FALCON: fast and unbiased reconstruction of high-density super-resolution microscopy data

Junhong Min[1], Cédric Vonesch[2*], Hagai Kirshner[2*], Lina Carlini[3*], Nicolas Olivier[3,4], Seamus Holden[3], Suliana Manley[3], Jong Chul Ye[1], Michael Unser[2]

[1] *Department of Bio and Brain Engineering, KAIST, Daejeon, Republic of Korea*

[2] *Institute of Microengineering, EPFL, Switzerland*

[3] *Institute of the Physics of Biological Systems, EPFL, Switzerland*

[4] *Department of Physics, King's College London, UK*

** Authors with equal contribution*

1 Supplementary Note

1.1 General approach

1.1.1 Least-squares formulation

At a given time instant, the activated fluorescent probes can be described by a collection of point sources:

$$f(\mathbf{x}) = \sum_{k=1}^{K} c_k \, \delta(\mathbf{x} - \mathbf{x}_k).$$

Here, K is the number of the point sources, $\mathbf{x}_k$ is the position of the k-th source and c_k is its brightness in terms of the number of emitted photons. Formally, the camera image is obtained by convolving the sources with the PSF h of the microscope, adding a background signal b and introducing a random noise distortion represented by $\mathcal{N}$:

$$g[\mathbf{n}] \sim \mathcal{N}\left(b + \sum_{k=1}^{K} c_k \, h(\mathbf{n} - \mathbf{x}_k)\right)$$

where n denotes n-th pixel of the camera image. Given a set of measurements $g[\mathbf{n}]$, we would like to find the parameters K, c_k and $\mathbf{x}_k$ that minimize the least-squares functional

$$\sum_{\mathbf{n}} \left(g[\mathbf{n}] - b - \sum_{k=1}^{K} c_k \, h(\mathbf{n} - \mathbf{x}_k)\right)^2 . \tag{1}$$

However the dependence of this functional on $\mathbf{x}_k$ is non-linear, which requires computationally expensive derivative of h at each iteration.

1.1.2 Taylor-series approximation

We thus propose to modify the problem by introducing a sub-pixel grid of the form $\mathbb{Z}^2/M$, where M is an integer oversampling factor. A given source position $\mathbf{x}_k$ can be expressed as the sum of the closest m-th point on the sub-pixel grid $\mathbf{m}_k$ and a small displacement $\boldsymbol{\epsilon}_k$:

$$\mathbf{x}_k = \frac{\mathbf{m}_k}{M} + \boldsymbol{\epsilon}_k. \tag{2}$$

Here, we consider linear approximation of h by Taylor series expansion as follows:

$$\begin{aligned} h(\mathbf{x}-\mathbf{c}) &= h(\mathbf{c}) + (\mathbf{x}-\mathbf{c})^T \nabla h(\mathbf{c}) + r(\mathbf{x}-\mathbf{c}) \\ h(\mathbf{x}-\mathbf{c}-\boldsymbol{\epsilon}) &= h(\mathbf{c}) + (\mathbf{x}-\mathbf{c}-\boldsymbol{\epsilon})^T \nabla h(\mathbf{c}) + r(\mathbf{x}-\mathbf{c}-\boldsymbol{\epsilon}) \\ &= (h(\mathbf{c}) + (\mathbf{x}-\mathbf{c})^T \nabla h(\mathbf{c}) + r(\mathbf{x}-\mathbf{c})) - \boldsymbol{\epsilon}^T \nabla h(\mathbf{c}) + (r(\mathbf{x}-\mathbf{c}-\epsilon) - r(\mathbf{x}-\mathbf{c})) \\ &\simeq h(\mathbf{x}-\mathbf{c}) - \boldsymbol{\epsilon}^T \nabla h(\mathbf{c}) \end{aligned}$$

where r denotes higher-order terms more than second-order and T is transpose. We can then apply a first-order Taylor approximation for every shifted PSF:

$$h(\mathbf{n}-\mathbf{x}_k) \;\simeq\; h(\mathbf{n}-\frac{\mathbf{m}_k}{M}) - \boldsymbol{\epsilon}_k^T \nabla h(\mathbf{n}-\frac{\mathbf{m}_k}{M}).$$

Inserting the latter expression into (1) and extending the inner sum to the entire sub-pixel grid leads to the functional

$$J(c,\boldsymbol{\epsilon}) = \sum_{\mathbf{n}} \left(g[\mathbf{n}] - b - \sum_{\mathbf{m}} c[\mathbf{m}] \left[h(\mathbf{n}-\frac{\mathbf{m}}{M}) - \boldsymbol{\epsilon}[\mathbf{m}]^T \nabla h(\mathbf{n}-\frac{\mathbf{m}}{M}) \right] \right)^2 , \tag{3}$$

which becomes quadratic with respect to c or $\boldsymbol{\epsilon}$ respectively. Note that determining K and $\mathbf{m}_k$ now amounts to identifying the support of the non-zero coefficients $c[\mathbf{m}]$.

1.2 FALCON: FAst Localization algorithm based on CONtinuous-spatial formulation

Our algorithm can be divided into three main stages (**Fig. 1**). The first two are primarily aimed at determining the support of $c[\mathbf{m}]$. They consist in deconvolving the measured image on the sub-pixel grid using two subsequent types of sparsity constraints. Then, detection of local-maxima can be performed to isolate the sources and to compute initial estimates for their brightness and locations. These quantities are then refined iteratively in the later stage of the algorithm, where the position coordinates are allowed to take values over a continuum.

1.2.1 Deconvolution with a space-domain sparsity prior

If we neglect the displacements $\boldsymbol{\epsilon}[\mathbf{m}]$ in (3), we are left with a deconvolution problem where the object is known to comprise of a sparse set of sources. Thus, it is natural to first try to minimize

$$J_1(c) = J(c, \mathbf{0}) + \phi(c) + R_{\mathbb{R}^+}(c), \tag{4}$$

where

$$\phi(c) = \sum_{\mathbf{m}} w[\mathbf{m}] \, |c[\mathbf{m}]|$$

is a weighted ℓ_1 norm that favors sparse source distributions on the sub-pixel grid and $R_{\mathbb{R}^+}(c)$ constrains the coefficients to be positive, which is defined as

$$R_\Omega(c) = \begin{cases} 0 & \text{if } \forall \mathbf{m}, c[\mathbf{m}] \in \Omega, \\ +\infty & \text{otherwise.} \end{cases}$$

As we shall see in Section 1.3, each weight $w[\mathbf{m}]$ determines a threshold below which the coefficient $c[\mathbf{m}]$ is set to zero. In our experiments, we obtained good results by choosing it proportional to a local uncertainty of background noise:

$$w[\mathbf{m}] = \kappa \times \sqrt{b[\mathbf{m}] \times \gamma_{\text{EM}}}.$$

Here $b[\mathbf{m}]$ is the local background signal, which is interpolated from the background estimate described in Section 1.4. γ_{EM} denotes the EM-CCD gain factor; it is typically equal to 1.4 (EM gain on) or 1 (EM gain off). In general, the noise mainly follows Poisson statistics, hence the square root to estimate its the standard deviation of the background noise. We investigated various choices of κ over a wide range of simulation settings (**Supplementary Fig. 6**); based on these results we used $\kappa = 2$ in all of our experiments.

1.2.2 Least-squares deconvolution with a fixed spatial support

In our synthetic experiments (**Fig. 6 and Supplementary Fig. 4**), we observed that the first stage of our algorithm resulted in clusters of non-zero pixels around the true particle locations. However, the positions of the local maxima of these clusters appeared to be biased, in the sense that the estimated distances among nearby particles were usually smaller than expected. This spatial bias seems to be a side-effect of the fact that the (weighted) ℓ_1 norm biases the pixel intensities towards zero.

We thus decided to incorporate a second stage in our algorithm where the sparsity constraint was replaced by a support constraint:

$$J_2(c) = J(c, \mathbf{0}) + S_{\Omega_1}(c) + R_{\mathbb{R}^+}(c). \tag{5}$$

Here

$$S_\Omega(c) = \begin{cases} 0 & \text{if } c[\mathbf{m}] \neq 0 \text{ only for } \mathbf{m} \in \Omega, \\ \\ +\infty & \text{otherwise.} \end{cases}$$

and Ω_1 is the support of the coefficients $c[\mathbf{m}]$ returned by the first stage of the algorithm. The support is defined as the set of pixels whose value are above small threshold (e.g. 5% of the maximum image intensity). Minimizing (5) amounts to a simple least-squares fit with a positivity and support constraints, but without the typical bias induced by the ℓ_1 penalty.

1.2.3 Continuous-domain refinement of positions and coefficients

Let $\tilde{c}[\mathbf{m}]$ denote the deconvolved image obtained in the previous stage. Based on this outcome, we can compute a first estimate of the coefficients $c[\mathbf{m}]$ and of the displacements $\boldsymbol{\epsilon}[\mathbf{m}]$, which we will then refine.

We proceed as follows. The number of sources K is first set as the number of local maxima in $\tilde{c}[\mathbf{m}]$. Let $\widetilde{\mathbf{m}}_k$ be the positions of these local maxima. Then, the position of the k-th source is estimated as a center of mass in a small neighborhood around the corresponding local maximum:

$$\mathbf{x}_k = \frac{\sum_{\mathbf{m} \in N(\widetilde{\mathbf{m}}_k)} \tilde{c}[\mathbf{m}]\mathbf{m}}{\sum_{\mathbf{m} \in N(\widetilde{\mathbf{m}}_k)} \tilde{c}[\mathbf{m}]}.$$

In our implementation the neighbourhood $N(\widetilde{\mathbf{m}}_k)$ was chosen to be a 3×3 window centered on $\widetilde{\mathbf{m}}_k$. This gives an initial estimate for $\mathbf{m}_k$ and $\boldsymbol{\epsilon}_k$ according to (2). From there on the support of the non-zero coefficients is fixed as

$$\Omega_2 = \{\mathbf{m}_k \,|\, k = 1, \ldots, K\}.$$

The initial value of these coefficients is simply given by the sum of the previous-stage coefficients in the local neighbourhood:

$$c[\mathbf{m}_k] = \sum_{\mathbf{m} \in N(\tilde{\mathbf{m}}_k)} \tilde{c}[\mathbf{m}].$$

We can then return to the bivariate functional (3), augmented with three regularization terms:

$$J_3(c, \boldsymbol{\epsilon}) = J(c, \boldsymbol{\epsilon}) + S_{\Omega_2}(c) + R_{\mathbb{R}^+}(c) + R_{[-\Delta,\Delta]^2}(\boldsymbol{\epsilon}). \qquad (6)$$

Here S_{Ω_2} and $R_{\mathbb{R}^+}$ express the support and positivity constraints on the coefficients. The additional term $R_{[-\Delta,\Delta]^2}$ simply prevents the displacement variables from becoming too large (in which case the Taylor approximation would become too inaccurate). In our implementation the distance parameter Δ is chosen to be $1/M$ in each lateral direction.

1.3 Implementation of FALCON

In general, (4), (5) and (6) involve millions of unknowns, which restricts the structure of the corresponding minimization algorithms. In the context of localization microscopy, sparsity-regularized functionals such as (4) and (5) have been tackled using linear programming on small image patches[1] or expectation-maximization (EM) methods[2]. Here we use the alternating direction method of multipliers (ADMM), which has recently been introduced in the field of image recovery and converges significantly faster[3]. To minimize (6) we have designed a custom alternating minimization scheme, taking advantage of the fact that $J(c, \boldsymbol{\epsilon})$ is essentially quadratic when one of its arguments is fixed.

Note that the step size of the sub-pixel grid has to be properly selected to balance computation complexity and localization performance; empirically between 60 nm and 20 nm. In all our experiments, the size of the CCD pixels was 100 nm and we used an oversampling factor of $M = 3$, i.e. a sub-pixel size of 33 nm.

1.3.1 Sparse deconvolution using ADMM

Using matrix notations, the sparsity-based cost functions (4) and (5) have the following general structure:

$$\|g - b - DHc\|^2 + \psi(c),$$

where g is the measurement, b is the background signal, D is a down-sampling matrix, H is a convolution matrix corresponding to the PSF h and ψ is a sparsity and positivity-enforcing regularizers. In practice, we introduce an auxiliary variable v and apply ADMM to minimize $\|g - b - DHc\|^2 + \psi(v)$ under the constraint $v = c$. This leads to the following iterative update scheme:

$$\begin{aligned} c^{(i+1)} &= \arg\min_c \|g - b - DHc\|^2 + \mu\|c - v^{(i)} - d^{(i)}\|^2; \\ v^{(i+1)} &= \arg\min_v \psi(v) + \mu\|c^{(i+1)} - d^{(i)} - v\|^2; \\ d^{(i+1)} &= d^{(i)} + c^{(i+1)} - v^{(i+1)}. \end{aligned}$$

Note that the value of μ affects the convergence rate of the optimization. However, the results is not sensitive to the choice of μ in the range of $0.1 - 0.01$. We have chosen $\mu = 0.05$ as a universal setting for all our experiments.

More explicitly, the update of c is given by the following closed-form expression:

$$c^{(i+1)} = (H^T D^T DH + \mu I)^{-1}(H^T D^T (g - b) - \mu(v^{(i)} + d^{(i)})).$$

Note that the matrix inversion in this equation can be carried out directly using the Fast Fourier Transform algorithm. The update of v can be written as $v^{(i+1)} = \mathcal{M}_{\psi/\mu}(c^{(i+1)} - d^{(i)})$ where $\mathcal{M}_{\psi/\mu}$ is the so-called Moreau proximal mapping corresponding to ψ/μ. The two proximal mappings that are relevant to the functionals (4) and (5) are defined by

$$\mathcal{M}_{(\phi+R_{\mathbb{R}^+})/\mu}(v)[\mathbf{m}] = \begin{cases} v[\mathbf{m}] - w[\mathbf{m}]/\mu & \text{if } v[\mathbf{m}] > w[\mathbf{m}]/\mu, \\ 0 & \text{otherwise;} \end{cases} \tag{7}$$

$$\mathcal{M}_{(S_\Omega+R_{\mathbb{R}^+})}(v)[\mathbf{m}] = \begin{cases} v[\mathbf{m}] & \text{if } v[\mathbf{m}] > 0 \text{ and } \mathbf{m} \in \Omega, \\ 0 & \text{otherwise.} \end{cases}$$

1.3.2 Continuous-domain refinement using alternating minimization

When one of its arguments is fixed (say $\boldsymbol{\epsilon} = \boldsymbol{\epsilon}^{(i)}$ or $c = c^{(i+1)}$), the cost function for the refinement stage satisfies the following equalities (up to some additive constants):

$$J_3(c, \boldsymbol{\epsilon}^{(i)}) = \|g - b - (DH - D\nabla H\mathcal{E}^{(i)})c\|^2 + S_{\Omega_2}(c) + R_{\mathbb{R}^+}(c); \tag{8}$$

$$J_3(c^{(i+1)}, \boldsymbol{\epsilon}) = \|g - b - DHc^{(i+1)} + D\nabla HC^{(i+1)}\boldsymbol{\epsilon}\|^2 + R_{[-\Delta,\Delta]^2}(\boldsymbol{\epsilon}). \tag{9}$$

Here ∇H is a matrix that corresponds to the gradient of the PSF and $\mathcal{E}^{(i)}$ and $C^{(i+1)}$ are matrices that depend on $\boldsymbol{\epsilon}^{(i)}$ and $c^{(i+1)}$, respectively.

Equations (8) and (9) define quadratic minimization problems with convex constraints. Thus, to minimize J_3, we propose an algorithm that alternates between two projected gradient descents:

$$\begin{aligned} c^{(i+1)} &= \mathcal{M}_{(S_\Omega+R_{\mathbb{R}^+})}\left[c^{(i)} - \tau^{(i)}(DH - D\nabla H\mathcal{E}^{(i)})^T\Big((DH - D\nabla H\mathcal{E}^{(i)})c^{(i)} + b - g\Big)\right]; \\ \boldsymbol{\epsilon}^{(i+1)} &= \mathcal{M}_{R_{[-\Delta,\Delta]^2}}\left[\boldsymbol{\epsilon}^{(i)} - \theta^{(i)}(D\nabla H(C^{(i+1)}))^T\Big(D\nabla HC^{(i+1)}\boldsymbol{\epsilon}^{(i)} - DHc^{(i+1)} - b + g\Big)\right]. \end{aligned}$$

Here the first projection operator has been defined in (7); the second is defined by

$$\mathcal{M}_{R_{[-\Delta,\Delta]^2}}(\boldsymbol{\epsilon})[\mathbf{m}] = \begin{cases} \boldsymbol{\epsilon}[\mathbf{m}] & \text{if } \boldsymbol{\epsilon}[\mathbf{m}] \in [-\Delta,\Delta]^2, \\ \Delta\epsilon[\mathbf{m}]/\|\boldsymbol{\epsilon}[\mathbf{m}]\| & \text{otherwise.} \end{cases}$$

At every iteration, the step sizes $\tau^{(i)}$ and $\theta^{(i)}$ are chosen so as to minimize the quadratic term in (8) and (9), respectively; this simply amounts to finding the roots of second-degree polynomials.

1.4 Background estimation of FALCON

In general, the background fluorescence signal varies smoothly over the field of view and the estimation of background is important to improve localization performance. We applied iterative wavelet transform method used in spectroscopy [4] to the step 1&2. In detail, we get a residual image by subtracting estimated photon contributions of molecules from a camera image, and then, the background is iteratively estimated from the lowest frequency wavelets bands of the residual image as shown in (**Supplementary Fig. 1**). In each iteration, all values of the residual image above current estimated background level are clipped. In the step 1&2, estimation of probes c and background b are conducted alternatively. The weighting factor w in the step 1 is also updated at every update of the background. More specifically, we used wavelet decomposition up to 8th level using Daubechies-6 wavelets. For an initial background level, we took a minimum value of median filtered camera image using 5×5 kernel.

We tested localization performances by chaining wavelet level from 6 to 9 with different number of iteration of the method (**Supplementary Fig. 7 and 8**). This simulation studies show that background is well estimated and localization performances are consistent. For all the other simulations in this paper, wavelet decomposition level is fixed at eight, and three iterations are used in the step 1 and one iteration is used in the step 2.

1.5 Other localization algorithms

1.5.1 Gaussian fitting by Non-linear least-squares minimization

In every local peak, we crop a small window centred on the peak. The cropped image is fitted with an two-dimensional symmetric Gaussian function:

$$I(x,y) = \frac{c}{2\pi\sigma^2} exp^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)} + b$$

where c is an estimated number of photon counts from a single molecule. σ is width of Gaussian PSF and b is an uniform background. All variables are optimized via non-linear least-squares minimization[5,6]. Using shape-based filtering, the localized molecules are excluded for all our experiments if min(σ) $<$ 1 pixel, max(σ) $>$ 2 pixels. The window size is 9 × 9 pixels in **Fig. 4 and Supplementary Fig. 13**.

1.5.2 DAOSTORM

DAOSTORM[7] performs simultaneous multi-emitter fitting of an experimentally derived model PSF to the input data, iteratively identify new candidate localizations from a residuals image of the fits to the data. Fitting to the input data using new candidate localizations is repeated until new candidate localizations fall below a target signal to noise ratio, or until a maximum number of iterations is reached, whichever occurs first. Model PSFs were derived from low density frames of the dataset (simulations) or low density data imaged under the same experimental conditions (Tubulin Alexa-647 data).

The key parameters in the DAOSTORM algorithm are: sigmaBG, the estimated background noise; nSigma, the SNR detection threshold; and nIter, the maximum number of iterations. sigmaBG was estimated from empty subregions of images in each dataset. nSigma was chosen by running DAOSTORM several times on single frames of the datasets, and visually estimating the value of nSigma which gave the best results. For the Tubulin Alexa-647 data, sigmaBG = 17, nSigma = 5. For the high SNR simulated data, sigmaBG = 10, nSigma = 10. For the low SNR simulated data, sigmaBG = 4, nSigma = 5. For all datasets, nIter = 4.

1.5.3 Compressed sensing STORM (CSSTORM)

In CSSTORM[1], the localization is formulated as a convex optimization with weighted ℓ_1 norm penalty for sparsity priors, which is solved by linear programming:

$$\textit{Minimize } \; w^T c$$

$$\textit{Subject to: } \; c[m] \geq 0 \;\; and \;\; \|g - (Hc + b)\|^2 \leq \alpha \cdot (\sum g[n])^{1/2}$$

where g is a measured camera image, c is molecule distribution in terms of the fluorescence intensity, b is uniform background fluorescence. The matrix H is convolution matrix of PSF and w is weighting vector. The parameter α determines the noise level of the camera image g to be allowed in the optimization. Since linear program requires high computational complexity, the problem is divided into the several sub-problem of the overlapped patches. In the every patch of 12 × 12 μm^2, 14 × 14 μm^2 image is reconstructed on the fine sub-pixel grid of 20 nm, and then only central 0.8 × 0.8 μm^2 area is included in the final solution. The noise parameter α is 1.5 as default setting (without electron multiplying(EM) gain)[1] for all our experiments. For live ER data with EM gain, we used $\alpha = 1.7$ which is slightly lower value than $\alpha = 1.7$ used in [1] in order to detect more molecules.

1.5.4 deconvolution-STORM (deconSTORM)

In deconSTORM[2], deconSTORM uses the maximum likelihood estimator to solve localization problem. Specifically, it is solved by modified Richardson–Lucy deconvolution algorithm imposing sparse prior distribution for every pixel intensities of fine sub-pixel grid of 20 nm. More specifically, deconSTORM approximates switching events of molecules between activated and off state as a Markov process. All state transitions of molecules are assumed to be statistically independent, which allows to update sparse prior distribution for each pixel independently. deconSTORM requires prior knowledge concerning constant background level and probabilities of the state transitions. We set all the required parameters as optimal values in our simulations.

References

1. Zhu, L., Zhang, W., Elnatan, D. & Huang, B. Faster storm using compressed sensing. *Nat. Methods* **9**, 721–723 (2012).

2. Mukamel, E. A., Babcock, H. & Zhuang, X. Statistical deconvolution for superresolution fluorescence microscopy. *Biophys. J.* **102**, 2391–2400 (2012).

3. Afonso, M. V., Bioucas-Dias, J. M. & Figueiredo, M. A. Fast image recovery using variable splitting and constrained optimization. *IEEE Trans. Image Process* **19**, 2345–2356 (2010).

4. Galloway, C., Ru, E. L. & Etchegoin, P. An iterative algorithm for background removal in spectroscopy by wavelet transforms. *Applied spectroscopy* **63**, 1370–1376 (2009).

5. Rust, M. J., Bates, M. & Zhuang, X. Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy STORM. *Nat. Methods* **3**, 793–796 (2006).

6. Betzig, E., Patterson, G. H., Sougrat, R., Lindwasser, O. W., Olenych, S., Bonifacino, J. S., Davidson, M. W., Lippincott-Schwartz, J. & Hess, H. F. Imaging intracellular fluorescent proteins at nanometer resolution. *Science* **313**, 1642–1645 (2006).

7. Holden., S. J., Uphoff., S. & Kapanidis, A. N. DAOSTORM: an algorithm for high-density super-resolution microscopy. *Nat. Methods* **8**, 279–280 (2011).

8. Olivier, N., Keller, D., Rajan, V. S., Gönczy, P. & Manley, S. Simple buffers for 3d storm microscopy. *Biomed. Opt.* **4**, 885–899 (2013).

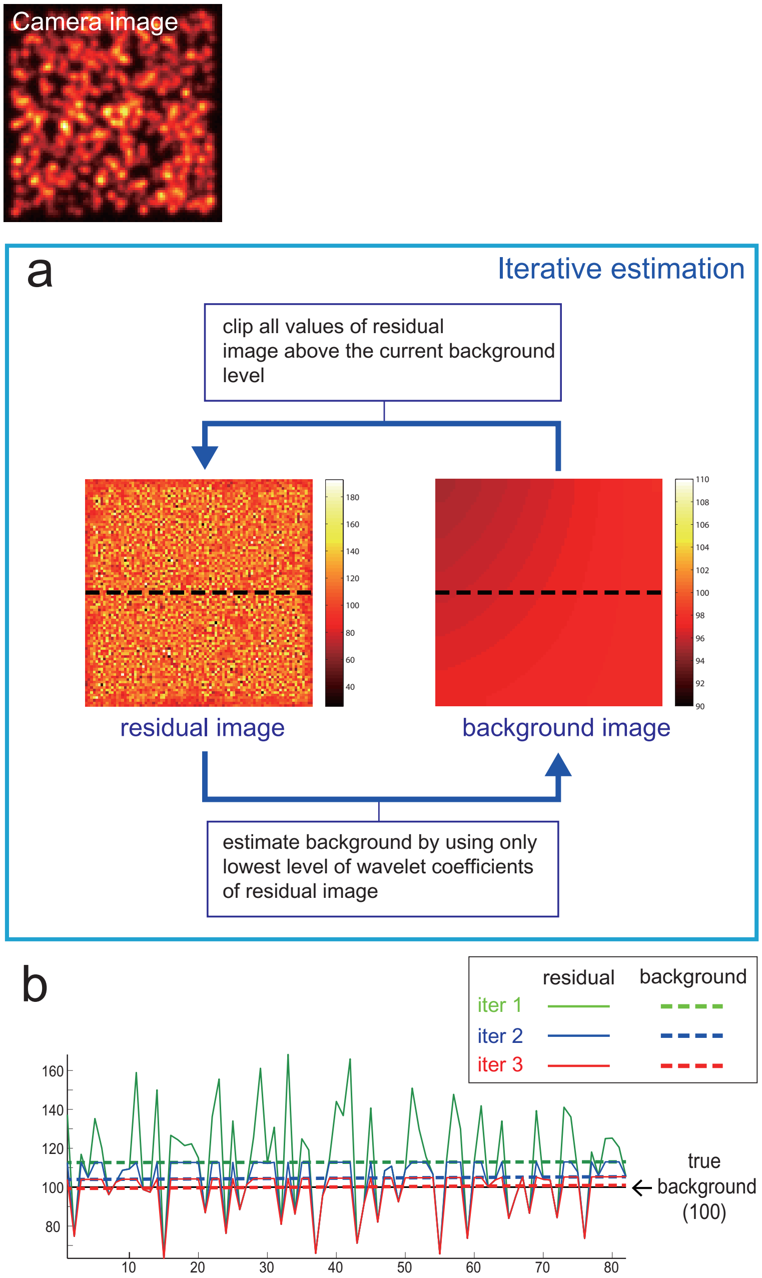

Figure 1 Background estimation of FALCON. (a) schematic diagram of the iterative background estimation. Background is iteratively estimated by using only lowest level of wavelet coefficients of residual image. (b) line profiles on dotted lines in (a) from estimated background and clipped residual image at $1-3$ iteration. The estimated background level at 3 iteration is close to true background level 100. For a detailed description, **see Supplementary text, section 1.4**.

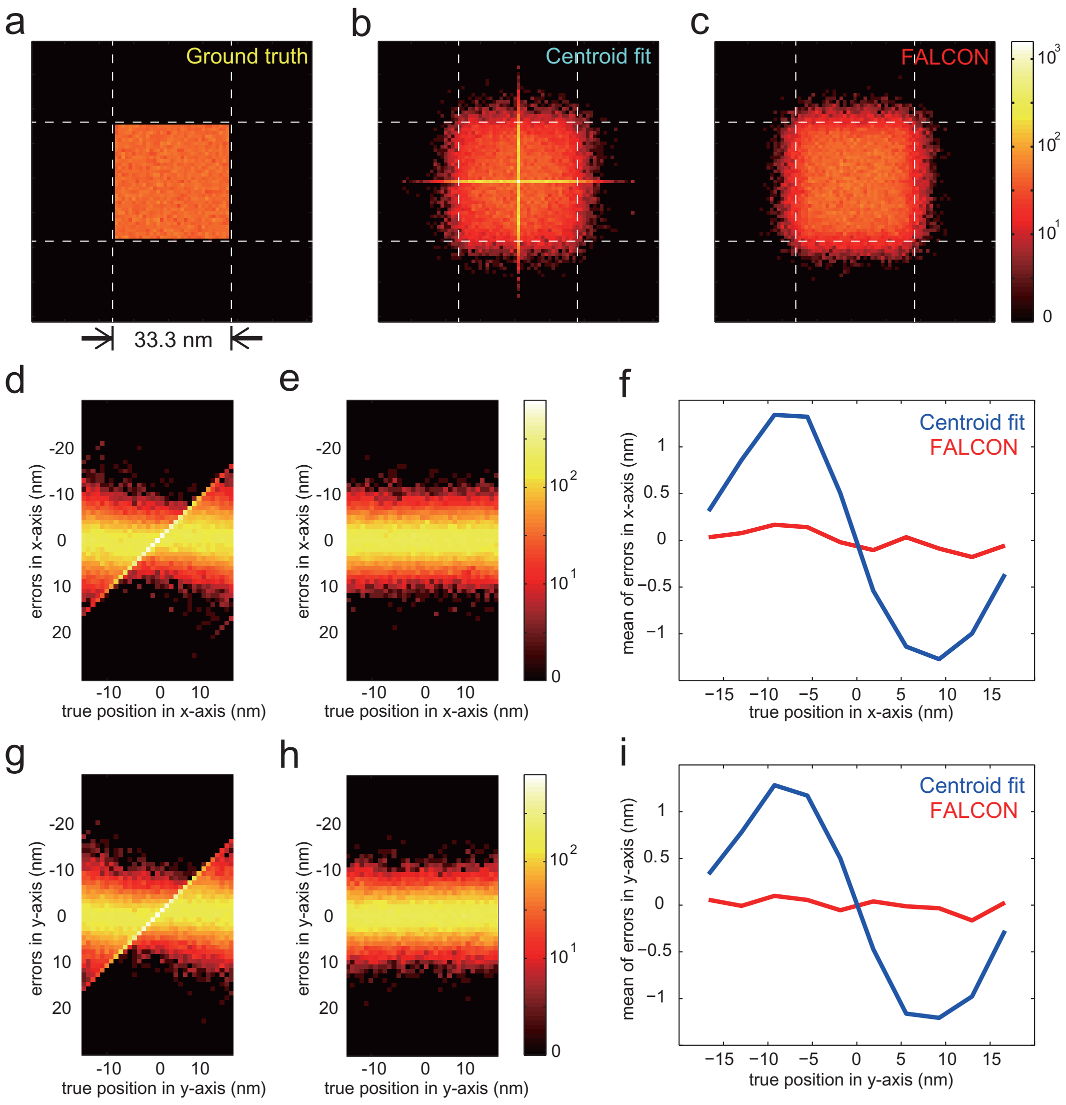

Figure 2 Performance analysis for continuous refinement on 50,000 simulated images with uniform distribution. A single molecule is randomly placed within a single sub-pixel area in each frame. (a) Histogram of the true distribution, (b) Histograms of the centroid fit applied to the deconvolved images by ℓ_1 minimization, (c) Histogram of FALCON. (d,e) Histograms of x-axis errors of centroid fit and FALCON respectively corresponding to positions of true molecules in x-axis, and (f) average errors of the histograms in x-axis. (g,h) Histograms of y-axis errors of centroid fit and FALCON respectively corresponding to positions of true molecules in y-axis, and (i) average errors of the histograms in y-axis.

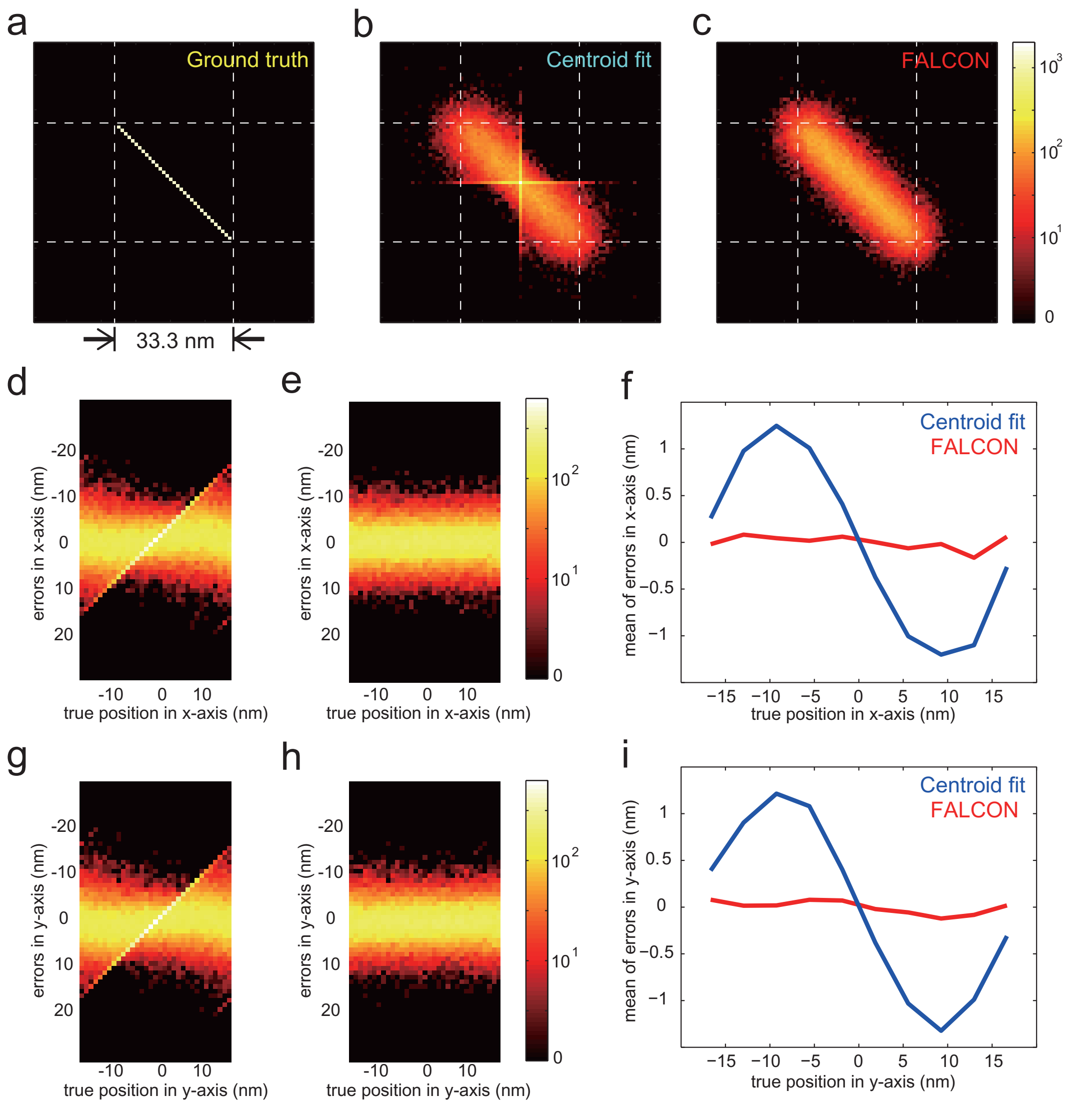

Figure 3 Performance analysis for continuous refinement on 50,000 simulated images of line distributions. A single molecule is randomly placed on a diagonal of a single sub-pixel in each frame. (a) Histogram of the true distribution, (b) Histograms of the centroid fit applied to the deconvolved images by ℓ_1 minimization, (c) Histogram of FALCON. (d,e) Histograms of x-axis errors of centroid fit and FALCON respectively corresponding to positions of true molecules in x-axis, and (f) average errors of the histograms in x-axis. (g,h) Histograms of y-axis errors of centroid fit and FALCON respectively corresponding to positions of true molecules in y-axis, and (i) average errors of the histograms in y-axis.

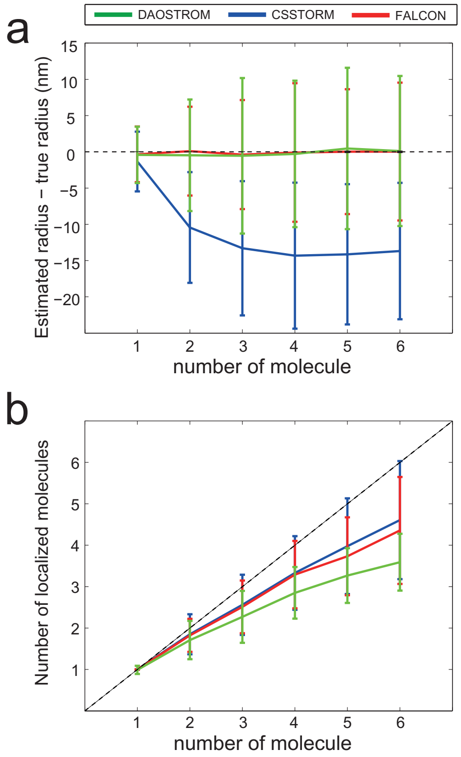

Figure 4 Performance analysis for synthetic ”ring” phantoms with various imaging densities. (a) Differences between true radius and estimated radius for various number of molecules on the circle of 200 nm radius in every simulated image. (b) Numbers of localized molecules by FALCON, DAOSTORM, and CSSTORM. The estimated radius is calculated by averaging the distances of localized molecules to the center of the circle. The error bars indicate standard deviations.

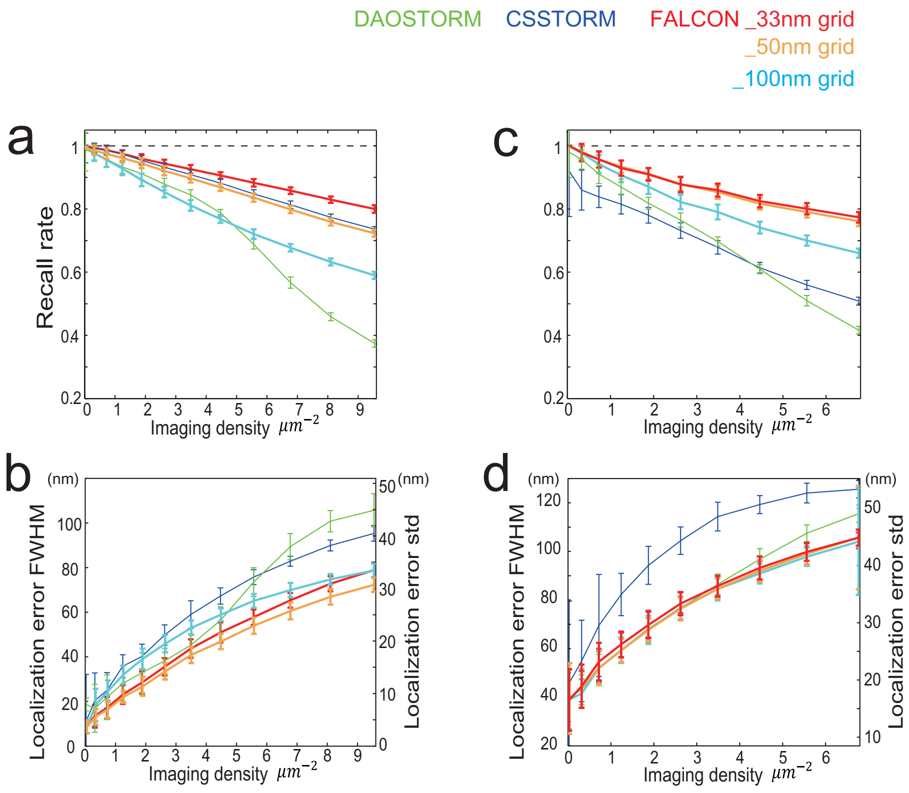

Figure 5 Performances of FALCON with different sub-pixel size for deconvolution in comparison of DAOSTORM and CSSTORM. The sub-pixel size for step1 and 2 of FALCON varies from 33.3 nm (up-sampling factor = 3) to 100 nm (without up-sampling). Simulated analysis on the random distribution of molecules over a wide range of imaging densities with high-photon emission rates: recall rates (a), localization accuracy (b) and low-photon emission rates: recall rates (c), localization accuracy (d). The error bars indicate standard deviations.

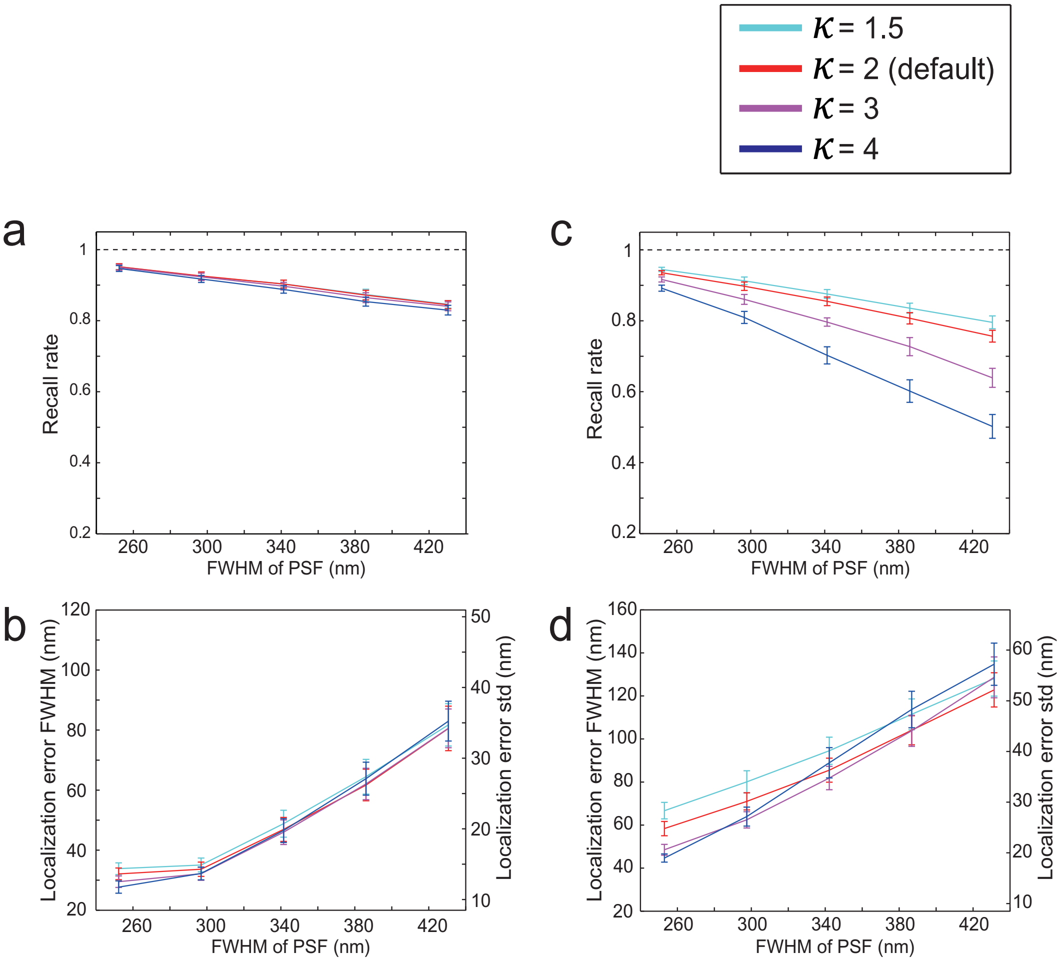

Figure 6 Parameter choice analysis with high & low SNR simulated data. κ determines minimum level of brightness coefficient to be counted in the first deconvolution step of FALCON. (a) Molecular recall rates, (b) Localization accuracy with κ ranging from 1.5 to 4 for high SNR simulated data and (c) Molecular recall rates, (d) Localization accuracy with κ ranging from 1.5 to 4 for low SNR simulated data. Imaging density is fixed to 6 μm^{-2} in all simulated image, and FWHM of PSF varies from 250 nm to 430 nm. The simulated image was generated by the following settings: on average, 5000 photons per molecule (with 2000 s.d.) and 100 uniform background photons for high SNR, and on average, 350 photons per molecule (with 70 s.d.) and 10 uniform background photons for low SNR. The error bars indicate standard deviations.

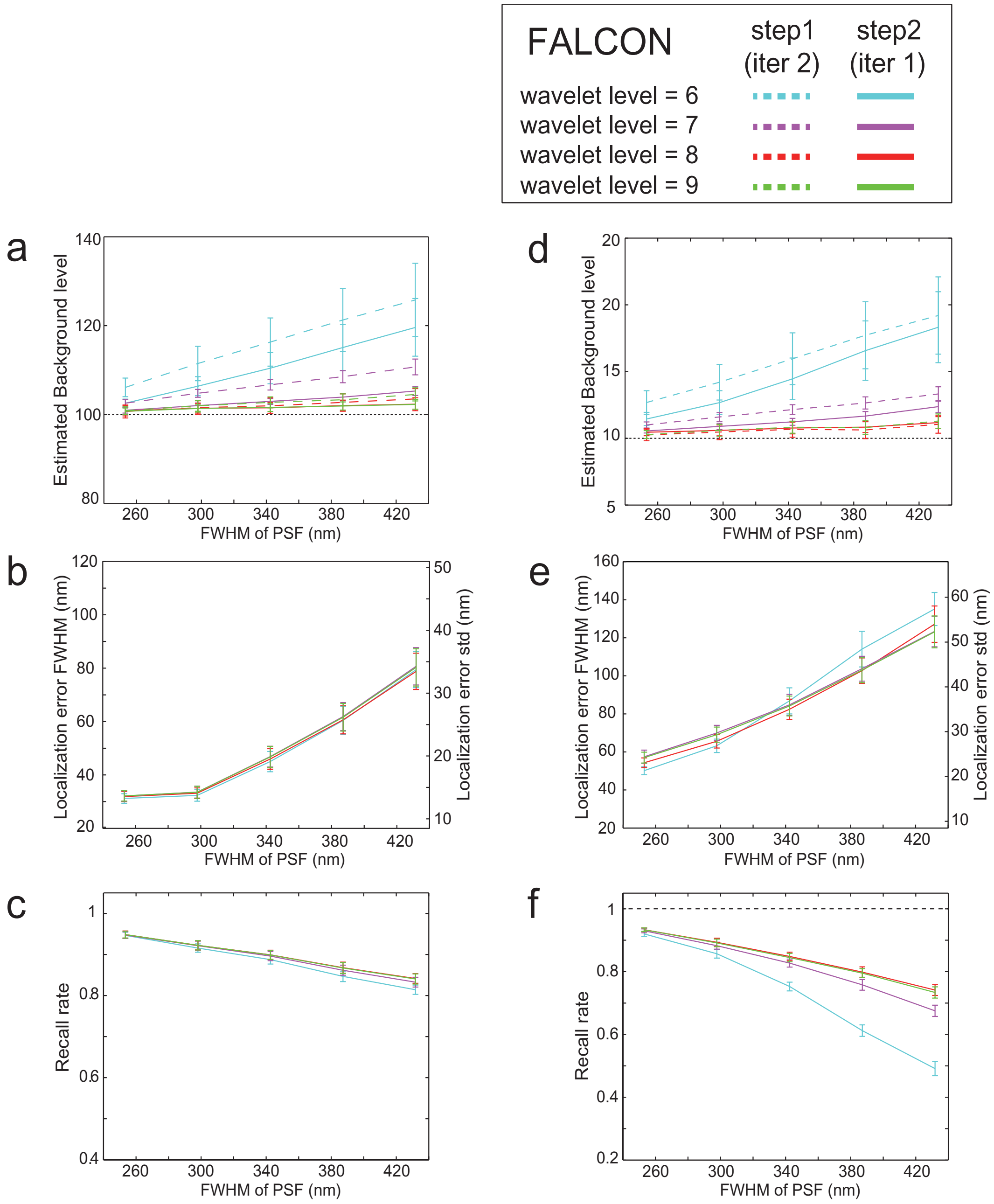

Figure 7 Parameter choice analysis for background estimation. Two iterations are used in the step 1 and one iteration is used in the step 2 at every update of the background. Background is estimated by only using coefficients of the highest wavelet level which varies from 6 to 9. (a) Estimated background level (true: 100) (b) Localization accuracy, (c) Molecular recall rates with various wavelet levels for high SNR simulated data, (d) Estimated background level (true: 10) (e) Localization accuracy, (f) Molecular recall rates with various wavelet levels for low SNR simulated

data. The simulated image was generated by the following settings: 100 uniform background photons and 5000 photons per molecule (with 2000 s.d.) on average for high SNR, and 10 uniform background photons and 350 photons per molecule (with 70 s.d.) on average for low SNR. The error bars indicate standard deviations.

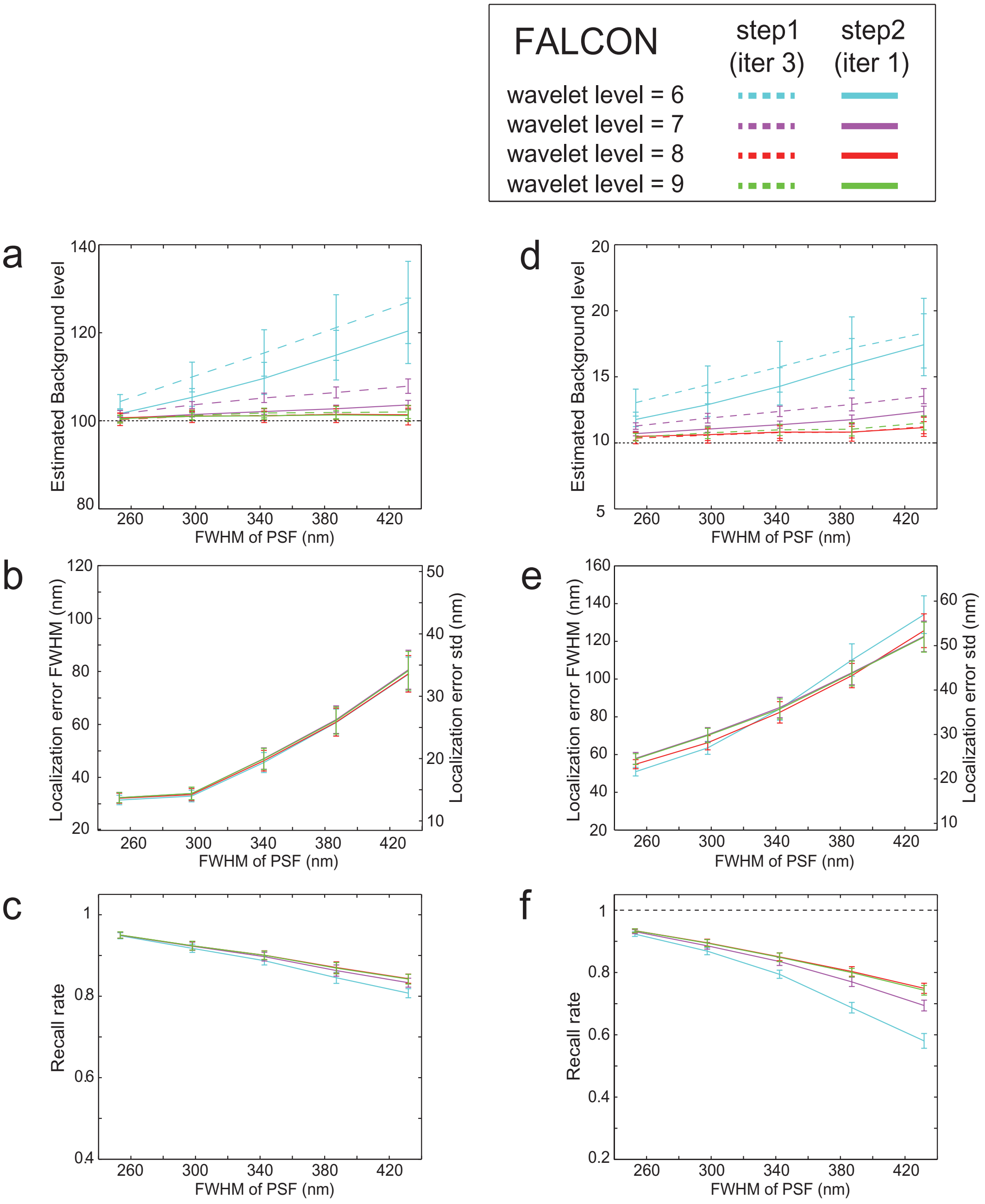

Figure 8 Parameter choice analysis for background estimation. Three iterations are used in the step 1 and one iteration is used in the step 2 at every update of the background. Background is estimated by only using coefficients of the highest wavelet level which varies from 6 to 9. (a) Estimated background level (true: 100) (b) Localization accuracy, (c) Molecular recall rates with various wavelet levels for high SNR simulated data, (d) Estimated background level (true: 10) (e) Localization accuracy, (f) Molecular recall rates with various wavelet levels for low SNR simulated

data. The simulated image was generated by the following settings: 100 uniform background photons and 5000 photons per molecule (with 2000 s.d.) on average for high SNR, and 10 uniform background photons and 350 photons per molecule (with 70 s.d.) on average for low SNR. The error bars indicate standard deviations.

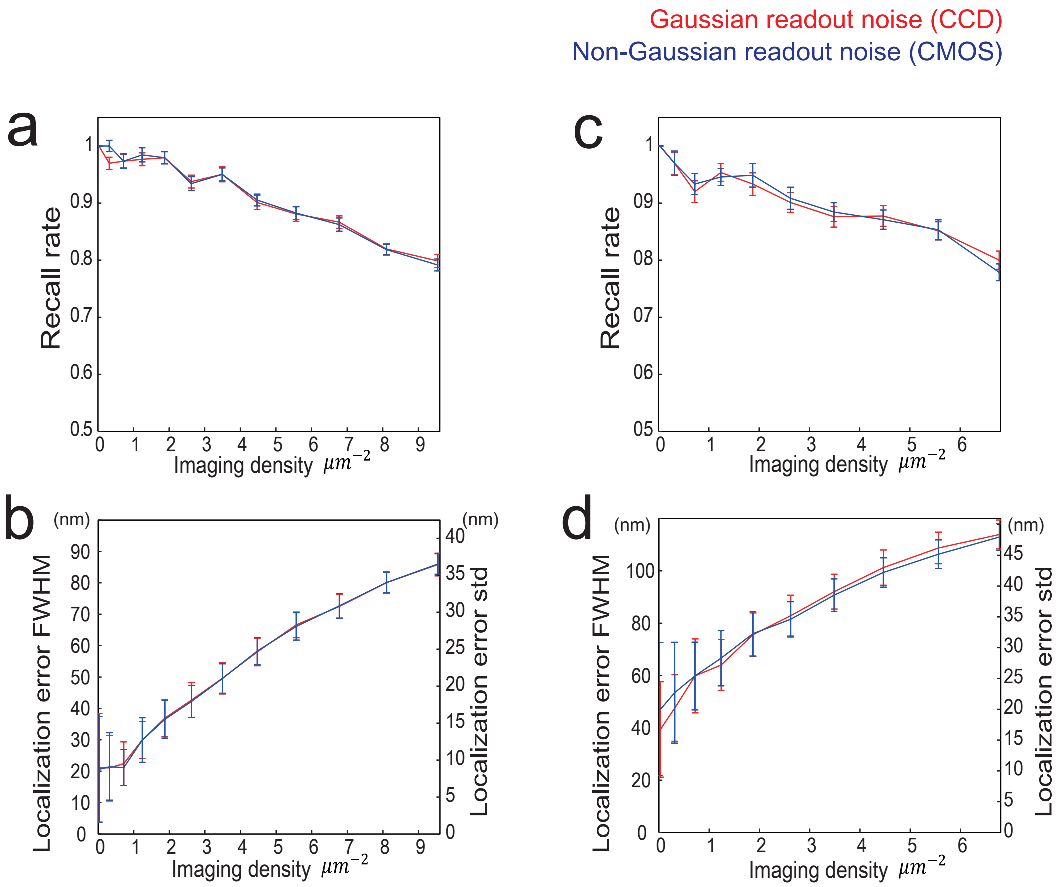

Figure 9 Performance analysis of FALCON under Gaussian(CCD) and non-Gaussian(CMOS) readout noise statistics. In order to approximate readout noise of CCD or CMOS type camera, simulated images are generated with Gaussian (zero mean with 2 s.d.) or log-normal distribution (unit mean with 2 s.d.). The simulated images are reconstructed by FALCON. Simulation results on the random distribution of molecules with high-photon emission rates: recall rates (a), localization accuracy (b) and low-photon emission rates: recall rates (c), localization accuracy (d). Simulation results under Gaussian and non-Gaussian readout noise are denoted by red and blue color respectively. The error bars indicate standard deviations.

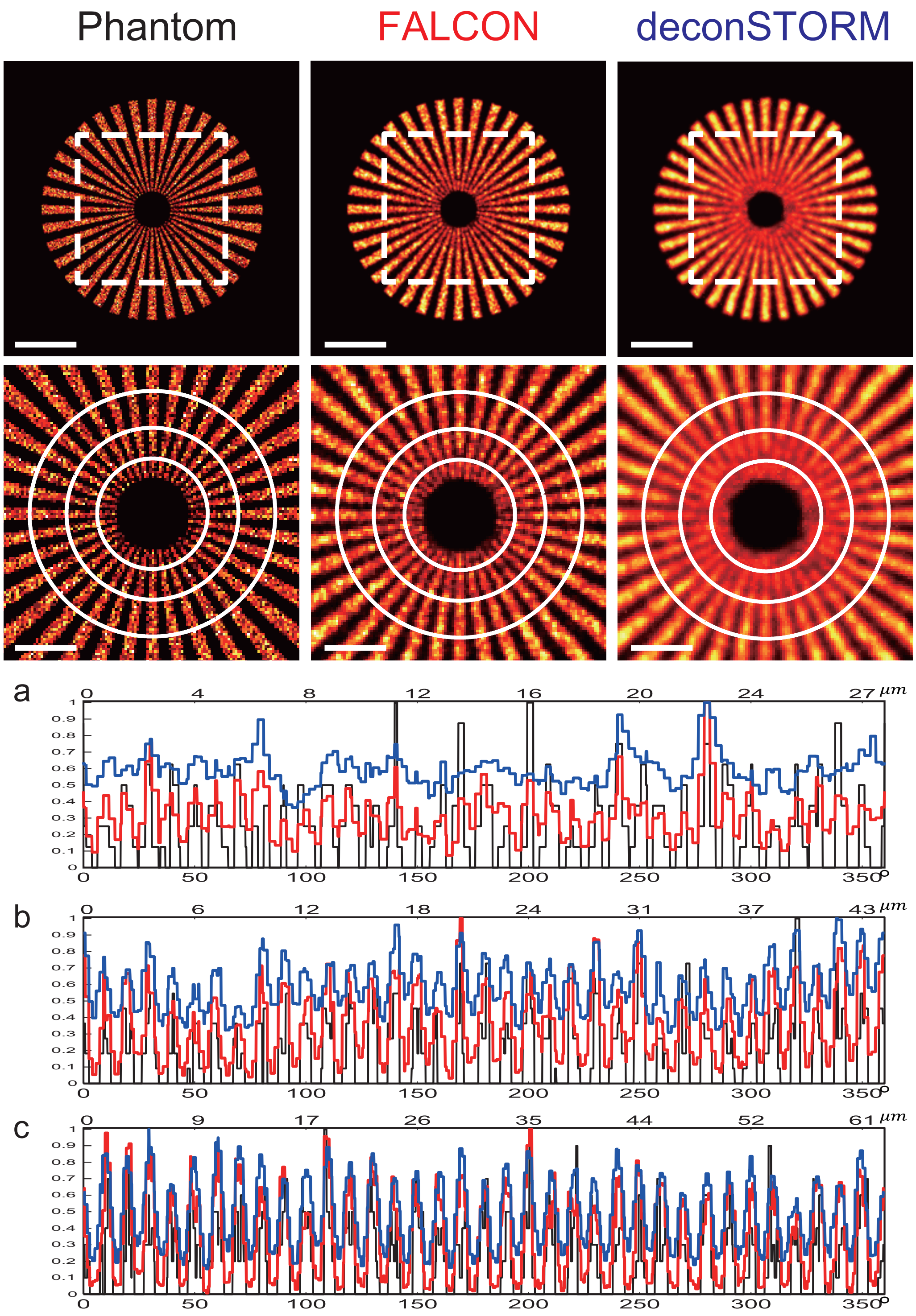

Figure 10 Performance analysis in comparison with deconSTORM in effective molecular density (on average, number of activated molecules/area of phantom) 8.6 μm^{-2}. Each activated molecule is shown in two consecutive frames on average. The 2000 simulated images were generated by the

following photon-emission settings: on average, 5000 photons per molecule (with 2000 s.d.) and 100 uniform background photons per camera pixel. The simulated phantom image and reconstructed images by FALCON and deconSTORM are shown in top row, and their close-up images in central region denoted by white box are presented in second row. For deconSTORM, required parameters of the algorithm such as average molecular activation time and background level set equal to simulation settings. Line profiles on circles from the inner to outer (a-c). profiles of the simulated phantom, FALCON and deconSTORM are denoted by black, red, blue colors respectively. Scale bars are $1\ \mu m$ in top row and 500 nm in close-up images.

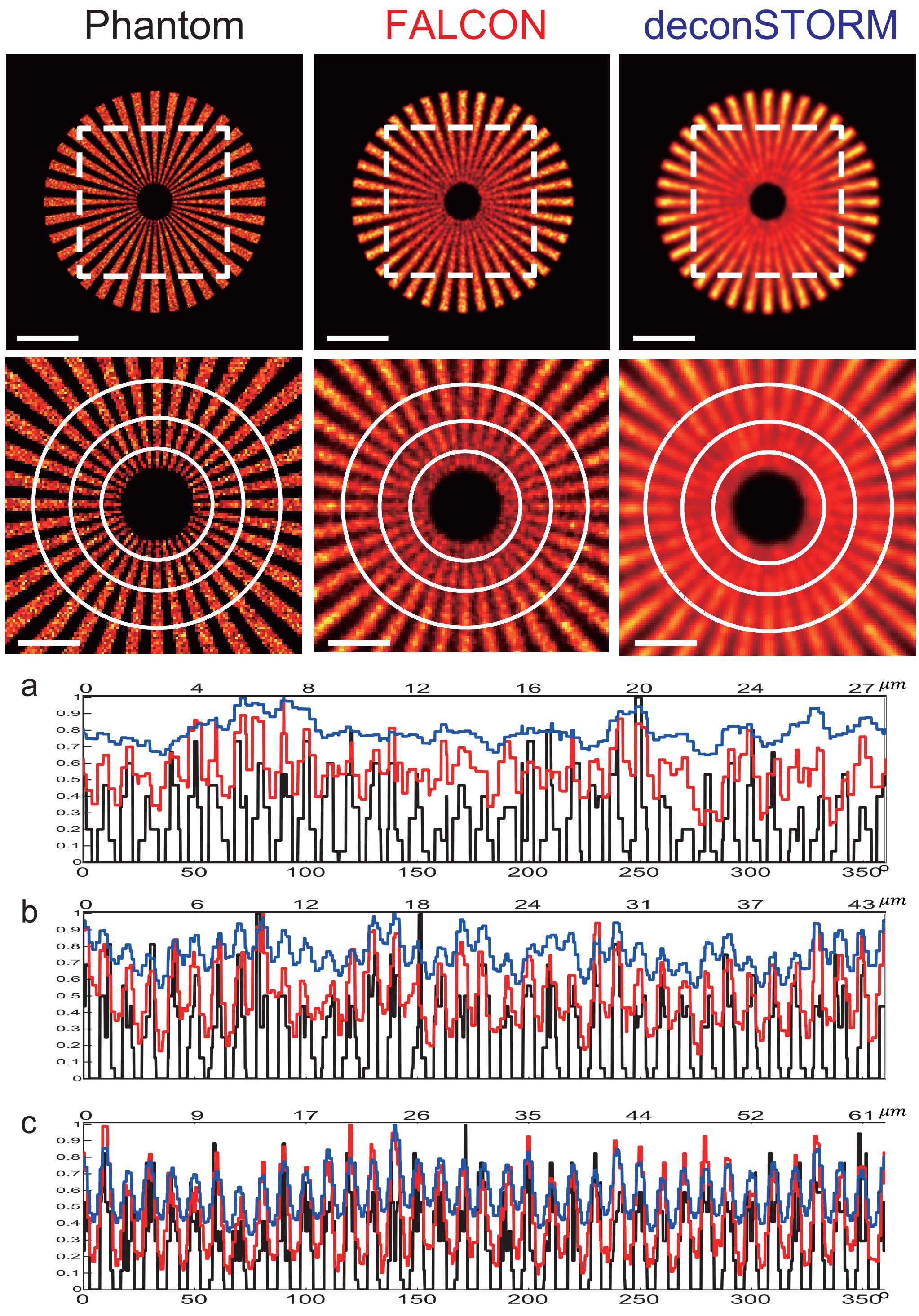

Figure 11 Performance analysis in comparison with deconSTORM in effective molecular density (on average, number of activated molecules/area of phantom) 17.2 μm^{-2}. Each activated molecule is shown in two consecutive frames on average. The 2000 simulated images were generated by the

following photon-emission settings: on average, 5000 photons per molecule (with 2000 s.d.) and 100 uniform background photons per camera pixel. The simulated phantom image and reconstructed images by FALCON and deconSTORM are shown in top row, and their close-up images in central region denoted by white box are presented in second row. For deconSTORM, required parameters of the algorithm such as average molecular activation time and background level set equal to simulation settings. Line profiles on circles from the inner to outer (a-c). profiles of the simulated phantom, FALCON and deconSTORM are denoted by black, red, blue colors respectively. Scale bars are $1\ \mu m$ in top row and 500 nm in close-up images.

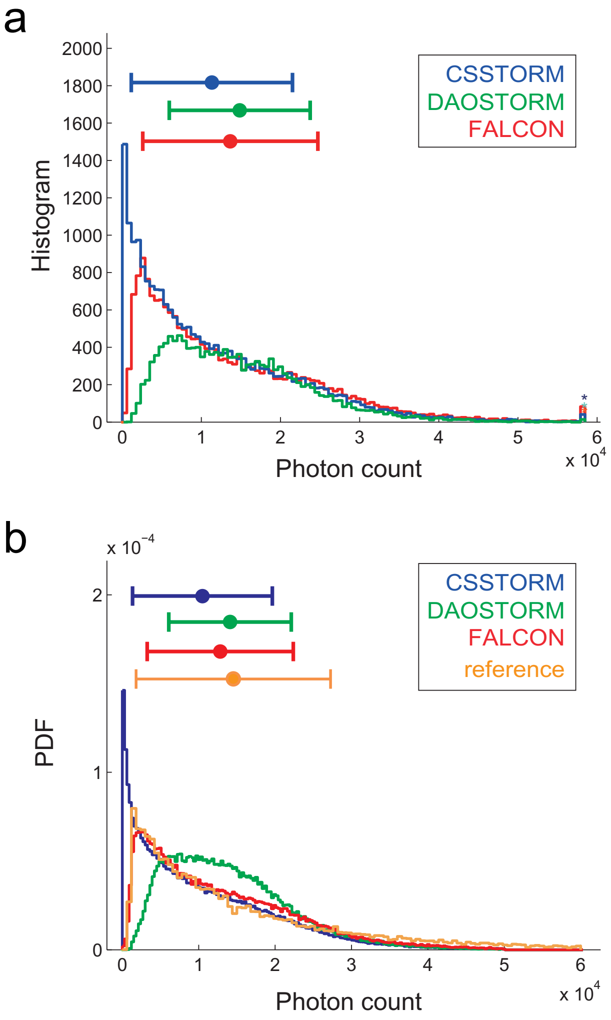

Figure 12 Photon counts analysis on high-density STORM images of microtubules (MTs) in a COS-7 cell. (a) Histograms of the photon-counts measured on the dataset used for figure 7, with the mean value and standard deviation displayed on top, for CSSTORM (blue), DAOSTORM (green) and FALCON (red). The three algorithms show significant differences for the range of low photon counts ($<$ 5,000). (b) Probability distributions of (a) in comparison with a reference distribution obtained from the low-density data under the identical preparation protocols [8]. FALCON shows similar distribution with that of the reference measurement, while DAOSTORM misses a lot of probes having low photon counts and CSSTORM overcounts probes with low photon counts ($<$ 500). These are likely false positive localizations. The error bars indicate standard deviations.

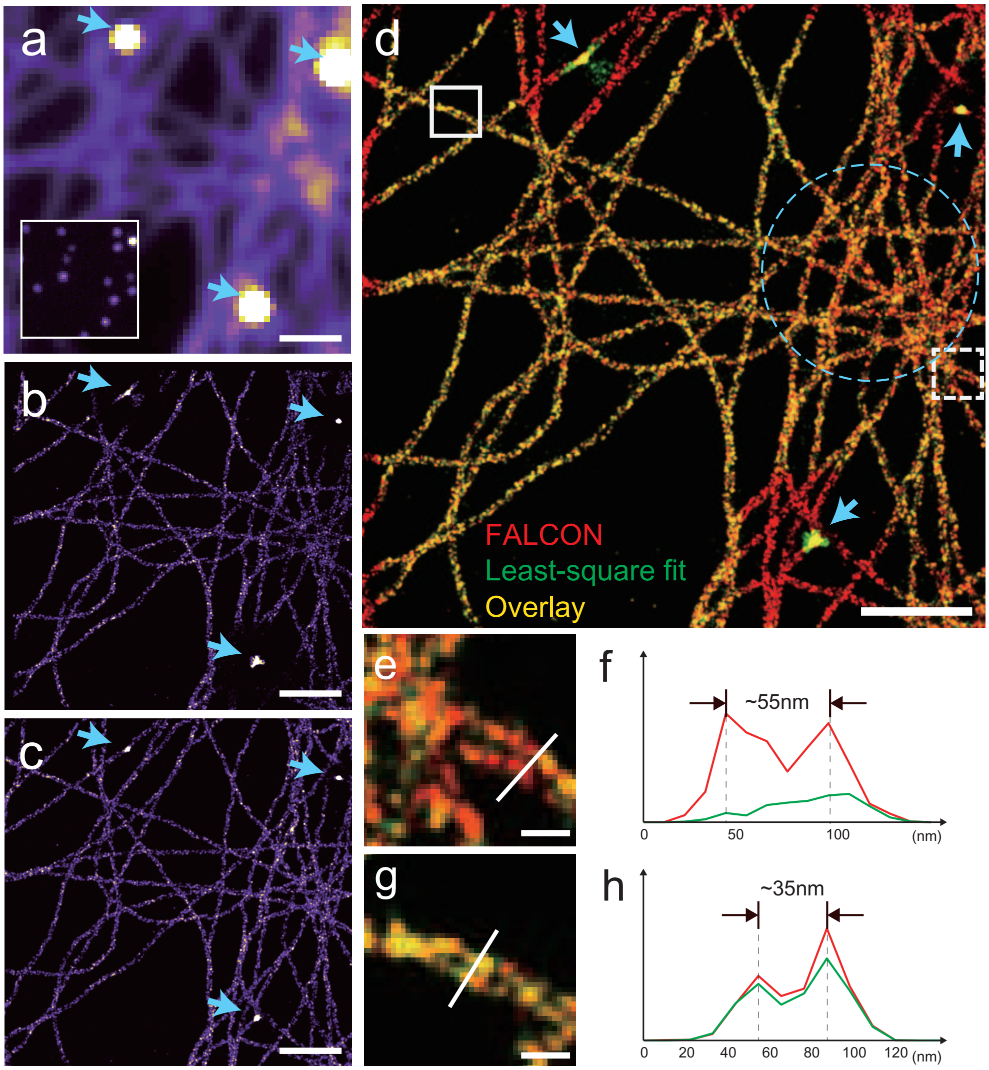

Figure 13 Low-density STORM images of microtubules (MTs) in a COS-7 cell. STORM dataset of 64 x 64 pixels x 10,000 frames recorded from microtubules (α-tubulin) stained with Alexa-647, and gold fiducial beads highlighted with blue arrows for de-drifting. (a) Wide-field image generated by summing all the frames. (b) STORM image reconstructed by Least-square (LS) fitting. (c) STORM image reconstructed by FALCON . (d) Overlay between the two images: LS fitting is missing molecules in the neighbouring areas of fiducial beads because the molecules too close to the fiducial beads are rejected, while the information is reserved by FALCON (red area around the blue arrows). Moreover, in the densest area of MTs without a fiducial bead denoted dotted blue circle, localization density by LS fitting is noticeably reduced due to the higher probability of PSF. However,

FALCON well resolved complex structures of MTs and has a more clear line profile (red curve) of two neighbouring MTs as can be seen in (e-f). We also took a line profile through MTs in lower density area where the LS fitting algorithm should perform optimally, and could resolve in both cases the hollowness of the structure, with two peaks distant from 35 nm consistent with a raw diameter of 25 nm augmented by the presence of the antibodies as can be seen in (h) where the green curve corresponds to LS fitting and the red curve to FALCON. Scale-bars are 1 μm in (a-d) and 100 nm in (e,g), respectively.

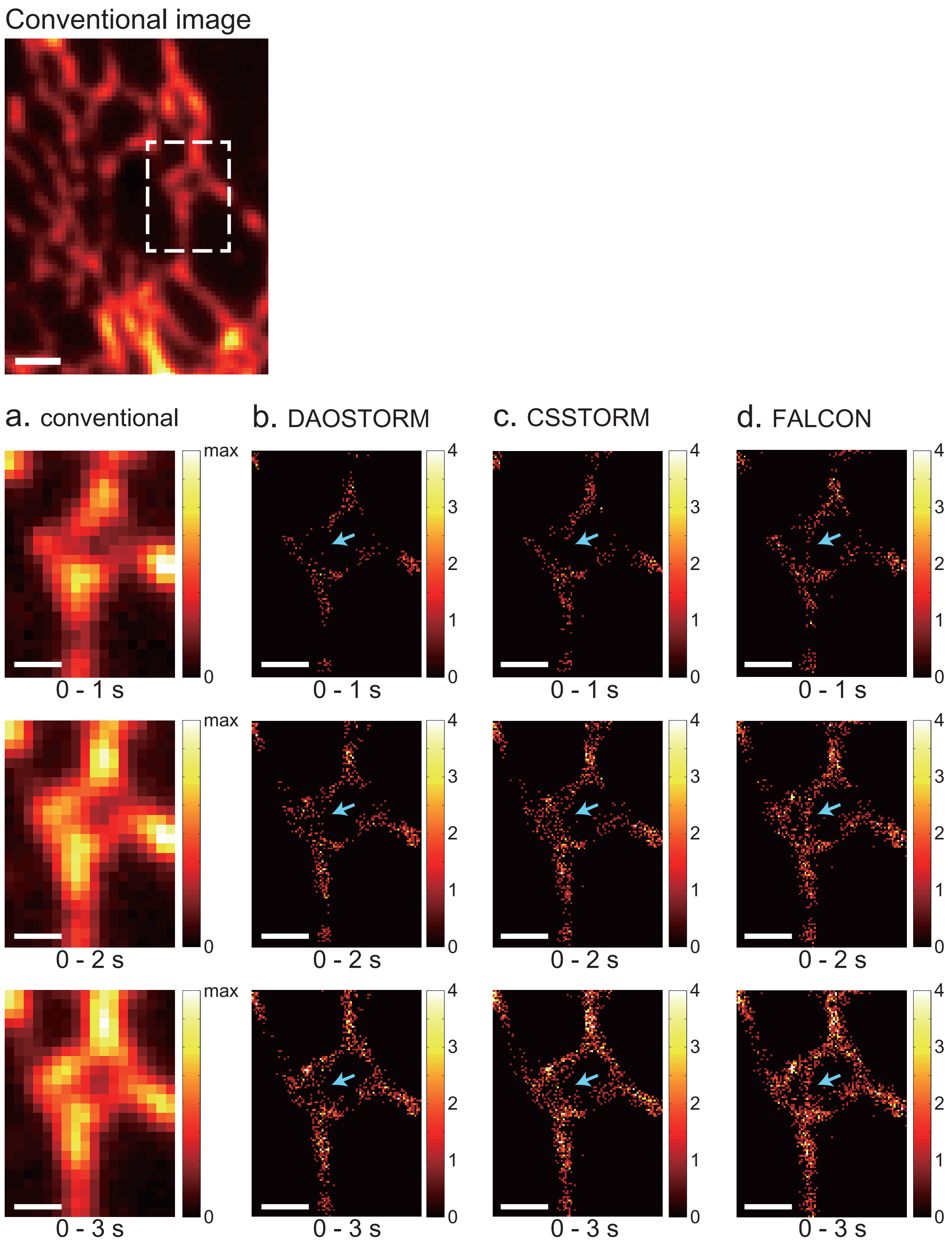

Figure 14 Super-resolution images reconstructed by DAOSTORM, CSSTORM and FALCON with live-imaging data of ER in U2OS cell. (a) conventional images and (b-d) reconstructed images by

DAOSTORM, CSSTORM and FALCON respectively with different time scales $(1s-3s)$ from the region of dotted white box in the large conventional image on top. The reconstructed images generated as 2D histograms of 20 nm bins in (b-d). In comparison with DAOSTORM and CSSTORM, FALCON detected more molecules up to 50$\%$, which can improve temporal resolution. The tubular structures of ER only well preserved by FALCON are highlighted by cyanine arrows. Imaging of ER was performed at a frame rate of 64 fps. Scale-bars are 1 μm in the large conventional image and 500 nm in (a-d).