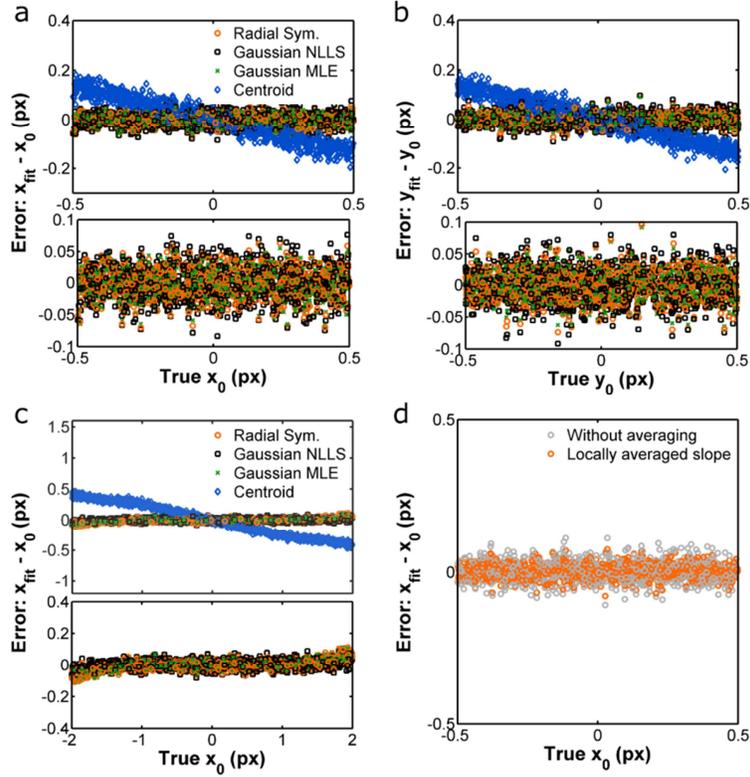


Rapid, accurate particle tracking by calculation of radial symmetry centers

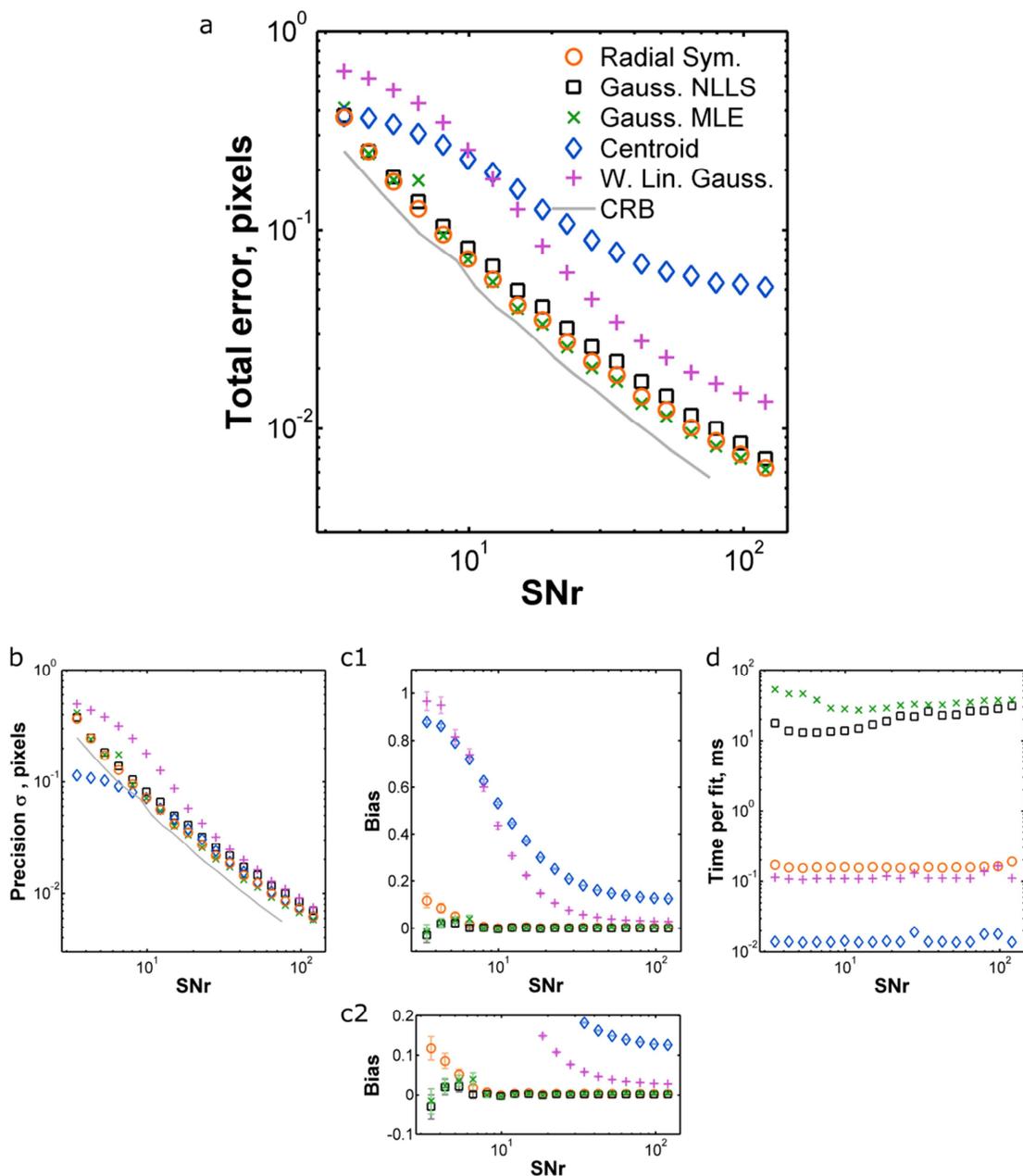
Raghuveer Parthasarathy

Supplementary Text and Figures

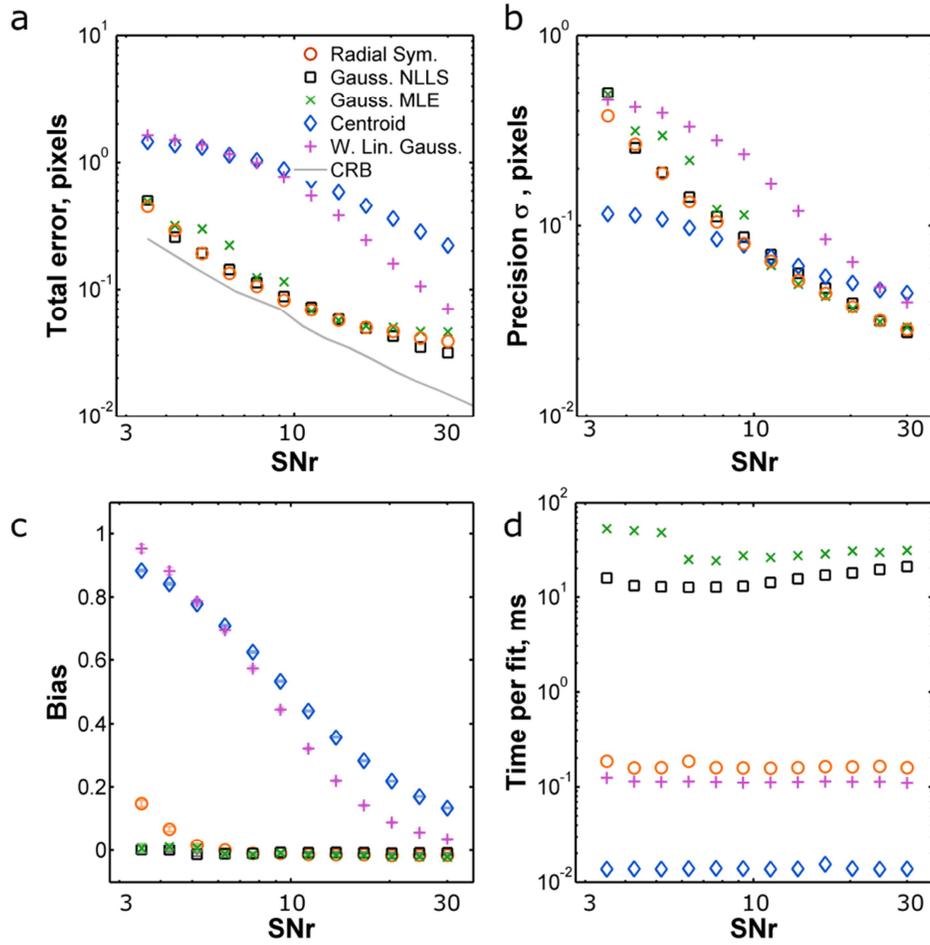
Supplementary Figures	
Supplementary Figure 1	Tracking accuracy for various particle localization algorithms applied to simulated particle images with a signal-to-noise ratio of 20.
Supplementary Figure 2	Localization accuracy over a range of SNr for radial-symmetry-based tracking and other methods
Supplementary Figure 3	Localization accuracy over a range of SNr, with true particle centers distributed over ± 2.0 pixels in x and y .
Supplementary Figure 4	Localization accuracy in the presence of a second particle.
Supplementary Figure 5	Localization accuracy for simulated images of concentric rings.
Supplementary Figure 6	More experimental tests of radial-symmetry-based particle localization.
Supplementary Figure 7	The relationship between total number of photons detected and signal-to-noise ratio (SNr).
Supplementary Note	Particle Localization Algorithm
Supplementary Results	Additional characterizations of particle localization accuracy
References	References for Supplementary Notes and Results



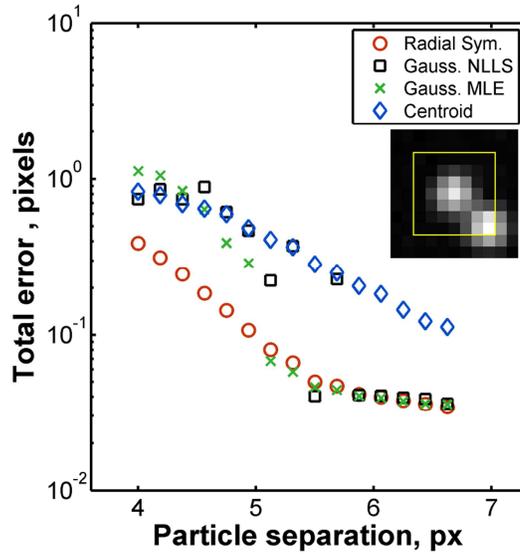
Supplementary Figure 1. Tracking accuracy for various particle localization algorithms applied to simulated particle images with a signal-to-noise ratio of 20. “Radial Sym.” indicates the radial-symmetry-based approach introduced here; “Centroid” indicates centroid finding; “Gaussian NLLS” and “Gaussian MLE” refer to Gaussian fitting performed using nonlinear least squares minimization and maximum likelihood estimation, respectively. (a,b) The total error, defined as the difference between the fit-determined and true particle centers, plotted as a function of the true center location. The true particle centers are randomly distributed over ± 0.5 pixels in x and y . The error in the x and y dimensions are plotted separately. The lower panels show the same points as the upper panels but with the centroid-fit values omitted. (c) The total error in x for true particle centers spanning ± 2.0 pixels. The mean errors for the radial symmetry, Gaussian NLLS, and Gaussian MLE methods are very similar to their values over the smaller range (noted in the main text): 0.045, 0.042, and 0.048 px, respectively. (d) The total error in x from radial symmetry based tracking as calculated with and without local smoothing of slopes.



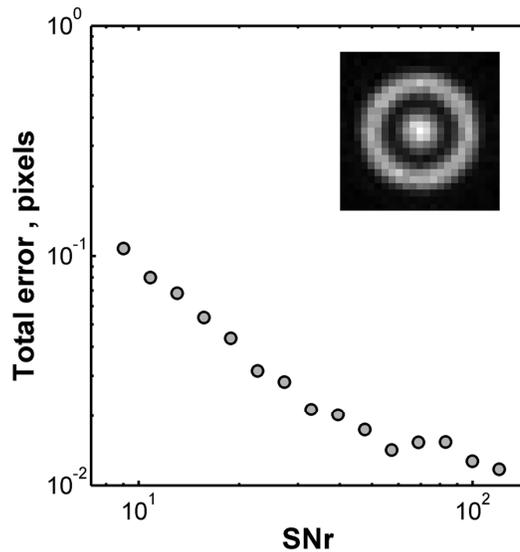
Supplementary Figure 2. Localization accuracy. (a) Total error, (b) precision, (c) bias, and (d) computation time from tracking simulated particle images over a range of SNr using radial-symmetry-based tracking, Gaussian NLLS and MLE fitting, centroid determination, and linearized weighted Gaussian fitting. (See Online Methods for precise definitions of terms.) The examined range of signal-to-noise ratios corresponds to a total number of detected photons from 100 to 300,000 (see Supplementary Results Figure 2). Each point denotes the average of 1,000 tests at each SNr. The true particle centers are randomly distributed over ± 0.5 pixels in x and y . The solid lines in (a) and (b) indicate the Cramér-Rao bound on localization precision. Panel (a) is the same as Figure 2b; the larger size facilitates discrimination between the various symbols. Panel (c2) magnifies Panel (c1).



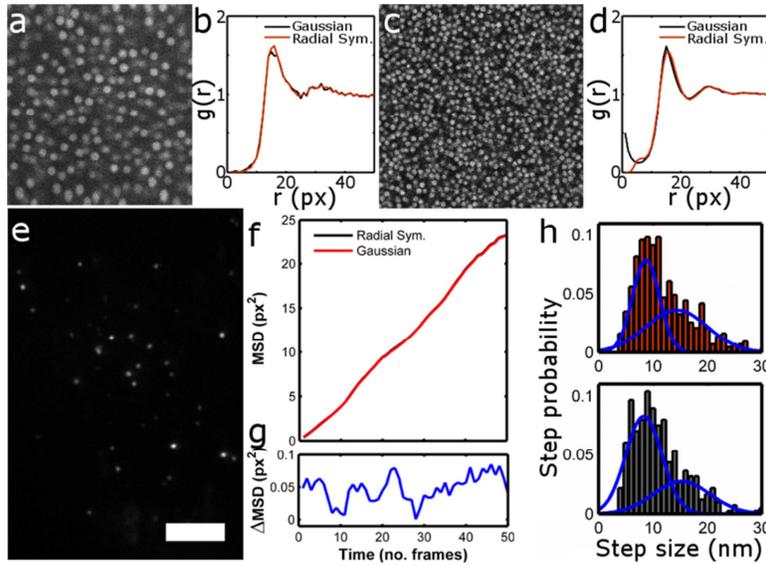
Supplementary Figure 3. Localization accuracy. (a) Total error, (b) precision, (c) bias, and (d) computation time as in **Supplementary Fig. 2**, but with true particle centers distributed over ± 2.0 pixels in x and y rather than 0.5 pixels. The solid line in (a) indicates the Cramér-Rao bound on localization precision.



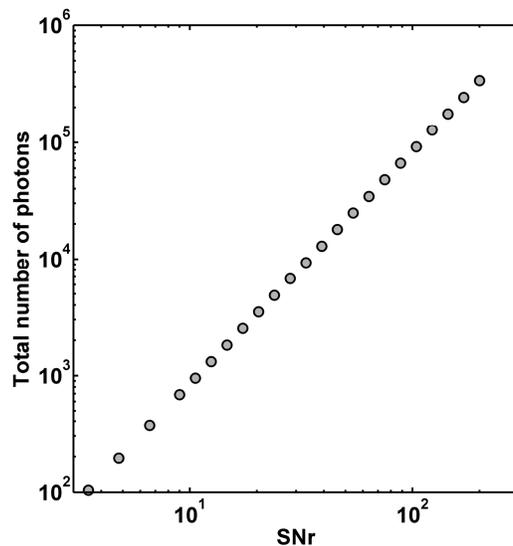
Supplementary Figure 4. Localization accuracy in the presence of a second particle. Model images were constructed at SNr=20, consisting of two particles at a given center-to-center separation and a random angular orientation. Each point denotes the average, over 400 images, of error in the position determined for the central particle. Inset: A model image with two particles separated by 4 px and oriented at -45 degrees with respect to the x -axis. The yellow box indicates the 7x7 pixel region analyzed to localize the central particle.



Supplementary Figure 5. The accuracy of center determination for simulated images of concentric rings over a range of signal-to-noise ratios. Each point denotes the average of 200 simulated images for SNr<20 and 50 images for SNr>20. Inset: A simulated image with SNr=20.



Supplementary Figure 6. More experimental tests of radial-symmetry-based particle localization. (a) Confocal micrograph of fluorescent 2.2 μm diameter PMMA microspheres; (b) the 2D pair correlation function $g(r)$. (c) Confocal micrograph of fluorescent 1.1 μm diameter PMMA microspheres; (d) the corresponding $g(r)$. (e) Fluorescence image of quantum dot labeled acetylcholine receptors in a HEK 293 cell. Bar = 5 microns. (f) Mean-square-displacement for a single quantum dot trajectory. The traces derived from radial-symmetry-based and Gaussian tracking are visually indistinguishable; their difference is shown in (g). (h) Histograms of step sizes derived from images of quantum-dot labeled kinesins tracked with radial-symmetry-based (upper) and Gaussian fitting (lower) methods. Solid lines: two-Gaussian fits to the histograms.



Supplementary Figure 7. The total number of photons detected in the CCD plane as a function of the peak SNr for the simulated images used in this paper. The number of photons detected at the brightest pixel is SNr^2 ; the plotted total sums over all pixels.

Supplementary Note: Particle Localization Algorithm

My approach to particle localization is to determine the center of radial symmetry of an intensity distribution sampled at discrete points, $\{I_{ij}(x_{ij}, y_{ij})\}$, where I denotes the intensity and the subscripts label rows and columns on a two-dimensional lattice with coordinates x and y . Typically, these coordinates will be those of pixels in a CCD array, and the set of intensities will correspond to a single particle imaged by a noisy optical device. Radial symmetry is not in itself a number whose value can be computed and optimized, and so we need a measure of symmetry that is calculable for a discretely sampled function. Many possibilities exist; a fruitful approach is to note that the gradient of a function that has perfect radially symmetry about the coordinate origin will, at any point, “point” toward the origin. A line through any point with orientation parallel to the local gradient will intersect the origin. (This basic fact has been observed before, e.g. in Ref. 12 in the context of concentric rings.) It is fruitful to think of the origin as the point that minimizes the distance between itself and all such lines, that distance being zero. This perspective, I propose, can be extended to noisy, imperfectly symmetric functions. We can identify the point of minimal distance to gradient-oriented lines as an estimate of the “true” particle center, and test the accuracy of this proposition by processing simulated images with known center positions.

Given the intensities at each pixel position, $\{I_{ij}(x_{ij}, y_{ij})\}$, we calculate the intensity gradient on a grid displaced half a pixel in each dimension using a Roberts cross operator^{S1} (**Fig. S1**). For any set of four pixels of indices $\{(i, j), (i+1, j), (i, j+1), (i+1, j+1)\}$ (Figure 2a), the gradient at the midpoint (x_k, y_k) is $\vec{\nabla}I_k = (I_{i+1, j+1} - I_{i, j})\hat{u} + (I_{i, j+1} - I_{i+1, j})\hat{v}$, where the w -axes are rotated 45° from xy (Figures 1c, 2a). The Roberts cross allows assignment of both components of the gradient at the same point, in contrast, for example, to calculating x -components from differences along columns $(I_{i+1, j} - I_{i, j})$ and y -components along rows $(I_{i, j+1} - I_{i, j})$. The slope of the gradient at the midpoint in the xy coordinate system, rotated 45 degrees with respect to w , is given by

$$m_k = \frac{(I_{i+1, j+1} - I_{i, j}) + (I_{i, j+1} - I_{i+1, j})}{(I_{i+1, j+1} - I_{i, j}) - (I_{i, j+1} - I_{i+1, j})};$$

the terms in parentheses are the u and v components of $\vec{\nabla}I_k$ noted above. For any of our grid midpoints, located at (x_k, y_k) , we can now write the equation of a line of slope m_k passing through (x_k, y_k) : $y = y_k + m_k(x - x_k)$. Such a line is illustrated in **Fig. S1b**.

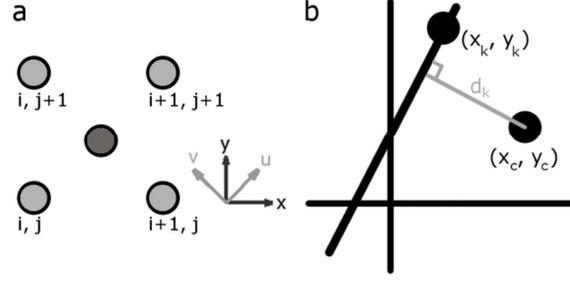


Figure S1. Pixel and gradient geometry. (a) The gradient of the image intensity is determined on a grid (one site of which is indicated in dark gray) displaced half a pixel in x and y from the grid of pixel centers (light gray). (b) At each point at which the gradient is calculated (e.g. (x_k, y_k)), we can consider a line parallel to gradient, passing through that point. We wish to locate the point (x_c, y_c) that minimizes the weighted sum of the squared distances of closest approach, d_k , to all such lines.

We next consider some arbitrary point (x_c, y_c) . It is a straightforward exercise in geometry to show that the distance, d_k , between (x_c, y_c) and the line $y = y_k + m_k(x - x_k)$ is given by

$$d_k^2 = \frac{[(y_k - y_c) - m_k(x_k - x_c)]^2}{m_k^2 + 1}$$

Considering all lattice midpoints, labeled k , we wish to determine the “best fit” point (x_c, y_c) that minimizes $\chi^2 \equiv \sum_k d_k^2 w_k$, where w_k is some weighting of point k , discussed below. Setting the derivative of χ^2 with respect to x_c equal to zero; after some algebra one obtains:

$$x_c \sum_k \frac{-m_k^2 w_k}{m_k^2 + 1} + y_c \sum_k \frac{m_k w_k}{m_k^2 + 1} = \sum_k \frac{m_k w_k (y_k - m_k x_k)}{m_k^2 + 1}$$

Similarly, for y_c :

$$x_c \sum_k \frac{-m_k w_k}{m_k^2 + 1} + y_c \sum_k \frac{w_k}{m_k^2 + 1} = \sum_k \frac{w_k (y_k - m_k x_k)}{m_k^2 + 1}$$

This yields expressions for x_c and y_c that are simple combinations of the above sums:

$$(1.1) \quad x_c = \Delta^{-1} \left[\left(\sum_k \frac{m_k w_k (y_k - m_k x_k)}{m_k^2 + 1} \right) \left(\sum_k \frac{w_k}{m_k^2 + 1} \right) - \left(\sum_k \frac{m_k w_k}{m_k^2 + 1} \right) \left(\sum_k \frac{(y_k - m_k x_k) w_k}{m_k^2 + 1} \right) \right]$$

$$(1.2) \quad y_c = \Delta^{-1} \left[\left(\sum_k \frac{m_k w_k (y_k - m_k x_k)}{m_k^2 + 1} \right) \left(\sum_k \frac{m_k w_k}{m_k^2 + 1} \right) - \left(\sum_k \frac{m_k^2 w_k}{m_k^2 + 1} \right) \left(\sum_k \frac{(y_k - m_k x_k) w_k}{m_k^2 + 1} \right) \right]$$

where

$$(1.3) \quad \Delta = \left(\sum_k \frac{m_k w_k}{m_k^2 + 1} \right)^2 - \left(\sum_k \frac{m_k^2 w_k}{m_k^2 + 1} \right) \left(\sum_k \frac{w_k}{m_k^2 + 1} \right)$$

Weighting

It is reasonable to apply different weightings to the distances calculated from different grid points. The intensity gradient is more accurately determined in areas of high intensity, as the image signal is larger compared to the noise. The distance to (x_c, y_c) is more accurately determined for points that are close to (x_c, y_c) , as errors in the slope m_k lead to errors in d_k that grow with the distance $d_{k,c}$ between the point (x_k, y_k) and the center. A simple choice of weighting that incorporates both of these ideas is

$$(1.4) \quad w_k = \frac{|\bar{\nabla} I_k|^2}{d_{k,c}}$$

Use of the intensity gradient rather than the intensity avoids the influence of constant background intensities on the weighting. The distance $d_{k,c}$ is of course unknown prior to calculation of (x_c, y_c) . It can, however, be approximated by the distance d_c between (x_k, y_k) and the centroid of $|\bar{\nabla} I_k|$; this approach is used throughout this paper.

There are many possible weighting functions, some of which may confer greater accuracy than the form written above. The results are not likely to be strongly dependent on the choice of denominator in the weighting function, however. For example, even the very naive approach of using the distance to the brightest pixel, rather than d_c , yields a roughly 1% change in mean accuracy, corresponding to less than 0.001 pixels at a signal-to-noise ratio of 20. Weighting by the square of the gradient magnitude and the inverse distance to a centroid has the virtue of conceptual simplicity and, as shown in the main text and in the **Supplemental Results**, it allows accurate determination of (x_c, y_c) . One could calculate (x_c, y_c) and re-calculate the weights to input into a revised determination of the center position; this method yields a negligible improvement in accuracy (not shown) at the expense of one computational repetition.

Smoothing

A small but significant improvement in accuracy results from “smoothing” the array of slopes, $\{m_k\}$. Again, many approaches to smoothing are available; a simple one is to replace each m_k with the average slope of the 3x3 pixel neighborhood centered on site k . I illustrate the accuracy of image center determination with and without this averaging below (Section 2.1).

Software

Computer code implementing the above algorithm for particle center localization, written in MATLAB, is provided as **Supplementary Software**, together with additional code.

Summary

Equations 1.1-1.4 provide analytically calculable expressions for the location of the center of an imaged radially symmetric particle.

Supplementary Results:

Additional characterizations of particle localization accuracy

Contents

1	Total error, bias, and precision	
2	The FluoroBancroft algorithm	
3	Localization accuracy for asymmetric images	
4	Tests using experimental data	

1 Total error, bias, and precision

I continue here the discussion of accuracy illustrated by localization tests on images with SNr=20, as in **Fig. 2a**. The total errors plotted in **Fig. 2a** have contributions from the methods' bias, i.e. the slope of the error graphs shown, and their precision, i.e. the scatter about this slope (see Online Methods for formal definitions of these properties). Centroid finding shows high precision together with a large, well-known bias that reports calculated center positions as being closer to the image center than the true particle center location (**Supplementary Fig. 1a,b**). Gaussian fitting and radial-symmetry-based tracking show small biases.

Smoothing

As described in the **Supplementary Note**, I apply a smoothing of the gradient slope prior to the determination of the symmetry center. Without smoothing of the local slopes, the total error of the radial symmetry method is slightly larger, e.g. 0.044 px compared to 0.027 px for the data at SNr=20 (**Supplementary Fig. 1d**), and the execution time is about 20% shorter. This faster performance is inconsequential given the intrinsically very high speed of radial symmetry based tracking, so all tests and implementation in this paper (except that reported in **Supplementary Fig. 1d**) employ the smoothing of m over a 3px x 3px neighborhood. It is reasonable that this local averaging of gradient slopes mitigates the effect of noise in the determination of the best-fit center location; I cannot, however, provide a mathematically rigorous justification of the effect of smoothing on the algorithm's performance.

Coordinate axis bias

As noted in the main text, the errors plotted in **Fig. 2a** and **Supplementary Fig. 1a,b**, and **d** are derived from simulated images with true centers spanning $x_0, y_0 \in [-0.5, 0.5]$ px, distributed throughout this two-dimensional space. To examine any potential bias towards particular coordinate axes, I plot in **Fig. S2a** the total error for $x_0=y_0$, i.e. true centers distributed along a line 45 degrees tilted from the x or y axes. The mean total errors are 0.028 px, 0.030 px, and 0.026 px for radial symmetry tracking, Gaussian NLLS fitting, and Gaussian MLE fitting, respectively, which are nearly identical to those determined for x_0, y_0 distributed independently throughout $[-0.5, 0.5]$ px. In addition, I plot in **Fig. S2b** the error in y versus the error in x , which does not show any appreciable asymmetry. The two eigenvalues of the covariance matrix of the x and y errors are similar to each other for each of the various tracking methods: 4.6×10^{-4} and 5.0×10^{-4} for the radial symmetry method, 6.4×10^{-4} and 7.1×10^{-4} for Gaussian NLLS, and 4.0×10^{-4} and 4.4×10^{-4} for Gaussian MLE. The Pearson correlation coefficients for the x and y errors are 0.0034, 0.014, and 0.0031 for radial symmetry tracking, Gaussian NLLS fitting, and Gaussian MLE fitting, respectively. There appears to be no discernible bias with respect to coordinate axes.

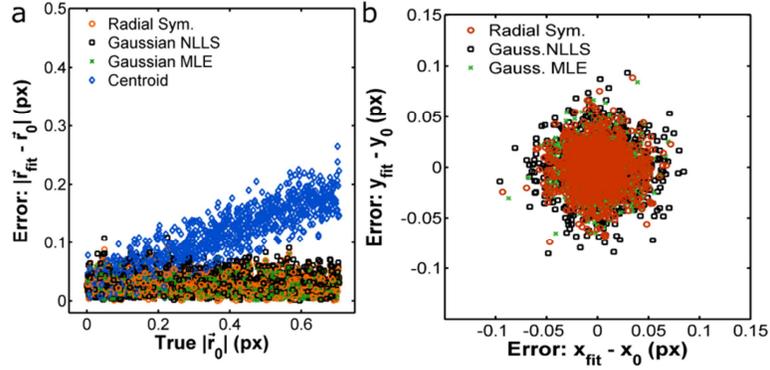


Figure S2. Tracking accuracy for various particle localization algorithms applied to simulated particle images with a signal-to-noise ratio of 20, in which the true centers are distributed along the line $x_0=y_0$. **(a)** Tracking accuracies. The symbol \vec{r} indicates the vector between the calculated or true particle location and the origin. **(b)** The error in the y coordinate of the particle center versus the error in the x coordinate; there is no appreciable asymmetry.

Correlations between methods

The errors resulting from radial symmetry tracking and Gaussian fitting of the same simulated image are strongly correlated (**Fig. S3**), suggesting that both reflect the inherent accuracy with which noisy images can be assessed.

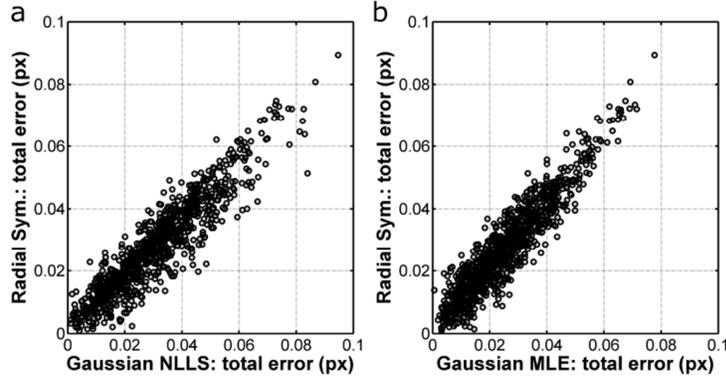


Figure S3. The total error for radial-symmetry-based tracking and Gaussian fitting of the same simulated particles, with Gaussian fitting performed using **(a)** nonlinear least squares minimization and **(b)** maximum likelihood estimation. For the same particle, the error obtained with each of these localization methods are highly correlated.

2 The FluoroBancroft algorithm

I briefly comment on another analytically exact particle tracking algorithm that has been recently proposed to be fast and accurate, the FluoroBancroft algorithm¹⁴. The method requires as a parameter the standard deviation of the presumed Gaussian intensity distribution, given theoretically as:

$$\sigma_{FB} = \frac{0.6\lambda}{\sqrt{2NA}},$$

where the symbols are as defined in Section 1. We can consider different widths, parameterized as $\sigma = f\sigma_{FB}$, with $f = 1$ giving the theoretical value. I plot in **Fig. S4** the tracking error and bias over a range of SNr for the FluoroBancroft algorithm with various f and for radial-symmetry-based tracking. The FluoroBancroft algorithm gives very large error, arising from a large bias – it is more accurate for particles located near the center of a pixel, and less accurate for particles farther away from the center. This bias was not noted in Refs. 14 and 16, but has been described in more recent work^{S2}. Also, we note that the tracking error is strongly dependent on the value of σ chosen, with the lowest error arising from $f = 1.5$. Since particle image widths are rarely exactly equal to theoretical values, and may vary from setup to setup, the strong dependence of tracking error on the input width parameter is a significant concern. Both the FluoroBancroft and weighted linearized Gaussian methods¹³ use the logarithm of intensity values to deduce properties of a Gaussian intensity function. Logarithmic scaling creates nontrivial transformations of noise and background values, however, easily overweighting noisy or background pixels. In fact, the tracking errors for the FluoroBancroft fits shown below assume zero background; incorporating the actual background level used in creating the simulated images results in greater error.

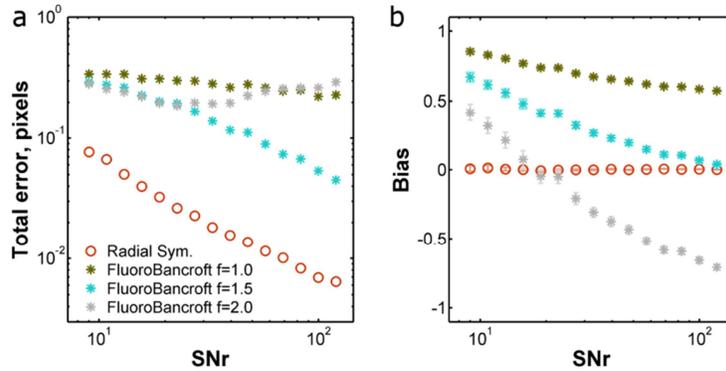


Figure S4. The accuracy of tracking simulated particles using radial-symmetry-based tracking and the FluoroBancroft algorithm for various values of f , including the theoretical value $f=1$. Each point denotes the average error over 1,000 images.

3 Localization accuracy for asymmetric images

One can ask how this radial-symmetry-based tracking and symmetric Gaussian fitting deal in practice with objects whose appearance is unintentionally asymmetric, for example due to optical aberrations. (Of course, for objects that have some known, intrinsic asymmetry, one should fit particle images to an asymmetric model.) There are an infinite number of possible asymmetries, the cataloging of which is beyond the scope of this paper, so as a simple illustration I examine simulated images for which the point spread function is asymmetric, being expanded and contracted by a factor of 1.5 in y and x , respectively. The accuracy of localization of these images is examined over a range of SNr (**Fig. S5**). The radial-symmetry-based tracking and Gaussian fitting methods are similar in their performance with respect to these images, and the overall accuracy is not dramatically worse than for symmetric images.

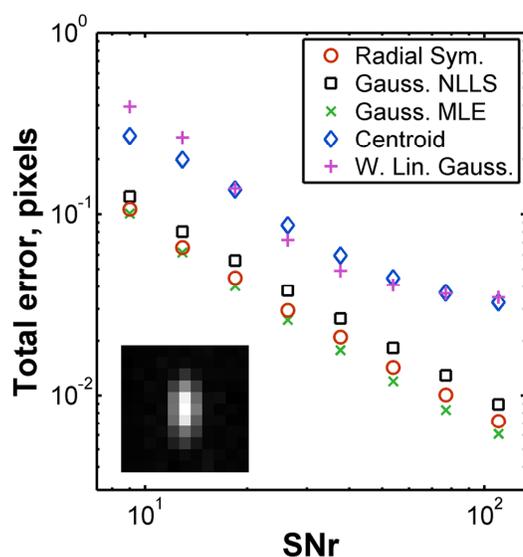


Figure S5. The accuracy of center determination for simulated images of asymmetric spots. Each point denotes the average of 100 simulated images, constructed using an asymmetric point spread function scaled by (1/1.5) and 1.5 in the x and y directions, respectively. Inset: A simulated image with SNr=20.

4 Tests Using Experimental Data

This section continues the discussion of tests of particle localization using experimental data begun in the main text. Sources of experimental data are described in Online Methods.

Imaging the structure and dynamics of colloidal assemblies provides important insights into the material properties of self-assembled structures. Confocal microscopy of fluorescent microparticles has enabled, for example, detailed characterizations of crystalline growth^{S3} and non-equilibrium states^{S4}. **Supplementary Figs. 6a,c** show confocal microscopy slices of three-dimensional colloidal assemblies composed of 2.2 and 1.1 μm diameter poly(methyl methacrylate) (PMMA) and microspheres, respectively (see Online Methods). The two-dimensional pair correlation function $g(r)$ averaged over each slice of the data sets provides a simple derived quantity with which to examine the tracking output. These $g(r)$ are plotted in **Supplementary Figs. 6b,d**, and are nearly identical for Gaussian fitting and radial-symmetry-based tracking. The radial-symmetry-based results may show fewer artifactual signatures of particles separated from other particles by less than one particle diameter (**Supplementary Fig. 6d**).

Tracking the motions of fluorophore-conjugated proteins reveals mechanisms underlying a variety of cellular activities. **Supplementary Fig. 6e** shows a fluorescence image of quantum dot labeled acetylcholine receptors in HEK 293 cells (see Online Methods), together with one example of the mean square displacement of a particle as a function of time (**Supplementary Fig. 6f,g**). Extracting the protein diffusion coefficient from the trajectories of approximately 60 receptors gives a mean value of 0.48 px^2/frame for the radial-symmetry-based tracking and, nearly identically, 0.49 px^2/frame for the Gaussian-fit tracking; the standard deviation across all trajectories is $\pm 0.34 \text{ px}^2/\text{frame}$ for both methods. The median difference between object positions for the two methods is 0.09 px.

Supplementary Fig. 6h shows results derived from fluorescence images of five quantum dot labeled kinesins traveling along microtubules. Prior studies of these motor proteins have revealed discrete 8 nm stepping motions¹. Assessment of particle displacements using a student's t-test⁵⁵ gives the probability distribution of step sizes shown in **Supplementary Fig. 6h**, the first peak of which is centered at 8.7 nm and 8.3 nm for the radial-symmetry-based and Gaussian-fit particle data, respectively.

References for Supplementary Notes and Results

- S1.Solomon, C. & Breckon, T. *Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab*. (John Wiley and Sons: 2011).
- S2.Shen, Z. & Andersson, S. B. Bias and Precision of the fluoroBancroft Algorithm for Single Particle Localization in Fluorescence Microscopy. *IEEE Transactions on Signal Processing* **59**, 4041–4046 (2011).
- S3.Ganapathy, R., Buckley, M. R., Gerbode, S. J. & Cohen, I. Direct measurements of island growth and step-edge barriers in colloidal epitaxy. *Science* **327**, 445–8 (2010).
- S4.Weeks, E. R., Crocker, J. C., Levitt, A. C., Schofield, A. & Weitz, D. A. Three-dimensional direct imaging of structural relaxation near the colloidal glass transition. *Science* **287**, 627–31 (2000).
- S5.Yardimci, H., van Duffelen, M., Mao, Y., Rosenfeld, S. S. & Selvin, P. R. The mitotic kinesin CENP-E is a processive transport motor. *Proc. Natl. Acad. Sci. U.S.A.* **105**, 6016–6021 (2008).