

## An optimized software framework for real-time, high-throughput tracking of spherical beads

J. P. Crossen, D. Dulin, and N. H. Dekker

Citation: [Review of Scientific Instruments](#) **85**, 103712 (2014); doi: 10.1063/1.4898178

View online: <http://dx.doi.org/10.1063/1.4898178>

View Table of Contents: <http://scitation.aip.org/content/aip/journal/rsi/85/10?ver=pdfcov>

Published by the [AIP Publishing](#)

---

### Articles you may be interested in

[A high-throughput cellulase screening system based on droplet microfluidics](#)

*Biomicrofluidics* **8**, 041102 (2014); 10.1063/1.4886771

[RTSPM: Real-time Linux control software for scanning probe microscopy](#)

*Rev. Sci. Instrum.* **84**, 013705 (2013); 10.1063/1.4775717

[Velocity scaling for optimizing replica exchange molecular dynamics](#)

*J. Chem. Phys.* **134**, 044124 (2011); 10.1063/1.3533236

[An open source/real-time atomic force microscope architecture to perform customizable force spectroscopy experiments](#)

*Rev. Sci. Instrum.* **80**, 084301 (2009); 10.1063/1.3194046

[Using open-source software technologies and standardized data structures to build advanced applications for high-throughput experimentation environments](#)

*Rev. Sci. Instrum.* **76**, 062203 (2005); 10.1063/1.1906104

---



**2014 Special Topics**

PEROVSKITES

2D MATERIALS

MESOPOROUS MATERIALS

BIOMATERIALS/ BIOELECTRONICS

METAL-ORGANIC FRAMEWORK MATERIALS

**AIP** | APL Materials

**Submit Today!**

# An optimized software framework for real-time, high-throughput tracking of spherical beads

J. P. Cnossen, D. Dulin, and N. H. Dekker<sup>a)</sup>

*Department of Bionanoscience, Kavli Institute of Nanoscience, Delft University of Technology, Lorentzweg 1, 2628 CJ Delft, The Netherlands*

(Received 27 July 2014; accepted 4 October 2014; published online 22 October 2014)

Numerous biophysical techniques such as magnetic tweezers, flow stretching assays, or tethered particle motion assays rely on the tracking of spherical beads to obtain quantitative information about the individual biomolecules to which these beads are bound. The determination of these beads' coordinates from video-based images typically forms an essential component of these techniques. Recent advances in camera technology permit the simultaneous imaging of many beads, greatly increasing the information that can be captured in a single experiment. However, computational aspects such as frame capture rates or tracking algorithms often limit the rapid determination of such beads' coordinates. Here, we present a scalable and open source software framework to accelerate bead localization calculations based on the CUDA parallel computing framework. Within this framework, we implement the Quadrant Interpolation algorithm in order to accurately and simultaneously track hundreds of beads in real time using consumer hardware. In doing so, we show that the scatter derived from the bead tracking algorithms remains close to the theoretical optimum defined by the Cramer-Rao Lower Bound. We also explore the trade-offs between processing speed, size of the region-of-interests utilized, and tracking bias, highlighting in passing a bias in tracking along the optical axis that has previously gone unreported. To demonstrate the practical application of this software, we demonstrate how its implementation on magnetic tweezers can accurately track (with  $\sim 1$  nm standard deviation) 228 DNA-tethered beads at 58 Hz. These advances will facilitate the development and use of high-throughput single-molecule approaches. © 2014 AIP Publishing LLC. [<http://dx.doi.org/10.1063/1.4898178>]

## INTRODUCTION

Single-molecule tethered particle motion and force spectroscopy techniques are widely used to study in a quantitative fashion an increasingly large number of biological processes such as RNA translation (Vanzi *et al.*, 2003), transcription (Schafer *et al.*, 1991), the behavior of molecular motors (Gelles *et al.*, 1988), protein-induced DNA bending (Tolić-Nørrelykke *et al.*, 2006), or DNA supercoiling (Crut *et al.*, 2007; Strick *et al.*, 1998; van Loenhout *et al.*, 2012a). In almost all cases, such experiments rely on the use of spherical beads, to which biological molecules such as DNA, RNA, or proteins are tethered. The motion of the spherical bead is then used to obtain information about the state of the tethered biological molecule. This is the case for tethered particle motion (Schafer *et al.*, 1991) (Figure 1(a)), flow stretching experiments (Kim *et al.*, 2007; van Oijen *et al.*, 2003) (Figure 1(b)), optical tweezers (either with photodiode- or with video-microscopy based bead tracking (Gibson *et al.*, 2008; Le Gall *et al.*, 2010; van Loenhout *et al.*, 2013) (Figure 1(c)), micro-rheology experiments (Wirtz, 2009), magnetic tweezers (Smith *et al.*, 1992; Strick *et al.*, 1996; Haber and Wirtz, 2000) (Figure 1(d)) and for their variants that make possible the measurement of torque and twist (Gore *et al.*, 2006; Lipfert *et al.*, 2010; Lipfert *et al.*, 2011) (Figures 1(e) and

1(f)). For example, by monitoring the coordinates of a spherical bead in optical or magnetic tweezers, changes in the length of the biological molecule tethered to the bead can be measured, which may report, e.g., on the binding of proteins to nucleic acids such as DNA (van Loenhout *et al.*, 2013; Xiao *et al.*, 2011).

The tracking of spherical beads in such experiments is typically performed using either quadrant photodiode detectors or by CCD-based video microscopy. Photodiode-based detection has the advantage of being fast (tens of kHz), which makes it possible to capture the full spectrum of fluctuations of spherical beads in optical traps using back-focal plane detection (Jannasch *et al.*, 2011; Neuman and Block, 2004). Using this type of approach, it has been possible to develop high-resolution optical trapping instruments capable of monitoring the motion of molecular machines at Ångstrom resolution (Abbondanzieri *et al.*, 2005). However, a quadrant photodiode is essentially a very fast four pixel camera that, in the context of optical trapping, typically monitors the motion of only a single bead. The multiplexing capabilities of photodiode-based tracking are therefore limited. In comparison, video microscopy-based bead tracking benefits from the use of vastly increased pixel numbers, at the expense of more complex computations required to convert the images into a time series of bead positions and a reduction in overall speed.

Fortunately, the state-of-the-art in video microscopy-based bead tracking has significantly advanced in recent years in terms of improved accuracy, increased speed, or increased

<sup>a)</sup> Author to whom correspondence should be addressed. Electronic mail: N.H.Dekker@tudelft.nl

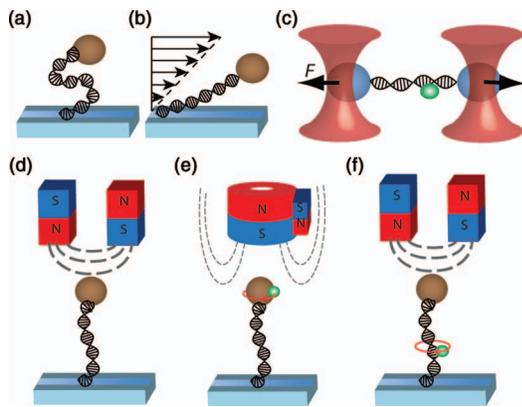


FIG. 1. Overview of single-molecule methods that rely on the tracking of spherical beads. (a) Tethered particle motion approach, in which biological molecules are tethered between a spherical bead and a surface. In this approach, there is no externally applied force acting on the spherical beads. (b) Flow stretching approach, in which biological molecules are tethered to spherical beads that are subjected to force via fluid flow. (c) Dual-beam optical trap approach, in which biological molecules are tethered between two dielectric spherical beads. Focused laser beams apply force to the spherical beads. (d) Magnetic tweezers approach, in which biological molecules are tethered between a magnetic spherical bead and a surface. External magnets apply force to the magnetic bead. (e) Magnetic torque tweezers, in which biological molecules are tethered between a magnetic spherical bead and a surface. A smaller spherical bead can be adhered to the magnetic bead to serve as a fiducial marker in angular tracking. (f) Magnetic tweezers including a rotor bead, in which biological molecules are tethered between a magnetic spherical bead and a surface. The rotor bead is adhered internally to the biological molecule and serves as a fiducial marker in angular tracking.

multiplexing, which we discuss in turn. A commonly used approach for the tracking of spherical beads in two dimensions is based on successive iterations of a 1D cross-correlation algorithm (Gosse and Croquette, 2002). Increased accuracy can be achieved at the expense of computational complexity by using a 2D image correlation method to compute  $x$ ,  $y$  coordinates, with a demonstrated accuracy of 1 nm (van der Horst and Forde, 2008). Alternatively, a more accurate implementation of the 1D cross-correlation algorithm for use in two-dimensional tracking denoted Quadrant Interpolation (QI) (van Loenhout *et al.*, 2012b) may be employed. Frequently, bead tracking algorithms are designed to process information at video rates, but higher tracking rates would be desirable as any subsequent signal smoothing could serve to reduce the influence of unwanted noise. Fortunately, in recent years technologies for high-speed tracking have been introduced, as demonstrated by the real-time video-based tracking of a single bead in an optical trap at 10 kHz rates (Otto *et al.*, 2010). In these experiments, the use of 1D cross-correlation tracking together with smoothing to 1 Hz resulted in a resolution of 2 nm. Similarly, video-based tracking of a single bead in magnetic tweezers at 10 kHz rates has been demonstrated (Lansdorp *et al.*, 2013). In this experiment, however, the image data were buffered in RAM memory prior to processing, which limited the measurement to a few seconds. The multiplexed tracking of spherical beads has also seen advances in recent years. For example, a tracking method relying on 1D cross-correlation was used to track 36 beads online at 60 Hz while maintaining a spatial resolution of 1.5 nm (Ribeck and Saleh, 2008). Work by De

Vlaminck *et al.* (2011) optimized the positioning of magnetic beads to maximize the yield of multiplexed tracking. For the most widespread flexibility in designing experiments and hence greatest impact, however, video microscopy-based bead tracking should combine these three key ingredients of high accuracy algorithms, high acquisition rates, and maximal multiplexing.

In this work, we have developed new software for video microscopy-based bead tracking in all three spatial dimensions that allows us to combine accuracy, speed, and multiplexing. The software, running on a custom-built magnetic tweezers instrument, is built to take into account and integrate recent advances in camera technology and computer hardware. In this way, we are able to carry out accurate tracking of hundreds of beads in parallel, in real time, with nanometer accuracy. Compared to previous real-time multiplexing methods (Ribeck and Saleh, 2008), we have improved the tracking accuracy in the  $x$ ,  $y$  dimensions by implementing Quadrant Interpolation (QI) (van Loenhout *et al.*, 2012b) and removed computational bottlenecks by implementing the core of the computational algorithms on high performance graphics hardware. This facilitates real-time processing, which has the advantage of allowing an experimentalist to perform very long measurements without disk space becoming limiting. We have tested the performance of our software on simulated images to explore the trade-offs in scatter and bias versus tracking speed and we show how these results compare to a theoretically optimal tracking algorithm by computing the corresponding Cramer-Rao Lower-Bound. Finally, we directly test the performance of the software on DNA-tethered magnetic beads, simultaneously tracking hundreds of beads in real time at an acquisition frequency of 60 Hz. The combination of all these advances is highly beneficial for experimentalists, who will not only be able to perform routine experiments much more quickly but will also acquire a wealth of biophysical data that reports on relatively rare events, such as the nucleation of proteins (Miné *et al.*, 2007; van der Heijden *et al.*, 2007) or polymerase pausing (Herbert *et al.*, 2006; Revyakin *et al.*, 2006).

## METHODS

### Apparatus

The magnetic tweezers assay (schematically depicted in Figure 2) is implemented on a custom-designed microscope. A pair of permanent magnets, oriented in a horizontal configuration (Lipfert *et al.*, 2009), applies a magnetic field to a tethered superparamagnetic bead. One stepping motor controls the vertical position of the magnet (M-126-PD, PI, Germany), while the other controls its rotation (C-150, PI, Germany). The beads are imaged with a 50x objective (CFI Plan 50XH, Nikon, Japan) on a 12 megapixel CMOS camera (12M Falcon2, Teledyne Dalsa, Canada) operated in a full camera link mode. The images are transferred from the camera to a frame grabber (PCIe 1433, NI, Texas) to be then processed via the custom Labview 2011 SP1/CUDA software. The  $z$  position of the objective is controlled by a piezo stage (Pifoc, PI, Germany), in order to build up a look-up table (LUT) of bead profiles (see Software design, below).

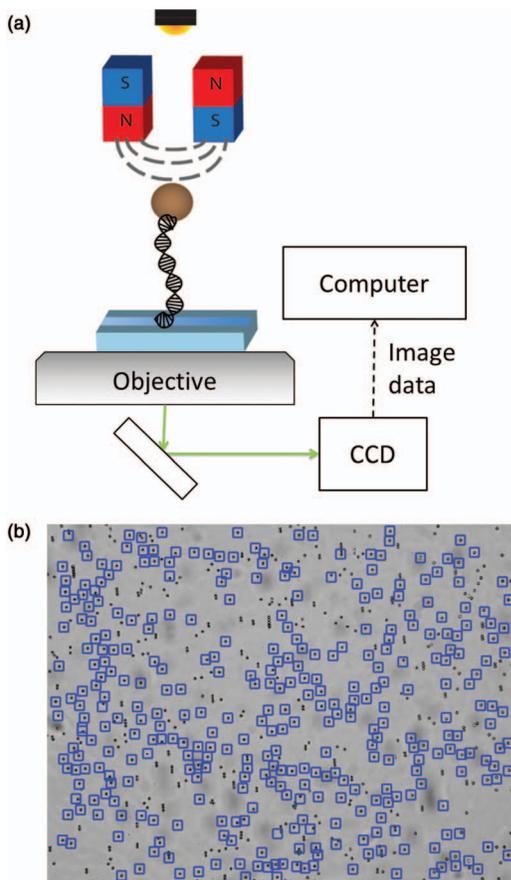


FIG. 2. Magnetic tweezers experimental configuration and typical field of view. (a) Magnetic tweezers experimental configuration shown in more detail compared to Figure 1(d). Surrounding the DNA-tethered magnetic bead are external magnets for force application, an objective and CMOS camera to collect the transmitted light, and the link through which image data are transferred to the computer. (b) Typical field of view in the magnetic tweezers. The field of view has a magnification of  $50\times$ , allowing for hundreds of simultaneously tracked beads. In this image, 349 beads are marked for tracking, each surrounded by a ROI of  $100 \times 100$  pixels (blue boxes). The coordinates of all of these magnetic beads are tracked and computed by the software algorithm in real time.

### Flow cell

Flow cells consist of two coverslips ( $24 \times 60$  mm number 1, Menzel Glazer, Germany). One of the coverslips, to which the tethered beads will be attached, is first coated with a 0.1% (m/V) nitrocellulose dissolved in ethanol. Then, reference beads are attached to this coverslip. For this, stock solution of polystyrene reference beads (Polysciences, USA) is diluted 250 times in the nitrocellulose solution. This solution is placed on the coverslip which is maintained on a hot plate at  $150^\circ\text{C}$  for three min, allowing the nitrocellulose solvent to evaporate and the polystyrene beads to melt onto the surface. The two coverslips are then fixed together using melted Parafilm (Dupont, USA) to form the flow cell.

### Bead tethering

Once the flow cell is mounted on the magnetic tweezers set-up, it is first flushed with buffer (Tris-HCl 10 mM pH 8, EDTA 0.5 mM, 150 mM NaCl, 0.01% Tween-20). Subsequently, anti-digoxigenin (1 g/l,  $40 \mu\text{l}$ ) is incubated in the

flow cell for approximately 30 min. After rinsing the flow-cell with 0.5 ml of buffer,  $50 \mu\text{l}$  of BlockAid (Life Technologies, Netherlands) is incubated for 30 min in the flow-cell and is then flushed away with 0.5 ml of buffer. We used a 21 kbp DNA strand to tether the superparamagnetic beads (streptavidin-coated M270 beads, Life Technologies) to the surface (Lipfert *et al.*, 2010) via its two handles (one of which is multiply labeled with digoxigenin, while the other is multiply labeled with biotin). A  $0.5 \mu\text{l}$  of this DNA (stock concentration  $10 \text{ ng}/\mu\text{l}$ ) is mixed with  $10 \mu\text{l}$  M270 beads and incubated in the flow-cell for 30 min. Following abundant rinsing with buffer, measurements can be started. This protocol produces tethers that are sufficiently stable to do measurements of several hours at forces up to 40 pN. To further increase yield and stability, magnetic tweezers users can consider tethers designed to form covalent bonds with the bead and/or flow cell surface (Janissen *et al.*, 2014).

### Software design

We have implemented software in C++, CUDA, and Labview that is suited to high-throughput tracking in magnetic tweezers. A flow-chart of the design of the software is shown in Figure 3. In short, region-of-interests (ROIs) on the camera image are selected by the user. These are copied from the camera frame by the CPU and placed into batches in which the images should have identical dimensions (Figure 3(a)). The input image data can be 8-bit, 16-bit or floating-point data and, if needed is converted to floating-point before further processing. When a batch has reached a critical size (typically around 500 images), the image data are copied to the GPU (Figure 3(b)), which processes it using the localization algorithms (Figures 3(c)–3(e) and 3(h)). Because there are no dependencies between beads in the localization procedure, the bead images can come from any frame or ROI and can be treated independently from one another, making the computation of bead positions highly suited to parallelization.

We have implemented the localization algorithms using the CUDA programming framework, which allows them to run on highly parallel graphics hardware. This provides high computational power for a relatively low cost. Once implemented in this fashion, the image processing speed can be scaled up by simply adding multiple graphics processing units (GPU) to the host computer. It is important to note, however, that GPU-based programming does add additional complexity to the program logic. For example, parallel computing in the GPU employs several processors, each of which contains internal sub-processors. The GPU performs efficiently only when it has enough parallel computing tasks to do, and when each task is in itself small enough to run on a single internal sub-processor. To achieve this, the code processes several batches simultaneously, each running on a separate CUDA command stream. A given CUDA stream is never blocked by the operations of the other streams that copy to memory images and localization results or wait on CPU commands. Whenever a batch is completed, the positions

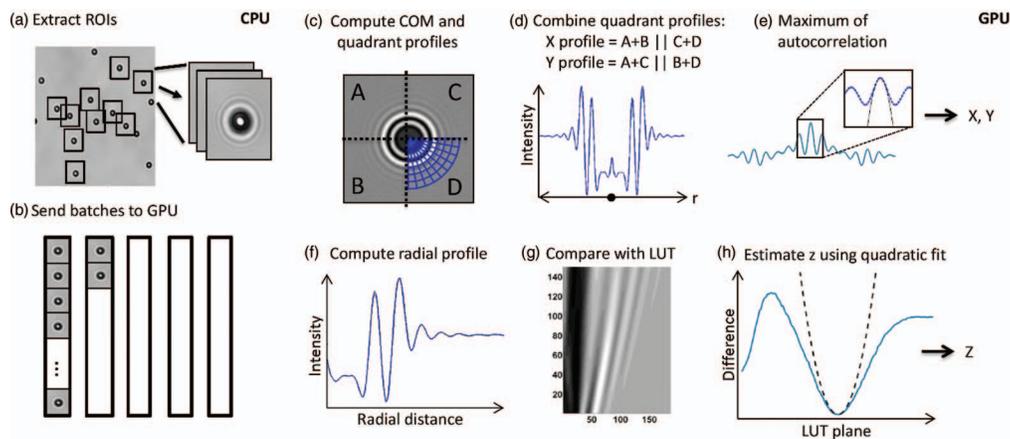


FIG. 3. Flow chart of the localization algorithm. (a) Overview of the steps involved in collecting images for processing. In a first step, the magnetic beads to be tracked are selected, either by user interface or automatic detection. For each bead the location of the ROI is fixed and bead center coordinates are always determined in absolute sense within the ROI. (b) Batches of bead images, which may derive from a single bead as a function of time or from multiple beads at the same time point, are queued together in batches for analysis and sent to the GPU. (c) Within the GPU code, a COM estimate is performed. Using the COM position, radial profiles are constructed for each of the 4 quadrants of a bead image (Q1–Q4). (d) The computed quadrant radial profiles are combined and concatenated (indicated by ||) to prepare for FFT autocorrelation. (e) The  $x$  and  $y$  profiles are convolved with their mirror versions using 1D FFT operations, after which the resulting autocorrelation signals can be used to refine the  $x$  and  $y$  positions. (f) To estimate  $z$ , a new radial profile is constructed by sampling the bead image with the refined  $x, y$  positions as its center. (g) The radial profile is compared with all the pre-recorded radial profiles in the LUT by computing the squared differences. (h) The bead is likely close to LUT planes that have matching radial profiles. To compute a sub-plane  $z$  position, a 5-point fit is done around the smallest difference value.

computed by the localization algorithms are transferred to regular system memory and saved to disk.

The completion of a batch implies that the  $x, y, z$  coordinates of the beads in all of the images have been determined. The flow of computations used to determine these coordinates is depicted in Figures 3(c)–3(h). In short, localization of each bead image starts by computing a first estimate of its center  $x, y$  coordinates using a center-of-mass (COM) computation, followed by further refinement using the quadrant interpolation (QI) algorithm (van Loenhout *et al.*, 2012b) (Figures 3(c)–3(e)). The QI algorithm is essentially an improved version of conventional 1D cross-correlation algorithms that are frequently employed in bead tracking applications (Gosse and Croquette, 2002; Lansdorp *et al.*, 2013; Ribeck and Saleh, 2008). To localize a spherical bead's  $x, y$  coordinates using 1D cross-correlation algorithms, one typically samples the image in a horizontal band and a vertical band that are respectively auto-correlated to compute the  $x, y$  shifts at which each band profile best mirrors itself. Rather than sampling two rectangular bands, the QI algorithm instead takes into account the spherical symmetry of the bead and samples each quadrant of the bead shape in a grid of polar coordinates (Figure 3(c)) from which  $x$  and  $y$  profiles can be reconstructed (Figure 3(d)) for auto-correlation (Figure 3(e)). The maximum of this auto-correlation function is used to shift the original  $x, y$  coordinates to more accurate values (the interested reader is referred to van Loenhout *et al.* (2012b) for further details). In practice, the QI algorithm can be executed multiple times per bead image to further refine these  $x, y$  positions.

Localization then proceeds by estimating a bead's  $z$  position is performed by creating a radial profile using the refined  $x, y$  coordinates (Figure 3(f)) and comparing this profile to a pre-recorded LUT of radial profiles (Figure 3(g)) (Gosse and Croquette, 2002). Such a LUT, of which one is stored

per bead, is generated at the very beginning of an experiment by moving the focal position through a range of  $z$  positions. For each focal position, the bead is localized in  $x, y$  using QI and the radial profile is computed and stored in the LUT. Each radial profile in the LUT is also normalized to correct for fluctuations in lighting by subtracting its mean intensity and then dividing by the root mean square of the intensity. In our experimental configuration, we found a 50 nm step size in the LUT to be optimal: for step sizes exceeding 100 nm, the  $z$  resolution was observed to decrease, whereas for step sizes below 50 nm, the LUT building process took a proportionally longer time, rendering it more vulnerable to the incorporation of drift in the  $z$  coordinate. A comparison of a measured radial profile to the stack of profiles contained in the LUT is performed by computing the squared difference with each LUT profile (Figure 3(h)). The bead's actual  $z$  position is then deduced by finding the closest corresponding radial profile in the LUT and computing a five-point least-squares quadratic fit on the difference values surrounding the closest matching radial profile (Figure 3(h)). The quadratic fit permits  $z$  localization to be more precise than the distance between LUT planes, but does come with some limitations (see Correction of look-up table bias, below).

### Theoretical estimate of the scatter

There is a limit to the amount of information that is encoded in any image and hence a limit to the information that can be extracted irrespective of the algorithm employed. If we have a set of random variables whose distributions depend on a set of parameters, the Cramer-Rao Lower Bound (CRLB) (van den Bos, 2007) describes a lower bound on the variance of an estimator that estimates the parameters given a sample of the random variables. In our case, the intensity of every

pixel from the measured image is assumed to derive from a Poisson distribution with a mean value that depends on the  $x$ ,  $y$ , and  $z$  coordinates. Calculating the CRLB thus provides insight into the amount of information about the bead position stored in any given image. In fluorescence microscopy, the CRLB has been computed to determine the theoretical limit for the localization of point spread functions modeled via 2D Gaussian functions (Smith *et al.*, 2010). We have similarly applied the CRLB to the case of spherical bead localization by replacing the 2D Gaussian with a measured LUT that defines the probability distribution for each pixel in the image.

We now describe the computation of the CRLB in our LUT model. In general, the CRLB is defined as  $\text{var}(\hat{\theta}) \geq I(\theta)^{-1}$ , with  $\hat{\theta}$  being the estimator of the parameters  $\theta = (\theta_x, \theta_y, \theta_z)$  and  $I(\theta)$  being the Fisher information matrix (van den Bos, 2007). This implies that the covariance matrix  $I(\theta)^{-1}$  will define the theoretical minimum variances for the bead coordinates  $x$ ,  $y$ ,  $z$ . To compute  $I(\theta)$  in our LUT model, we assume that the pixel values in our images are described by a Poisson distribution, as would be expected for camera read-out with noise dominated by shot-noise. We can then define the expected value  $\mu_k$  of each pixel  $k$  as a function of the parameters  $\theta_x, \theta_y, \theta_z$ :

$$\mu_k(\theta_x, \theta_y, \theta_z) = LUT(\sqrt{(\theta_x - x_k)^2 + (\theta_y - y_k)^2}, \theta_z). \quad (1)$$

Here,  $LUT(r, z)$  defines the lookup table a function of the radial distance  $r$  from the center of the bead (whereby  $(x_k, y_k)$  are the center pixel coordinates for pixel  $k$ ) and the  $z$  position in the LUT. To compute the value of  $LUT(r, z)$ , we perform a linear interpolation between the radial bin values in the LUT. Values of  $r$  and  $z$  that are outside of the recorded range are clamped to the closest known LUT value. Pixelation errors caused by only evaluating  $\mu_k(\dots)$  at the pixel center were determined to be negligible, as CRLB values are not noticeably influenced when  $\mu_k(\dots)$  is computed using an average of 16  $LUT(r, z)$  values in a  $4 \times 4$  grid within each pixel. The values of  $\mu_k$  can then be used to compute the Fisher information matrix for an image with Poisson-distributed pixel values (Smith *et al.*, 2010) can be computed according to

$$I_{ij}(\theta) = \sum_k^N \frac{1}{\mu_k(\theta)} \frac{\partial \mu_k(\theta)}{\partial \theta_i} \frac{\partial \mu_k(\theta)}{\partial \theta_j}. \quad (2)$$

In this equation,  $i$  and  $j$  can be substituted by  $x$ ,  $y$ , or  $z$ , for example, to compute the variance of  $x$  ( $I_{xx}$ ), or the covariance of  $x$  and  $z$  ( $I_{xz}$ ). We assume that the pixel intensity values are high enough for Stirling's approximation ( $\ln n! \approx n \ln n - n$ ) to hold. This is reasonable, since in typical bright-field imaging conditions mean pixel intensity values are around 150 grey levels (8-bit camera), as a result of which the use of Stirling's approximation contributes a relative error smaller than 1%. To compute values for  $\frac{\partial \mu_k(\theta)}{\partial \theta_i}$ , we applied straightforward numerical differentiation using the central difference method ( $\frac{\partial f}{\partial x} \approx \frac{f(x+h) - f(x-h)}{2h}$  for small values of  $h$ ).

As our computation of the CRLB relies on an experimentally measured LUT for input, it is important to assess the potential influence of noise. To do so, we computed a low-noise radial profile by selecting 10 images per LUT plane and averaging their profiles. Comparison of the CRLB of a LUT

based on 1 image per plane with this 10-image-per-plane LUT displayed no noticeable differences in the CRLB values, indicating that the noise in a single LUT is sufficiently low for the computation of an accurate CRLB value. All displayed computations of the CRLB for video-based spherical bead tracking (Figures 5 and 6) therefore rely on single LUTs. These computations provide quantitative estimates of the best accuracy that a tracking algorithm could achieve, given the information present in the image of a spherical bead under the tested conditions.

### Correction of look-up table bias

Similarly to previous methods (for instance Lansdorp *et al.*, 2013), our  $z$  localization method applies a quadratic fit to five points around the minimum of the squared difference (Figure 3(h)). While the neighborhood of this minimum approximates a quadratic function, it is not an exact match. We have computed the bias resulting from this mismatch throughout the lookup table and found that it can cause deviations in the tracking of  $z$  coordinates of up to 20 nm (depending on the bead position within the LUT). One way to address this would be to utilize a better-matched fitting function (e.g., a higher-order polynomial). However, doing so will likely also require more points to accurately produce a fit, and hence increase the computational complexity. Instead, we have opted for a more pragmatic approach that consists first of estimating the LUT bias by localization on images extracted from the LUT prior to the experiment, and second of correcting for this LUT bias on all subsequently tracked images. The advantage of this approach is its rapid, straightforward implementation. Furthermore, this implementation can also potentially be employed to correct other forms of bias such as experimentally determined bias.

To estimate the LUT bias, a set of bead images is extracted from the LUT at a number of steps within the range of possible  $z$  values and processed by the tracking algorithm. Bead images are generated by evaluating  $\mu_k(0, 0, \theta_z)$  (Eq. (1)) for each pixel  $k$ , whereby the bead is placed in the center of the generated image. Localization is then performed on this generated image, and the resulting  $z$  error is stored in separate table (displayed in Figure 5(c), inset) that contains the LUT bias for every bead throughout the LUT range. We used five bias correction points for every LUT plane. Given a biased estimate of the  $z$  position, the tracking algorithm has to find the unbiased estimate by using the bias correction table, essentially inverting the function defined by this table. In other words, we can look up  $Bias(z_{measured})$ , but we wish to determine the value of  $z$  for which  $z_{true} = z_{measured} + Bias(z_{true})$  holds. By assuming that  $Bias(z_{measured}) \approx Bias(z_{true})$ , we can obtain a less biased estimate for  $z_{true}$ . Iteration of this computation will converge towards a  $z$  position that is unbiased, at least according to our pre-computed table. In pseudo-code, this can be written as follows:

```
Z ← MeasuredZ + Bias(MeasuredZ)
For N iterations do
  Z ← MeasuredZ - Bias(Z)
```

The evaluation of  $Bias(z)$  is performed using linear interpolation of the values in the bias correction table. In our case, four iterations were sufficient to converge to a stable position.

## RESULTS

We have tested our tracking software on both simulated and experimental data. Performing localization on simulated data allowed for a wide range of configurations to be tested, while the experimental measurements show the practical value of the approach we introduce.

Using simulated images, we explored different aspects of the tracking software including its accuracy, the effect of multiple QI iterations, and the speed of different implementations. To create a simulated bead image, a random position was generated from a  $200\text{ nm} \times 200\text{ nm} \times 200\text{ nm}$  box with a uniform distribution, after which the image's pixel values were calculated by interpolation of an experimentally measured LUT using the definition of  $\mu_k(\theta_x, \theta_y, \theta_z)$  (Methods). The center of this uniform distribution in  $x, y$  coordinates is the center of the image, and the  $z$  coordinate of the bead is sampled around the center of the LUT (100 nm in each direction). We then applied Poisson noise to the image at a level that matches the experimental configuration. On such a set of simulated images with known coordinates, we quantify the performance of the tracking software by computing the *bias* and the *scatter*. Here, *bias* refers to a tracking error that depends on the bead position, while *scatter* refers the uncorrelated tracking error primarily caused by shot-noise.

The tracking performance is effectively determined by how much information about the bead position reaches the camera. Factors influencing this are the pixel size, the size of the bead, the magnification, and the amount of light. We assume that the ROI is always chosen large enough to capture the full bead shape, and that lighting conditions are appropriately chosen to employ the full range of pixel intensities. To compare algorithms, we ran tracking simulations using the COM tracking algorithm, conventional 1D cross-correlation algorithms (Gosse and Croquette, 2002; Lansdorp *et al.*, 2013; Ribeck and Saleh, 2008), and up to three iterations of the QI algorithm. In addition to the number of iterations, the number of points in the polar grid that the QI algorithm uses to convert an image into quadrant profiles also strongly influences the processing speed. The number of points can be chosen in such a way that no pixels are skipped and therefore no information is lost. The specific values for this can be found in our software documentation. As the ROI size is increased, one can observe that the scatter and bias errors in tracking decrease for various tracking algorithms tested (Figure 4). This is evident for images analyzed in both the  $x$  (Figures 4(a) and 4(b)) and  $z$  (Figures 4(c) and 4(d)) dimensions (analysis in the  $y$  dimension is omitted, as it yields similar results to the  $x$  dimension given the spherical bead shape). Each data point in the graph was computed by running the localization algorithm on 300 simulated images. Pixel size and LUT step size were matched to our experimental setup (118 nm/pixel and 50 nm, respectively). For reference, a ROI of 100 pixels is suitable for tracking  $2.8\text{ }\mu\text{m}$  diameter magnetic beads at a magnification of  $50\times$ . The decrease of scatter

and bias with increasing ROI are to be expected, as at higher ROI sizes, more pixels contribute to the positional information available in the image. This trend is also reflected in the CRLB.

Using these simulations, we can quantitatively compare different bead tracking algorithms (Figures 4(a)–4(d)). While all bead tracking algorithms show the same general trend of a decrease in scatter and bias error with an increased ROI, there are substantial differences among them. Center-of-mass (COM) localization is known to have a large pixelation bias (van Loenhout *et al.*, 2012b), which is reflected in the relatively poor results shown in the  $x, y$  scatter and bias plots (Figures 4(a) and 4(b)). In comparison, the QI tracking algorithm yields greatly improved results for  $x, y$  tracking. The benefit of executing multiple QI iterations is, however, limited to the lower ROI sizes (ROI < 90 pixels). Especially if the  $z$  position is the only value of interest (as is often the case), running a single QI or Cross-Correlation iteration suffices for accurate  $z$  tracking. For each ROI, we have also computed and plotted the CRLB, which indicates the minimum achievable scatter that could be obtained with a perfect bias-free tracker. This indicates that the best achievable scatter value that we achieve exceeds the CRLB by only a factor of two.

For this simulated dataset, we have computed the processing speed as a function of ROI size (Figure 4(e)). The processing speed was estimated by measuring the time required to process 60 000 8-bit images, where the large number of images was required to smooth out any time delays caused by CUDA batching or CPU caching. An nVidia Geforce 690 card was used to test the CUDA implementation, and a 6-core Xeon E5-1650 processor was used to test a separate CPU-only C++ implementation. To make a practical and fair comparison, each of the algorithms included a lookup step in the  $z$  coordinate. It should be noted however, that this makes COM computation appear a lot slower than it is. For example, the COM processing speed is 81 kHz for a ROI of  $100 \times 100$  pixels, but if the  $z$  coordinate is also computed, this speed drops to 41 kHz. Whichever algorithm is used, it can be seen that the processing speed is roughly inversely proportional to the number of pixels in the ROI (a ROI of  $160 \times 160$  pixels will be processed four times more slowly than a ROI of  $80 \times 80$  pixels). The specifics of the implementations of Fast-Fourier-Transforms used by the 1D Cross-Correlation method and QI result in drops processing speed for ROI sizes of  $\sim 70$  and  $\sim 130$  pixels. This can be understood from the fact that FFTs are computed using array lengths based on powers of two: at these sizes, the algorithm switches to a larger array length. Overall, these processing speed measurements demonstrate the clear benefit of using graphics hardware for localization algorithms: tests performed using a single graphics card outperform by a factor of three in speed the current state-of-the-art Xeon CPU that can simultaneously process 12 localization processes. It can be seen that the software can track up to 770 beads in real-time at 60 Hz, if a ROI size of  $80 \times 80$  pixels is chosen.

Our measurements on individual CUDA command streams indicated that the image processing is limited by computational speed, as opposed to data transfer speed or available video memory size. In a simulation with  $100 \times 100$  ROIs

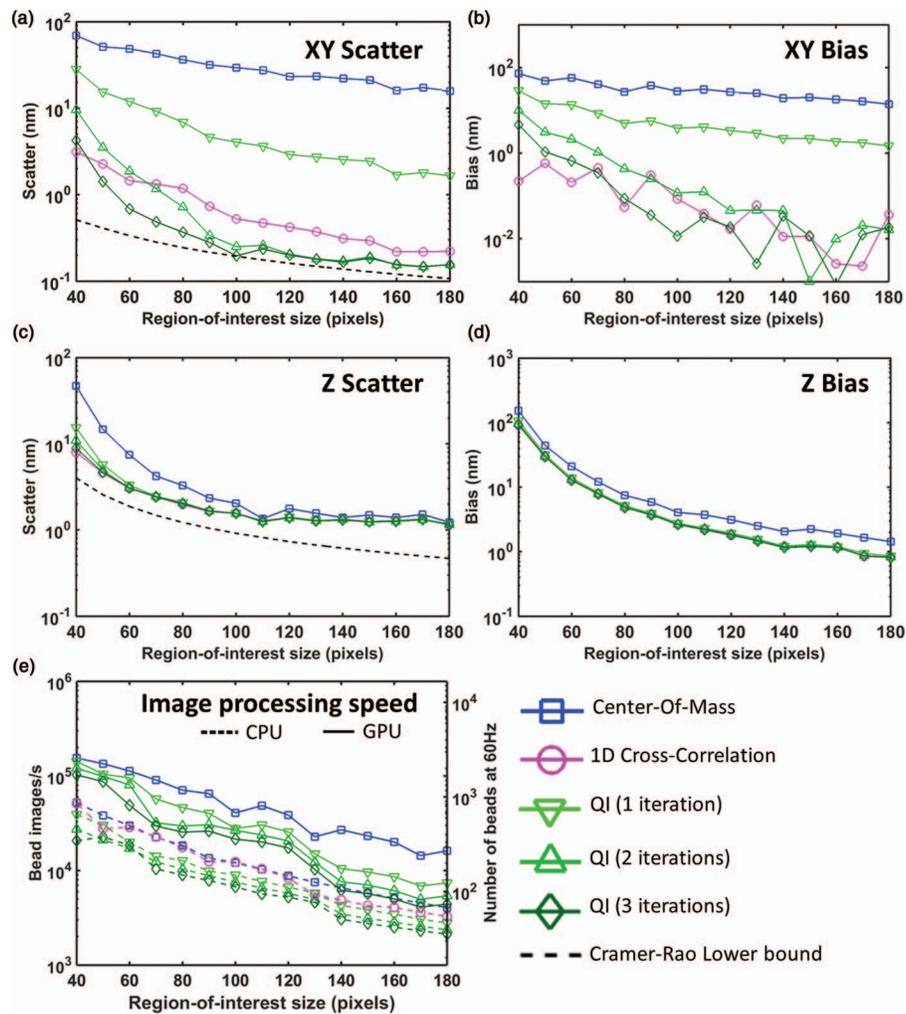


FIG. 4. Simulated bias, scatter, and image processing speed in the tracking of spherical beads as a function of region-of-interest-size. (a) Scatter in the determination of a bead's  $x$  coordinate as a function of ROI size. (b) Bias in the determination of a bead's  $x$  coordinate as a function of ROI size. (c) Scatter in the determination of a bead's  $z$  coordinate as a function of ROI size. (d) Bias in the determination of a bead's  $z$  coordinate as a function of ROI size. (e) Image processing speed as a function of ROI size. Note that a comparison between computational speed obtained on a GPU card and obtained on a high-end CPU is included. For all plots, the mentioned quantities are evaluated for the use of different tracking algorithms or their number of iterations, as indicated in the legend. For every datapoint in this plot, 300 localizations are performed. The black dotted line shows the theoretical lower limit of an unbiased tracker as provided by the CRLB (see Methods section).

and 2 QI iterations, we found that 14% of the GPU time was spent on COM computation, 62% on QI iterations, and 24% on  $z$  computation. Image transfer to the GPU and result transfer back to the CPU does not involve GPU computations and is essentially done “for free” by other parts of the GPU during the time that other batches are being processed. In the same simulation, the process allocated  $\sim 216$  MB of GPU memory for storage of image data and temporary variables, enough for buffering 1024 bead images for parallel processing). As modern GPUs contain gigabytes of memory space, memory size is never a limiting factor

Using the simulations, we also illustrate the effect of tracking at different locations in the LUT (Figure 5). Determining this trade-off as a function of LUT position is important, since a user of the tracking software must decide both in what  $z$  range the LUT should be built and in which focal plane to run the experiment. Therefore, to estimate the effect of focal plane position, we have measured the LUT over a long range of focal plane positions (Figure 5(a)), and performed

a tracking simulation using images generated from that LUT. Scatter, bias, and the lower bound on the scatter (CRLB) were determined for both the  $x$ ,  $y$  (Figure 5(b)) and  $z$  (Figure 5(c)) dimensions. To illustrate the effect of our  $z$  bias correction, Figure 5(c), inset shows the  $z$  bias if no bias correction (Methods) is applied. At the right side of the focal plane, this is on the order of a few nm. In a typical experiment, a bead's  $z$  position is monitored over a few hundred nanometer length, and the amount displacement of the bead is used to infer information about the tether. The change in  $z$  bias will influence this measured displacement, shown as a relative error (Figure 5(d)). The  $z$  bias correction is effective in reducing the error in a bead movement measurement.

These plots were generated by simulating tracking throughout the lookup table range with 1000 steps. For every step, 200 images were generated and tracked. As can be seen for both  $x$ ,  $y$ , and  $z$  tracking, there is an optimal region in the lookup table in which the scatter is small and close to the CRLB. Surprisingly, the scatter also drops below the CRLB

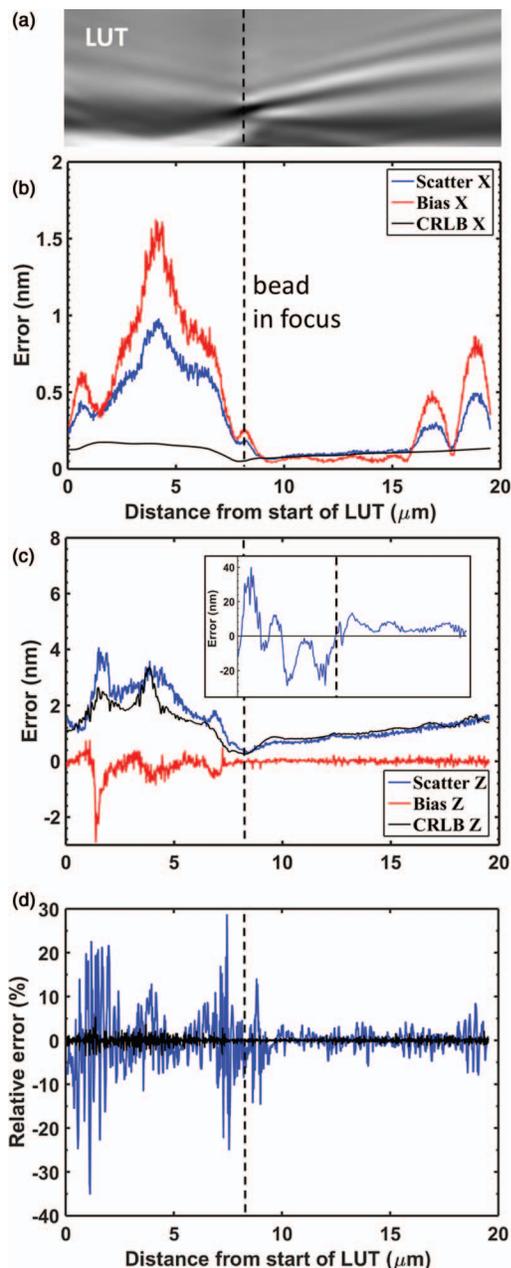


FIG. 5. Simulated bias and scatter in the tracking of spherical beads as a function of the position in the look-up table. (a) Experimentally measured look-up table used to generate simulated images from. (b) Scatter (blue), bias (red), and the minimum theoretical scatter as computed according to the CRLB (black) in  $x$  and  $y$  dimensions. The bias is absolute, to improve readability. (c) Scatter, bias, and the CRLB for  $z$  tracking. (c, inset) The combination of a quadratic least square fit with a radial profile look-up-table results in a large constant bias that depends directly on the  $z$  position. This bias is calculated prior to performing measurements and is used to correct errors in the tracking of  $z$  coordinates (see Methods section). (d) The relative error in bead  $z$  movements due to bias, with correction (black) and without correction (red). For example, a relative error of 10% means that after moving 10 nm, the  $z$  bead position is measured to be 11 nm further than it should be.

at some points. This may be due to bias: the CRLB that we have computed is a bound on an optimal, *unbiased* estimator. If there is bias present, it can push the coordinates in a particular direction and hence cause a reduced scatter. Even though the data shown here are derived from simulations, physical effects due to the optics still affect tracking through the fea-

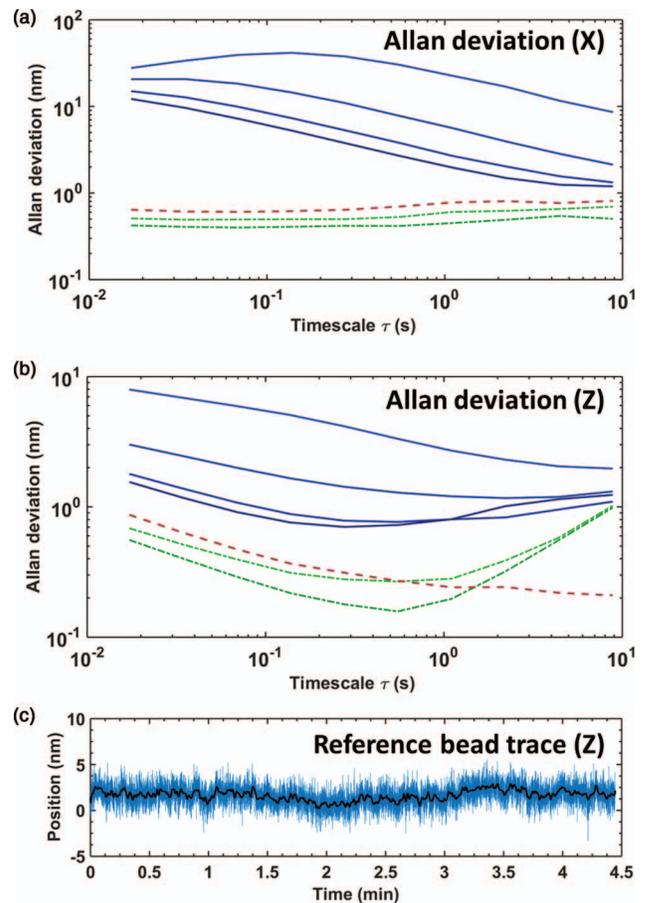


FIG. 6. Tracking of DNA-tethered beads in the magnetic tweezers. (a) Allan deviations plotted for measured  $x$  coordinates. The blue lines represent this quantity as a function of force (2.51 pN, 10.2 pN, 20.9 pN, and 29.6 pN, increasing forces arranged from light blue to dark blue). Each line corresponds to the average over 26 beads. The dotted green lines correspond to two distinct reference beads and the black dashed line reflects the Allan variance that is computed once the  $x$  coordinate of one reference bead is subtracted from that of the other. (b) Allan deviations plotted for measured  $z$  coordinates, same selection of bead traces used in (a). The optimal stability of 0.1 nm is achieved at a timescale of 0.5 s (2 Hz). On longer timescales, drift affects  $z$  tracking more than  $x$  tracking. (c) It can be seen that drift-like effects are still present in the signal. Even when one reference bead  $z$  position is subtracted from another, correlated fluctuations on the order of  $\sim 1$  nm remain visible in the signal.

tures in the LUT. Below the focal plane position ( $< 8 \mu\text{m}$  in the plot), the radial profiles lose their distinct patterns, likely due to optical aberrations (the objective is corrected to image objects above the focal position), and tracking scatter and bias increase as a consequence in both  $x$ ,  $y$ , and  $z$  axes. As before, we compare the results of our simulations to the CRLB (Figures 5(b) and 5(c)). As the CRLB estimate for the theoretically lowest  $z$  scatter is very close to the  $z$  tracking results, we conclude that our  $z$  tracking scatter, at least in the typical measurement region, is very close to optimal.

Using this input from simulations, we can now fully test our software by performing an experiment in the magnetic tweezers (Figure 6). In this experiment, we selected a field of view containing 228 tethered magnetic beads, each captured in a ROI of  $80 \times 80$ , and tracked them simultaneously at 58 Hz using three iterations of the QI implementation. From this dataset, we then selected 26 tethered beads for further

analysis on the basis of their having only a single biomolecule of the proper length tethered to their south poles, a commonly employed criterion in single-molecule biophysics experiments. To quantitatively analyze these beads, we compute their mean Allan variance (Czerwinski *et al.*, 2011; Lansdorp *et al.*, 2013), a measure of signal stability over a given timescale  $\tau$  that allows one to directly view the magnitude of drift and uncorrelated noise, in both the  $x$  (Figure 6(a), blue curves) and  $z$  (Figure 6(b), blue curves). The same quantities were computed for two reference beads that were simultaneously tracked (Figures 6(a) and 6(b), green curves). In all cases, we observe that the Allan variance initially rises, peaks, and falls, prior to rising again. The final rise is typically attributed to experimental drift, which will dominate at long timescales despite drift compensation through reference bead subtraction. The Allan variances are largest for the DNA-tethered beads subjected to the lowest forces, and they are smallest for reference beads attached directly to the surface. This is due to the low stiffness of DNA-tethered beads pulled upon at low forces and the comparatively higher stiffness of surface-tethered reference beads. A differential measurement, in which the  $z$  coordinate of a one of the reference beads shown in Figures 6(a) and 6(b) is subtracted from the other reference bead, demonstrates that any remaining effects of drift (correlated noise) contribute to an amplitude of only  $\sim 1$  nm (Figure 6(c)). Thus, these measurements demonstrate that our tracking algorithm is capable of analyzing large datasets, at high resolution, in a single experimental run.

## DISCUSSION AND CONCLUSIONS

We have advanced key aspects of the instrumentation that is required for the tracking of spherical beads, including both software and hardware advances. In terms of software, we have created a new software library designed to do accurate bead tracking in a fast and highly parallel manner. Due to the parallelism of CUDA code, the software can benefit well from new hardware improvements or the use of additional GPU's installed in the system. As such, the combined improvements in software and hardware allow us to vastly parallelize the tracking of spherical beads in magnetic tweezers, making real-time tracking of the coordinates of tens or even hundreds of beads possible. To aid the user, we have shown the limits of tracking bias and scatter as a function of ROI size, position of the bead in the LUT, type of tracking algorithm employed, and computational speed. These limits are compared to the theoretical limit as indicated by the CRLB, which indicate that these advances in tracking approach the theoretical limit to within a factor of two in practical settings. We have also highlighted a bias in  $z$  coordinate estimation that results from the quadratic least square fitting employed in by many previous bead tracking approaches, and proposed a pragmatic solution for correcting the resulting  $z$  bias with a LUT. Together, these improvements allow us to simultaneously track hundreds of spherical beads in magnetic tweezers in real time (770 beads at 60 Hz, for example), with low bias and scatter.

With these advances, high-throughput real-time single-molecule studies come within reach. This is of key impor-

tance, because increased parallelism in tethered bead experiments will greatly improve the statistics obtainable in biophysical experiments. Such improved statistics will allow us to fill in many missing gaps in, for instance, our knowledge of molecular motors still has many missing gaps. For example, the stochastic behavior of polymerase backtracking (Shaevitz *et al.*, 2003) or error incorporation are likely to be reflected in dwell-time distributions, which can make detailed models of motor functioning provided that sufficient statistics underpin their proper analysis. Additionally, our real-time tracking provides significant benefits to experiments that involve rare events, such as nucleation events (Miné *et al.*, 2007; van der Heijden *et al.*, 2007). Detecting such events will be more likely with a measurement system that is not limited in runtime.

Our C++/CUDA tracking software is open-source and available at <http://www.github.com/jcnossen/qtrk>. A Labview binding is also provided to allow users to integrate the software into their own measurement systems. CUDA-supporting hardware is recommended for high-throughput applications. Additionally, our Labview measurement software that integrates the tracking code is available at <http://www.github.com/jcnossen/BeadTracker>.

## ACKNOWLEDGMENTS

We thank Bojk Berghuis for help with experiments, Margreet Docter for advice on coding, and Jacob Kerssemakers for advice on coding and assistance with figures. This work was financially supported by the Netherlands Organisation for Scientific Research (NWO) and by the European Research Council (ERC) through a Starting Grant.

- Abbondanzieri, E. A., Greenleaf, W. J., Shaevitz, J. W., Landick, R., and Block, S. M., "Direct observation of base-pair stepping by RNA polymerase," *Nature (London)* **438**(7067), 460–465 (2005).
- Andersson, M., Czerwinski, F., and Oddershede, L. B., "Optimizing active and passive calibration of optical tweezers," *J. Optics* **13**(4,SI), 044020 (2011).
- Crut, A., Koster, D. A., Seidel, R., Wiggins, C. H., and Dekker, N. H., "Fast dynamics of supercoiled DNA revealed by single-molecule experiments," *Proc. Natl. Acad. Sci. U.S.A.* **104**(29), 11957–11962 (2007).
- De Vlaminck, I., Henighan, T., van Loenhout, M. T. J., Pfeiffer, I., Huijts, J., Kerssemakers, J. W. J., Katan, A. J., van Langen-Suurling, A., van der Drift, E., Wyman, C., and Dekker, C., "Highly parallel magnetic tweezers by targeted DNA tethering," *Nano Lett.* **11**(12), 5489–5493 (2011).
- Gelles, J., Schnapp, B. J., and Sheetz, M. P., "Tracking kinesin-driven movements with nanometre-scale precision," *Nature (London)* **331**(6155), 450–453 (1988).
- Gibson, G. M., Leach, J., Keen, S., Wright, A. J., and Padgett, M. J., "Measuring the accuracy of particle position and force in optical tweezers using high-speed video microscopy," *Opt. Express* **16**(19), 14561–14570 (2008).
- Gore, J., Bryant, Z., Nöllmann, M., Le, M. U., Cozzarelli, N. R., and Bustamante, C., "DNA overwinds when stretched," *Nature (London)* **442**(7104), 836–839 (2006).
- Gosse, C. and Croquette, V., "Magnetic tweezers: micromanipulation and force measurement at the molecular level," *Biophys. J.* **82**(6), 3314–3329 (2002).
- Haber, C. and Wirtz, D., "Magnetic tweezers for DNA micromanipulation," *Rev. Sci. Instrum.* **71**(12), 4561 (2000).
- Herbert, K. M., La Porta, A., Wong, B. J., Mooney, R. A., Neuman, K. C., Landick, R., and Block, S. M., "Sequence- resolved detection of pausing by single RNA polymerase molecules," *Cell* **125**(6), 1083–1094 (2006).

- Janissen, R., Berghuis, B. A., Dulin, D., Wink, M., van Laar, T., and Dekker, N. H., "Invincible single molecules: covalent DNA anchoring strongly enhances the measurable time window and force range," *Nucl. Acids Res.* **42**(18), e137 (2014).
- Jannasch, A., Mahamdeh, M., and Schäffer, E., "Inertial effects of a small brownian particle cause a colored power spectral density of thermal noise," *Phys. Rev. Lett.* **107**(22), 228301 (2011).
- Kim, S., Blainey, P. C., Schroeder, C. M., and Xie, X. S., "Multiplexed single-molecule assay for enzymatic activity on flow-stretched DNA," *Nat. Methods* **4**(5), 397–399 (2007).
- Lansdorp, B. M., Tabrizi, S. J., Dittmore, A., and Saleh, O. A., "A high-speed magnetic tweezer beyond 10,000 frames per second," *Rev. Sci. Instrum.* **84**(4), 044301 (2013).
- Le Gall, A., Perronet, K., Dulin, D., Villing, A., Bouyer, P., Visscher, K., and Westbrook, N., "Simultaneous calibration of optical tweezers spring constant and position detector response," *Opt. Express* **18**(25), 26469–26474 (2010).
- Lipfert, J., Hao, X., and Dekker, N. H., "Quantitative modeling and optimization of magnetic tweezers," *Biophys. J.* **96**(12), 5040–5049 (2009).
- Lipfert, J., Kerssemakers, J. W. J., Jager, T., and Dekker, N. H., "Magnetic torque tweezers: measuring torsional stiffness in DNA and RecA-DNA filaments," *Nat. Methods* **7**(12), 977–980 (2010).
- Lipfert, J., Klijnhout, S., and Dekker, N. H., "Torsional sensing of small-molecule binding using magnetic tweezers," *Nucl. Acids Res.* **38**(20), 7122–7132 (2010).
- Lipfert, J., Wiggin, M., Kerssemakers, J. W. J., Pedaci, F., and Dekker, N. H., "Freely orbiting magnetic tweezers to directly monitor changes in the twist of nucleic acids," *Nat. Commun.* **2**, 439 (2011).
- Miné, J., Disseau, L., Takahashi, M., Cappello, G., Dutreix, M., and Viovy, J.-L., "Real-time measurements of the nucleation, growth and dissociation of single Rad51-DNA nucleoprotein filaments," *Nucl. Acids Res.* **35**(21), 7171–7187 (2007).
- Neuman, K. C., and Block, S. M., "Optical trapping," *Rev. Sci. Instrum.* **75**(9), 2787–809 (2004).
- Otto, O., Czerwinski, F., Gornall, J. L., Stober, G., Oddershede, L. B., Seidel, R., and Keyser, U. F., "Real-time particle tracking at 10,000 fps using optical fiber illumination," *Opt. Express* **18**(22), 22722–22733 (2010).
- Revyakin, A., Liu, C., Ebricht, R. H., and Strick, T. R., "Abortive initiation and productive initiation by RNA polymerase involve DNA scrunching," *Science* **314**(5802), 1139–1143 (2006).
- Ribeck, N. and Saleh, O. A., "Multiplexed single-molecule measurements with magnetic tweezers," *Rev. Sci. Instrum.* **79**(9), 094301 (2008).
- Schafer, D. A., Gelles, J., Sheetz, M. P., and Landick, R., "Transcription by single molecules of RNA polymerase observed by light microscopy," *Nature (London)* **352**(6334), 444–448 (1991).
- Shaevitz, J. W., Abbondanzieri, E. A., Landick, R., and Block, S. M., "Backtracking by single RNA polymerase molecules observed at near-base-pair resolution," *Nature (London)* **426**(6967), 684–687 (2003).
- Smith, C. S., Joseph, N., Rieger, B., and Lidke, K. A., "Fast, single-molecule localization that achieves theoretically minimum uncertainty," *Nat. Methods* **7**(5), 373–375 (2010).
- Smith, S. B., Finzi, L., and Bustamante, C., "Direct mechanical measurements of the elasticity of single DNA molecules by using magnetic beads," *Science (N. Y.)* **258**(5085), 1122–1126 (1992).
- Strick, T. R., Allemand, J. F., Bensimon, D., Bensimon, A., and Croquette, V., "The elasticity of a single supercoiled DNA molecule," *Science (N. Y.)* **271**(5257), 1835–1837 (1996).
- Strick, T. R., Allemand, J. F., Bensimon, D., and Croquette, V., "Behavior of supercoiled DNA," *Biophys. J.* **74**(4), 2016–2028 (1998).
- Tolić-Nørrelykke, S. F., Rasmussen, M. B., Pavone, F. S., Berg-Sørensen, K., and Oddershede, L. B., "Stepwise bending of DNA by a single TATA-box binding protein," *Biophys. J.* **90**(10), 3694–3703 (2006).
- Van den Bos, A., *Parameter Estimation for Scientists and Engineers* (Wiley & Sons, 2007), retrieved from <http://eu.wiley.com/WileyCDA/WileyTitle/productCd-0470147814.html>.
- Van der Heijden, T., Seidel, R., Modesti, M., Kanaar, R., Wyman, C., and Dekker, C., "Real-time assembly and disassembly of human RAD51 filaments on individual DNA molecules," *Nucl. Acids Res.* **35**(17), 5646–5657 (2007).
- Van der Horst, A. and Forde, N. R., "Calibration of dynamic holographic optical tweezers for force measurements on biomaterials," *Opt. Express* **16**(25), 20987–21003 (2008).
- van Loenhout, M. T. J., de Grunt, M. V., and Dekker, C., "Dynamics of DNA supercoils," *Science* **338**(6103), 94–97 (2012a).
- van Loenhout, M. T. J., De Vlamincq, I., Flebus, B., den Blanken, J. F., Zweifel, L. P., Hoening, K. M., and Dekker, C., "Scanning a DNA molecule for bound proteins using hybrid magnetic and optical tweezers," *PLoS One* **8**(6), e65329 (2013).
- van Loenhout, M. T. J., Kerssemakers, J. W. J., De Vlamincq, I., and Dekker, C., "Non-bias-limited tracking of spherical particles, enabling nanometer resolution at low magnification," *Biophys. J.* **102**(10), 2362–2371 (2012b).
- Van Oijen, A. M., Blainey, P. C., Crampton, D. J., Richardson, C. C., Ellenberger, T., and Xie, X. S., "Single-molecule kinetics of lambda exonuclease reveal base dependence and dynamic disorder," *Science (N. Y.)* **301**(5637), 1235–1238 (2003).
- Vanzi, F., Vladimirov, S., Knudsen, C. R., Goldman, Y. E., and Cooperman, B. S., "Protein synthesis by single ribosomes," *RNA* **9**, 1174–1179 (2003).
- Wirtz, D., "Particle-tracking microrheology of living cells: principles and applications," *Annu. Rev. Biophys.* **38**, 301–326 (2009).
- Xiao, B., Zhang, H., Johnson, R. C., and Marko, J. F., "Force-driven unbinding of proteins HU and Fis from DNA quantified using a thermodynamic Maxwell relation," *Nucl. Acids Res.* **39**(13), 5568–5577 (2011).