIT[3]}
                if [ "$TOTAL" -lt "$THRESH" ]
                then
                        if [ "$REMAIN" -lt "$INJECT" ]
                        then
                                sed -n "${CURRENT},${LINES}p" $SDIR > swarmpart.txt
                                SWARMID=`swarm -f swarmpart.txt --module 3B --
jobarray`
                                REMAIN=0
                        else
                                let "INJONE=INJECT-1"
                                let "NEXT=CURRENT+INJONE"
```

```
                                let "STOP=CURRENT+INJECT"
                                sed -n "${CURRENT},${NEXT}p;${STOP}q" $SDIR >
swarmpart.txt
                                swarm -f swarmpart.txt --module 3B --jobarray
                                CURRENT=$STOP
                                let "REMAIN-=INJECT"
                        fi
                fi
        done
fi

if [ -e swarmpart.txt ]
then
        rm swarmpart.txt
fi

INDEX=`expr index $SWARMID "\["`
if [ $INDEX == "0" ]
then
        INDEX=`expr index $SWARMID "."`
fi
let "INDEX-=1"
SWARMID=${SWARMID:0:INDEX}
RUNNING=`qselect -u $USER | grep $SWARMID`
while [ -n "$RUNNING" ]
do
        sleep $WAITTIME
        RUNNING=`qselect -u $USER | grep $SWARMID`
done
```

The text below can be used to make the "**showq.bat**" file.

```
#!/bin/bash

QUEUE=`qstat -t -u ${USER} | grep ${USER}`
JOBNUM=`qstat -t -u ${USER} | grep ${USER} | wc -l`
SPLIT=( $QUEUE )
JOBS=""
IND=0
ACTIVE=0
TOTAL=0
for ((a = 0; a<JOBNUM; a++))
do
        let "IND=a*11"
        JOB=${SPLIT[$IND]}
        BIOIND=`expr index $JOB .`
        let "BIOIND-=1"
        NUMBER=${JOB:0:$BIOIND}
        JOBS+="${NUMBER} "
done

SPLIT=( $JOBS )

for ((a = 0; a<JOBNUM; a++))
do
        JOB=${SPLIT[$a]}
        STATELINE=`qstat -f $JOB | grep job_state`
        STATELINE=( ${STATELINE} )
        STATE=${STATELINE[2]}
        RESLINE=`qstat -f $JOB | grep Resource_List.cores`
        RESLINE=( ${RESLINE} )
        CORES=${RESLINE[2]}
        if [ "$STATE" != "X" ]
        then
                let "TOTAL+=CORES"
        fi
        if [ "$STATE" == "R" ]
        then
                let "ACTIVE+=CORES"
        fi
done

echo "Active $ACTIVE Total $TOTAL"
```