IUCrJ
ISSN 2052-2525
CRYO | EM

A Bayesian approach to beam-induced motion correction in cryo-EM single-particle analysis

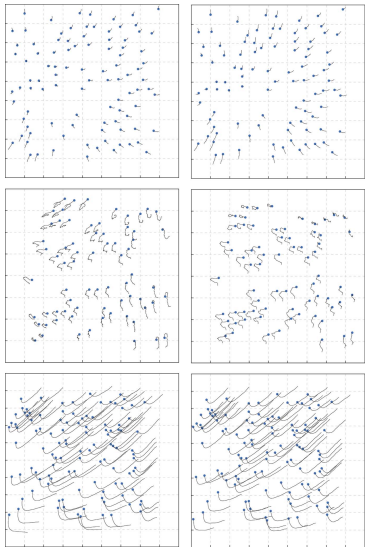
Jasenko Zivanov,* Takanori Nakane and Sjors H. W. Scheres*

Medical Research Council Laboratory of Molecular Biology, Cambridge CB2 0QH, England. *Correspondence e-mail: jzivanov@mrc-lmb.cam.ac.uk, scheres@mrc-lmb.cam.ac.uk

Received 6 August 2018
Accepted 16 October 2018

Edited by F. Sun, Chinese Academy of Sciences, China

Keywords: Bayesian particle polishing; beam-induced motion correction; cryo-EM; single-particle analysis; electron cryo-microscopy.



OPEN ACCESS

A new method to estimate the trajectories of particle motion and the amount of cumulative beam damage in electron cryo-microscopy (cryo-EM) single-particle analysis is presented. The motion within the sample is modelled through the use of Gaussian process regression. This allows a prior likelihood that favours spatially and temporally smooth motion to be associated with each hypothetical set of particle trajectories without imposing hard constraints. This formulation enables the *a posteriori* likelihood of a set of particle trajectories to be expressed as a product of that prior likelihood and an observation likelihood given by the data, and this *a posteriori* likelihood to then be maximized. Since the smoothness prior requires three parameters that describe the statistics of the observed motion, an efficient stochastic method to estimate these parameters is also proposed. Finally, a practical algorithm is proposed that estimates the average amount of cumulative radiation damage as a function of radiation dose and spatial frequency, and then fits relative *B* factors to that damage in a robust way. The method is evaluated on three publicly available data sets, and its usefulness is illustrated by comparison with state-of-the-art methods and previously published results. The new method has been implemented as Bayesian polishing in *RELION*-3, where it replaces the existing particle-polishing method, as it outperforms the latter in all tests conducted.

1. Introduction

Recent advances in electron-detector technology have allowed cryo-EM single-particle analysis to uncover the structures of many biological macromolecules to resolutions sufficient for *de novo* atomic modelling. The primary impediment to high-resolution reconstruction is the radiation damage that is inflicted on the molecules when they are exposed to an electron beam. This requires low-dose imaging, and hence reconstructions from very noisy images. In addition, exposure to the electron beam leads to motion in the sample, which destroys information, particularly at high spatial frequencies.

Because the new detectors allow multi-frame movies to be captured during exposure of the sample, it is possible to estimate and correct for beam-induced motion. This requires sufficient signal in the individual movie frames, which is challenging as each frame only contains a fraction of the total electron dose, resulting in even lower signal-to-noise ratios. The earliest approaches to beam-induced motion correction were performed in *FREALIGN* (Brilot *et al.*, 2012; Campbell *et al.*, 2012) and *RELION* (Bai *et al.*, 2013), and estimated particle positions and orientations independently in each movie frame and for each particle. Both programs averaged the signal over multiple adjacent frames to boost the low signal-to-noise ratios. Still, these approaches were only applicable to relatively large (>1 MDa in molecular weight) particles (*i.e.* molecules or molecular complexes). These early studies revealed correlations between the direction and extent

of motion of particles that are in close proximity to each other. In this paper, we will refer to this property as the *spatial smoothness* of motion.

The approach in Bai *et al.* (2013) was subsequently extended to cover smaller molecules. This was possible by (i) still performing template matching on averages over multiple adjacent frames, (ii) fitting a linear path of constant velocity through the unreliably detected positions and (iii) averaging these constant-velocity vectors over local areas of the micrograph. This means that consistency with the observations and the (absolute) temporal and (partial) spatial smoothness of the trajectories were imposed one after the other. This algorithm, together with the radiation-dose weighting scheme described below, was termed particle polishing (Scheres, 2014) and was implemented as the method of choice for beam-induced motion correction in the *RELION* package (Scheres, 2012).

In the meantime, a second class of motion-estimation algorithms have been developed that do not rely on the availability of a three-dimensional reference structure, and which therefore can be applied much earlier in the image-processing workflow. Instead of comparing individual particles with their reference projections, these algorithms estimate the motion entirely from the frame sequence itself by cross-correlating individual movie frames or regions within them. Two advantages of reference-free methods are that they are not susceptible to errors in the references, for example unresolved structural heterogeneity, and that sources of structural noise that move together with the particles, for example high-contrast contamination, may be used as signal for motion estimation. An important disadvantage of reference-free methods, and the main motivation for using a reference in this paper, is the lower signal-to-noise ratio in the cross-correlation functions between noisy movie frames compared with the cross-correlation with a high-resolution reference projection. In addition, reference-free methods are susceptible to sources of structured noise on the detector (for example dead or hot pixels, or imperfect gain normalizations), which favour a zero velocity. Such noise is typically not present in the reference, as it is reconstructed from many images in different orientations.

Two of the early reference-free methods, *MotionCorr* (Li *et al.*, 2013) and *Unblur* (Grant & Grigorieff, 2015), relaxed the spatial smoothness assumption, allowing nonlinear trajectories. While *MotionCorr* allowed completely free motion over time, it required discrete regions of the image to move as rigid blocks. *Unblur* imposed a certain amount of temporal smoothness on the motion and required the entire image to move as a rigid block. The method of Abrishami *et al.* (2015) was based on an iterative version of the Lucas–Kanade optical flow algorithm (Lucas & Kanade, 1981) and abandoned the idea of rigid regions in favour of a model that allows spatially smooth deformations of the image. Later, a more robust noise model was proposed in *Zorro* (McLeod *et al.*, 2017), which required uniform movement of the entire micrograph, and a variant, *SubZorro*, that worked on rigid regions.

An early method to formulate motion estimation as a minimization of a cost function in order to simultaneously satisfy consistency with the observations and temporal smoothness was *alignparts-lmbfgs* (Rubinstein & Brubaker, 2015). It estimated the motion of each particle separately, so that spatial smoothness of the motion was enforced only after the fact, by forming local averages over trajectories of neighbouring particles. Although *alignparts-lmbfgs* works on individual particles, the program does not use reference projections, but minimizes a weighted phase difference between the Fourier components of individual movie frames of boxed-out particles.

A reference-free method that is very popular today is *MotionCor*2 (Zheng *et al.*, 2017). This program enforces neither spatial nor temporal smoothness absolutely. Instead of working on individual particles, it splits the micrograph into tiles and fits the motion of each tile to a global polynomial function of time and space. This is performed by picking independent, most likely positions of each block and then fitting the coefficients of the polynomial to these discrete positions. We will compare our new method with *MotionCor*2 in Section 3.

Unlike particle motion, radiation damage cannot be corrected for explicitly. Nevertheless, the deleterious effects of radiation damage on the reconstruction can be reduced by down-weighting the contribution of the higher spatial frequencies in the later movie frames. This is because radiation damage affects the signal at high spatial frequencies faster than the signal at low spatial frequencies (Hayward & Glaeser, 1979). For this reason, it was proposed to discard the later movie frames for high-resolution reconstruction (Li *et al.*, 2013). The particle-polishing program in *RELION* (Scheres, 2014) would then extend this to a continuous radiation-damage weighting scheme. This approach used a relative *B*-factor model (based on the temperature factors that are commonly used in X-ray crystallography) to describe the signal fall-off with resolution. Later, building on the idea of a critical exposure by Unwin & Henderson (1975) and early calculations and measurements of this exposure by Hayward & Glaeser (1979) and Baker & Rubinstein (2010), Grant and Grigorieff measured a more precise exponential damage model from a reconstruction of a rotavirus capsid (Grant & Grigorieff, 2015). The latter is currently in use in many programs.

In this paper, we describe a new method, which we have termed Bayesian polishing and which has been implemented in the *RELION* package. This method still uses the original *B*-factor model for the relative weighting of different spatial frequencies in different movie frames, although we do propose a new method to estimate the *B* factors. We chose the *B*-factor model because, as opposed to the exponential model of Grant and Grigorieff, it allows us to model both radiation damage and any residual motion that is not corrected for. However, as the *B* factors can only be determined once the motion has been estimated, we do use the exponential model during the initial motion-estimation step.

The two main disadvantages of the motion-estimation process in the original particle-polishing algorithm in *RELION* that prompted these developments were the absolute

temporal smoothness assumption and the feed-forward nature of the fitting process: a linear path that best fits the estimated noisy positions might not be the linear path that leads to the greatest overall consistency with the observed data. In other words, the per-frame maxima are picked prematurely. This is illustrated in Fig. 1. The same is also true for the spatially smooth velocity field that results from the averaging of multiple such linear trajectories. The motion-estimation method that we propose in this paper overcomes both of these disadvantages.

2. Materials and methods

In the following, we will discuss the different components of our proposed Bayesian polishing approach. We will begin by describing the motion model and the motion-estimation process in Section 2.1. After that, we will explain how the parameters for our prior, *i.e.* for the statistics of motion, are determined in Section 2.2. Although these have to be known in order to estimate the most likely motion, we chose to describe their determination afterwards, since its understanding requires knowledge of the actual motion model. We then describe the process of measuring the relative B factors and recombining the frames in Section 2.3, and we conclude this section with a description of our evaluation process in Section 2.5.

2.1. Motion estimation

2.1.1. Outline. The central idea behind our motion estimation consists of finding a set of particle trajectories in each micrograph that maximize the *a posteriori* probability given the observations. Note that we assume that a reference map, the viewing angles and defoci of the particles, and the parameters of the microscope are known by this point.

Formally, we express the particle trajectories as a set of positions $s_{p,f} \in \mathbb{R}^2$ for each particle $p \in \{1 \dots P\}$ and frame $f \in \{1 \dots F\}$. The corresponding per-frame particle displacements are denoted by $v_{p,f} = s_{p,f+1} - s_{p,f}$ for $f \in \{1 \dots F-1\}$. We will

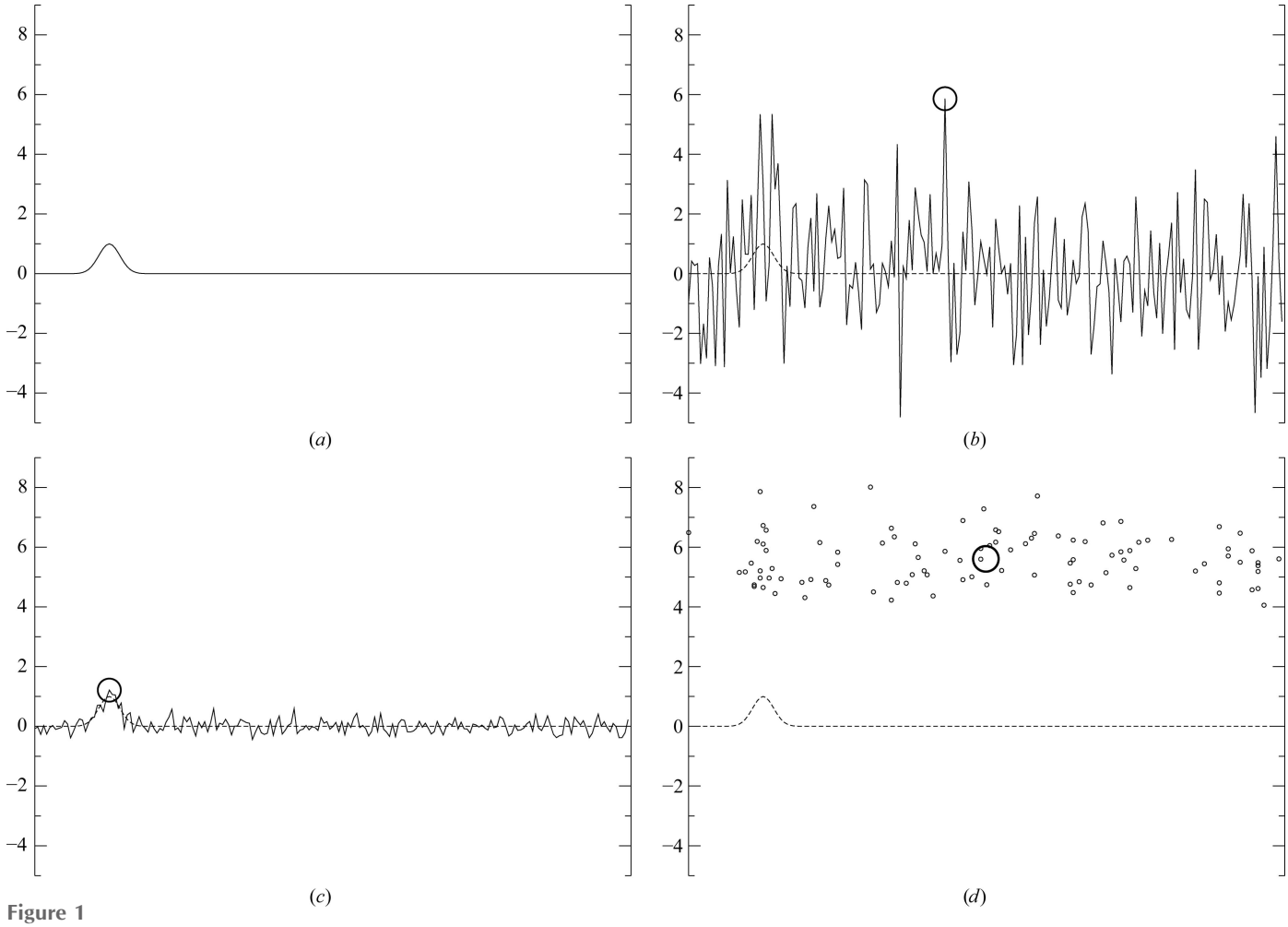


Figure 1
A simulated example illustrating the issue of premature maximum picking. (*a*) A Gaussian representing the cross-correlation between the reference and the observation of a particle. (*b*) This cross-correlation distorted by Gaussian white noise of realistic intensity for the cross-correlation between a noise-free reference and one observed frame of one single particle ($\sigma = 2$; the circle indicates the maximum). (*c*) The average over 100 such noisy functions and its maximum. (*d*) The maxima of those 100 noisy functions (small circles) and their average (large circle). Note how the average of the noisy maxima (*d*) is much further from the true maximum than the maximum of the average (*c*). Our proposed method avoids picking individual maxima of noisy functions; it instead aims to maximize the cross-correlations of all particles and the prior smoothness assumptions simultaneously.

refer to $v_{p,f}$ as per-frame velocities in the following, since they are equal to the mean velocities between the two frames if time is measured in units of frames.

Let $s = \{s_{p,f} | p \in \{1 \dots P\}, f \in \{1 \dots F\}\}$ denote the set of all particle trajectories in a micrograph. The *a posteriori* probability $P_{\mathrm{AP}}(s|\mathrm{obs})$ of these trajectories given the observations obs is then given by Bayes' law,

$$P_{\mathrm{AP}}(s|\mathrm{obs}) \propto P_{\mathrm{prior}}(s) P_{\mathrm{obs}}(\mathrm{obs}|s), \tag{1}$$

where the term $P_{\mathrm{prior}}(s)$ describes the prior probability of this set of trajectories and is described by the statistics of motion, while $P_{\mathrm{obs}}(s|\mathrm{obs})$ describes the probability of making the observations obs given these trajectories.

We will first describe our motion model that gives rise to $P_{\mathrm{prior}}(s)$ in Section 2.1.2 and then the observation model that defines $P_{\mathrm{obs}}(s|\mathrm{obs})$ in Section 2.1.3.

2.1.2. The motion model. We model particle motion using Gaussian process (GP) regression. GPs have been in use by the machine-learning community for decades (Rasmussen, 2004), and they have found applications in the fields of computer vision (Lüthi *et al.*, 2018), computer graphics (Wang *et al.*, 2008) and robotics (Nguyen-Tuong *et al.*, 2009).

Formally, a GP is defined as a distribution over the space of functions $f(x)$ such that for every finite selection of x_i the corresponding $f(x_i)$ follow a multivariate normal distribution. A GP can therefore be thought of as an extension of the concept of a multivatiate normal distribution to cover the (infinitely dimensional) Hilbert space of functions. Although the term 'process' suggests x to be a one-dimensional time variable, a GP can in fact be defined over any domain. In our case, we use the particle positions in the micrograph (*i.e.* a two-dimensional plane) as that domain, while the function $f(x)$ will be used to describe the velocity vectors of particles.

In its most general form, a GP is defined by a mean $\mu(x)$ and a covariance function $C(x_1, x_2)$. In our specific case, we will assume the mean velocity to be zero, and we will work with *homogeneous* GPs, where the covariance between two points x_1 and x_2 depends only on their distance $d = |x_2 - x_1|$. We will use the GP to enforce spatial smoothness of the motion vectors. This means that the covariance $C(d)$ between two velocity vectors will be greater for particles that are closer together.

Specifically, the covariance between the velocities of two particles p and q is modelled by the exponential kernel,

$$C(v_p, v_q) = \sigma_V^2 \exp(-|s_p - s_q|/\sigma_D), \tag{2}$$

where σ_V describes the expected amount of motion, while σ_D describes its spatial correlation length. We use a single value of σ_V and of σ_D for all micrographs in the data set. Since the overall beam-induced motion of the particles is generally far smaller than their mutual distance (a few ångströms versus hundreds of ångströms), we chose to compute the covariance based on the initial particle positions alone: this is why the subscript f is missing in (2).

We can write the covariances of all particles $C(v_p, v_q)$ into a $P \times P$ covariance matrix Σ_V, which then describes the per-frame multivariate normal distribution of all velocity vectors $v_{p,f}$. As is common in GP regression, we perform a singular-value decomposition on σ_V to obtain a more practical parametrization for our problem:

$$\Sigma_V = U \Lambda W^T. \tag{3}$$

This allows us to define a set of basis vectors $b_i = \lambda_i^{1/2} w_i$, where $\lambda_i \in \mathbb{R}$ is the ith singular value and $w_i \in \mathbb{R}^P$ is its associated singular vector (*i.e.* column of W or row of W^T). For each frame, the x and y components of the velocity vectors v_p of all particles p can now be expressed as linear combinations of b_i with a set of P coefficients c_i:

$$v^{(x)} = \sum_i c_i^{(x)} b_i, \quad v^{(y)} = \sum_i c_i^{(y)} b_i. \tag{4}$$

In this parametrization, the per-frame joint likelihood of this set of velocities has a particularly simple form:

$$P_f(c) = (2\pi)^{-P} \exp\left(-\frac{1}{2} \sum_{i=1}^{P} |c_i|^2\right). \tag{5}$$

For this reason, we use $F - 1$ sets of coefficients $c_{i,f}$ as the unknowns in our problem. Since the c_i only describe the velocities, they only determine the positions $s_{p,f}$ up to a per-particle offset. The complete set of unknowns for a micrograph therefore also has to include the initial positions $s_{p,0}$. The initial positions have no effect on the prior probability, however.

Formally, for $c_{i,f} = [c_{i,f}^{(x)}, c_{i,f}^{(y)}]^T$, the positions are then given as a function of all $c_{i,f}$ by

$$s_{p,f} = s_{p,0} + \sum_{f'=1}^{f-1} v_{p,f'} \tag{6}$$

$$= s_{p,0} + \sum_{f'=1}^{f-1} \sum_{i=1}^{P} b_i c_{i,f'}. \tag{7}$$

So far, we have only modelled the spatial smoothness of the motion. To impose temporal smoothness, we define the complete prior probability as

$$P_{\mathrm{prior}}(c) = P_{\mathrm{space}}(c) P_{\mathrm{time}}(c), \tag{8}$$

with

$$P_{\mathrm{space}}(c) = \prod_{f=1}^{F} P_f(c), \tag{9}$$

$$P_{\mathrm{time}}(c) = \prod_{f=2}^{F-1} \prod_{p=1}^{P} \frac{1}{2\pi\sigma_A^2} \exp\left(-\frac{1}{2} \frac{|v_{p,f} - v_{p,f-1}|^2}{\sigma_A^2}\right), \tag{10}$$

where σ_A is the third and final motion parameter that describes the average acceleration of a particle during a frame, *i.e.* the standard deviation of the change in velocity between two consecutive frames. Again, we use a single value of σ_A for all micrographs in the data set.

The temporal smoothness term P_{time} corresponds to that proposed by Rubinstein & Brubaker (2015) for individual particles. From the orthogonality of the basis b_i, it follows that in our parametrization

$$P_{\rm time}(c) = \prod_{f=2}^{F-1}\prod_{i=1}^{P}\frac{1}{2\pi\sigma_A^2}\exp\left(-\frac{1}{2}\frac{\lambda_i|c_{i,f}-c_{i,f-1}|^2}{\sigma_A^2}\right). \quad (11)$$

The motion model could in principle be made more precise, for example by adding parameters to describe the observation that particles tend to move faster in early movie frames. However, the increased dimensionality would lead to a significant increase in the computational cost of the parameter hyper-optimization scheme described in Section 2.2, rendering the approach less practical.

2.1.3. The observation model. In the following, we will derive the observation likelihood $P_{\rm obs}({\rm obs}|x)$. Since we assume a three-dimensional reference map, the viewing angles and the microscope parameters to be known, we can predict the appearance of a particle using the reference map (Scheres, 2012). This is performed by integrating the reference map along the viewing direction, which can be accomplished efficiently by extracting a central slice in Fourier space and then convolving the resulting image with the known contrast transfer function (CTF).

To maintain the nomenclature from previous *RELION* papers, we denote pixel $j \in \mathbb{N}^2$ of frame f of the experimental image of particle p by $X_{p,f}(j)$ and the same pixel in the prediction by $V_{p,f}(j)$. The spectral noise power is measured from all X in a micrograph, and both X and V are filtered in order to whiten the image noise (*i.e.* decorrelate the noise between the pixels) and to scale it to unit variance. In addition, we use the exponential damage model (Grant & Grigorieff, 2015) to suppress the high frequencies in the later frames in V.

By assuming that the noise in the pixels is Gaussian and independent, it follows that

$$\begin{aligned} P_{\rm obs}({\rm obs}|s) &\propto \prod_{p,f,j}\exp\left\{-\frac{1}{2}[(X_{p,f}(j)-V_{p,f}(j-s_{p,f})]^2\right\} \\ &= \exp\left\{-\frac{1}{2}\sum_{p,f,j}[X_{p,f}(j)-V_{p,f}(j-s_{p,f})]^2\right\}. \end{aligned} \quad (12)$$

Since the prediction V is zero outside the molecule, the image area over which this sum is evaluated only influences the scale of $P_{\rm obs}$ and not its shape. In practice, we cut out a square from the micrograph that contains the molecule (including a certain amount of padding around it to account for its motion) and evaluate $P_{\rm obs}$ on that square.

In order to evaluate $P_{\rm obs}({\rm obs}|s)$ efficiently for different hypothetical particle positions s, we use the following identity:

$$\sum_j [X_{p,f}(j)-V_{p,f}(j-s_{p,f})]^2 \quad (13)$$

$$= -2\sum_j X_{p,f}(j)V_{p,f}(j-s_{p,f}) + K \quad (14)$$

$$= -2{\rm CC}_{p,f}(s_{p,f}) + K, \quad (15)$$

where ${\rm CC}_{p,f}$ denotes the cross-correlation between $X_{p,f}$ and $V_{p,f}$, which is computed for a Cartesian grid of integral s simultaneously *via* a convolution in Fourier space. The constant offset K merely scales the resulting probability $P_{\rm obs}$, so it does not alter the location of the maximum of $P_{\rm AP} = P_{\rm prior}P_{\rm obs}$. We can thus define

$$P'_{\rm obs}({\rm obs}|s) = \exp\left[\sum_p^P\sum_f^F {\rm CC}_{p,f}(s_{p,f})\right]. \quad (16)$$

To determine the values of ${\rm CC}_{p,f}$ at fractional coordinates, we apply cubic interpolation. This ensures a continuous gradient.

2.1.4. Optimization. To avoid numerical difficulties, we maximize $P_{\rm AP}(s|{\rm obs})$ by instead minimizing its doubled negative log, $E_{\rm AP} = -2\log(P_{\rm AP})$. The doubling serves to simplify the terms. All of the products in $P_{\rm AP}$ become sums in $E_{\rm AP}$, yielding

$$\begin{aligned} E_{\rm AP}(s|{\rm obs}) &= E_{\rm prior}(x) + E'_{\rm obs}({\rm obs}|s) \\ &= E_{\rm space}(s) + E_{\rm time}(s) + E'_{\rm obs}({\rm obs}|s), \end{aligned} \quad (17)$$

where the terms $E_{\rm space}$, $E_{\rm time}$ and $E'_{\rm obs}$ are defined analogously. Inserting the terms defined in Sections 2.1.2 and 2.1.3 yields

$$\begin{aligned} E_{\rm AP}(c, s_0|{\rm obs}) = &\sum_{f=1}^{F-1}\sum_{i=1}^{P}|c_{i,f}|^2 \\ &+ \frac{1}{\sigma_A^2}\sum_{f=2}^{F-1}\sum_{i=1}^{P}\lambda_i|c_{i,f}-c_{i,f-1}|^2 \\ &- 2\sum_{f=1}^{F}\sum_{p=1}^{P}{\rm CC}_{p,f}[s_{p,f}(c, s_0)]. \end{aligned} \quad (18)$$

The expression in (18) is differentiated with respect to the coefficients $c_{i,f}$ and initial positions $s_{p,0}$ for all i and f, and the combination that minimizes $E_{\rm AP}(c, s_0|{\rm obs})$ is determined using the *L-BFGS* algorithm (Liu & Nocedal, 1989). In order to avoid overfitting, all particles are aligned against a reference computed from their own independently refined half-set (Scheres & Chen, 2012).

2.2. Parameter estimation for the statistics of motion

The estimation procedure described in Section 2.1 requires three parameters (σ_V, σ_D and σ_A) for the prior that encapsulate the statistics of particle motion. Since the precise positions of the particles can never be observed directly, measuring these statistics requires performing a process of hyper-optimization, *i.e.* optimizing motion parameters that produce the best motion estimates. This renders the entire approach an empirical Bayesian one. The simplest solution would be to perform a complete motion estimation for each hypothetical triplet of motion parameters. As the motion estimation usually takes multiple hours on a nontrivial data set, this would become prohibitive for a three-dimensional grid of parameters.

Instead, we estimate the optimal parameters using the following iterative procedure. Firstly, we select a representative random subset $\mathcal{M}$ of micrographs that contain at least a pre-defined minimal number of particles (25 000 in our experiments). We then perform the following three steps iteratively.

(i) Choose a hypothetical parameter triplet σ_V, σ_D and σ_A.

(ii) Align all micrographs in $\mathcal{M}$ using these parameters.

(iii) Evaluate the parameters.

The iterations are performed using the Nelder–Mead uphill simplex algorithm (Nelder & Mead, 1965), which does not rely on the function over which it optimizes being differentiable.

In order to evaluate a parameter triplet, we perform the alignment only on a limited range of spatial frequencies (the *alignment circle*, $\{k \in \mathbb{N}^2, |k| < T\}$). The remainder of frequencies, the *evaluation ring* $\{|k| > T\}$, is used to evaluate this alignment. To avoid overfitting, *i.e.* to retain a strict separation of the two half-sets, we perform the alignment against a reference obtained from the half-set to which the respective particle belongs. For the evaluation, we use a reference obtained from the opposite half-set to avoid the particle 'finding itself' (Grant & Grigorieff, 2015) in the reference. Note that the latter does not incur any risk of overfitting, since the alignment is already known by the time it is evaluated, and the small number of parameters (*i.e.* three values) leaves no room for overfitting.

The partition of frequency space into an alignment circle and an evaluation ring is necessary: if the alignment and the evaluation were to be performed on the same frequencies k, then a weaker prior would always produce a greater correlation than a stronger one. Note that this would happen in spite of splitting of the particles into independent half-sets, because an insufficiently regularized alignment will align the noise *in the images* with the signal in the reference, while the two references share the same signal in the frequency range in which they are meaningful.

The evaluation itself is performed by measuring what we propose to call the thick-cylinder correlation [$\mathrm{TCC}(x) \in \mathbb{R}$] between the aligned images and the reference,

$$\mathrm{TCC}(s) = \frac{\sum\limits_{\substack{m,p,f,k \\ |k|>T}} \mathrm{Re}\big[\hat{X}_{m,p,f}(k)\hat{V}^{*}_{m,p,f,s}(k)\big]}{\left\{\left[\sum\limits_{\substack{m,p,f,k \\ |k|>T}} |\hat{X}_{m,p,f}(k)|^2\right]\left[\sum\limits_{\substack{m,p,f,k \\ |k|>T}} |\hat{V}_{m,p,f,s}(k)|^2\right]\right\}^{1/2}}, \tag{19}$$

where $\hat{X}_{m,p,f}(k)$ and $\hat{V}_{m,p,f,s}(k) \in \mathbb{C}$ are the Fourier-space amplitudes of frequency k of the observed image and the prediction, respectively. The indices denote frame f of particle p in micrograph $m \in \mathcal{M}$. The prediction $\hat{V}_{m,p,f,s}(k)$ has been shifted according to the estimated $s_{m,p,f}$, *i.e.* $\hat{V}_{m,p,f,s}(k) = \exp(-2\pi i\langle s, k\rangle)\hat{V}_{m,p,f}(k)$. The asterisk indicates complex conjugation and $\langle\rangle$ indicates a two-dimensional scalar product.

2.3. Damage weighting

Once the frames of a movie have been aligned, we compute a filtered average over them that aims to maximize the signal-to-noise ratio in each frequency. In the original particle-polishing method (Scheres, 2014), the proposed image-recombination approach was based on relative B factors. We use the same approach here, but we propose a more practical and more robust means of estimating the relative B factors.

The original technique required the computation of two full three-dimensional reconstructions from particle images of every frame, one for each independently refined half-set. In a typical data set comprising 40 frames, this would amount to computing 80 individual reconstructions, which requires days of CPU time. The two corresponding reconstructions would then be used to determine the Fourier shell correlation (FSC) in order to estimate the spectral signal-to-noise ratio (SSNR) of the three-dimensional reconstruction.

Our new method is more practical in that it avoids the computation of these three-dimensional reconstructions. Instead, we directly measure the correlation between the aligned frames and the reference as soon as the particles in a movie have been aligned. This is performed by evaluating what we have termed the Fourier-cylinder correlation $\mathrm{FCC}(f, \kappa)$ for each frame index f and Fourier shell κ. This amounts to correlating the set of Fourier rings of radius κ against the reference for all particles simultaneously, hence the term Fourier cylinder.

Formally, the FCC is defined as

$$\mathrm{FCC}(f, \kappa) = \frac{\sum\limits_{\substack{m,p,k \\ |k-\kappa|<0.5}} \mathrm{Re}\big[\hat{X}_{m,p,f}(k)\hat{V}^{*}_{m,p,f,s}(k)\big]}{\left\{\left[\sum\limits_{\substack{m,p,k \\ |k-\kappa|<0.5}} |\hat{X}_{m,p,f}(k)|^2\right]\left[\sum\limits_{\substack{m,p,k \\ |k-\kappa|<0.5}} |\hat{V}_{m,p,f,s}(k)|^2\right]\right\}^{1/2}}, \tag{20}$$

for k and κ given in pixels. It can be evaluated by iterating over the data set only once, updating the three sums in (20) for each particle in each micrograph.

The FCC allows us to estimate the SSNR of the aligned images themselves, not of the three-dimensional reconstructions. The fact that these SSNR values are different is of no concern, as we are only interested in their relative change as a function of frame index f. Since the value of each voxel of a three-dimensional reconstruction is an average over the pixels from many images, the relative change in the SNR of that voxel over time is the same as for the corresponding pixels.

Once the FCC has been determined, we proceed to fit the relative B factors. This is performed by finding a B_f and $C_f \in \mathbb{R}$ for each frame f and a $D_\kappa \in \mathbb{R}$ for each frequency ring κ that minimize

$$\sum_{f,\kappa}\big[\mathrm{FCC}(f, \kappa) - D_\kappa \exp(C_f + 4B_f\kappa^2)\big]^2. \tag{21}$$

Here, the coefficients D_κ are nuisance parameters that encapsulate the amount of signal in the reference in each frequency band κ. This allows the B_f and C_f factors to only express the variation in signal over the frame index f. The D_κ are higher for frequencies that are more prominent in the structure (such as those of α-helices) and they are zero beyond the resolution of the current reference map. In the previous particle-polishing formulation, the D_κ correspond to a Gaussian over κ given by the average B factor. The coefficients B_f and C_f maintain the same meaning as in the previous formulation, *i.e.* the change in high-frequency information and overall contrast over time, respectively. An illustration of the model is shown in Fig. 2.

The factors B_f, C_f and D_κ are estimated iteratively by first finding the optimal D_κ for each κ given the current B_f and C_f, and then the optimal B_f and C_f given the current D_κ. The optimal D_κ can be determined linearly, while the B_f and C_f are found through a recursive one-dimensional search over B_f; the optimal C_f for a given B_f can also be determined linearly. In our implementation, the entire procedure is run for five iterations, and it typically takes less than a second to complete.

The final weight of each Fourier-space pixel is then given by

$$w_{\kappa,f} = \frac{\exp(C_f + 4B_f\kappa^2)}{\sum_{f'} \exp(C_{f'} + 4B_{f'}\kappa^2)}. \quad (22)$$

2.4. Implementation

The motion-estimation algorithm has been implemented using MPI, allowing it to align multiple micrographs in parallel on different computers. The processes that are run on each of these computers are further parallelized using OpenMP, which allows the user to exploit all of the available CPU cores on all of the available computers at the same time. Although it is also possible to align multiple micrographs on the same computer simultaneously by running multiple MPI processes there, we discourage this since it requires each of those processes to maintain its own data in memory. If the multiple CPU cores of the same computer are instead allowed to cooperate in aligning the same micrograph, then the memory is only taken up once.

The memory footprint of the motion-estimation algorithm consists primarily of the two three-dimensional reference maps (one for each independently refined half-set) and the pixels of the micrograph that is currently being processed. In most cases, this requires approximately 20 GB of memory for each MPI process.

Owing to its iterative nature, the parameter hyper-optimization algorithm does not allow MPI parallelization. Furthermore, in order to avoid loading the subset of micrographs from disk in each iteration, all of the necessary data are stored in memory. For this reason, the memory footprint of the parameter hyper-optimization algorithm could exceed 60 GB for the 25 000 particles used in our experiments. Although a smaller number of particles does reduce this footprint, it also renders the estimated optimal parameters less accurate.

Finally, in order to save disk space, the entire motion-estimation pipeline supports micrographs stored as compressed TIFF images. Such images contain the integral numbers of counted electrons for each pixel, which enables very efficient compression, usually by a factor of about 30. Owing to the integral pixel values, an external gain reference has to be provided if such TIFF images are being used.

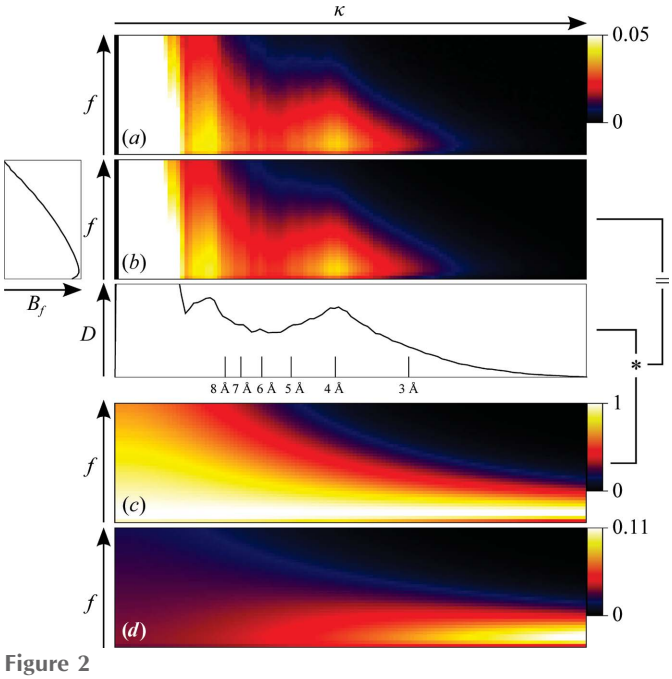


Figure 2
An illustration of FCC-based *B*-factor fitting using the β-galactosidase data set as an example (best viewed in colour). (*a*) The FCC computed using (20) as a function of spatial frequency κ and frame *f*. (*b*) A fit of B_f, C_f and D_κ according to (21) with plots of B_f and D_κ shown in relation. (*c*) The same fit with all D_κ set to 1 (*i.e.* the numerator of equation 22). (*d*) The normalized weights $w_{\kappa,f}$ as given by (22). The asterisk indicates a multiplication.

2.5. Experimental design

We evaluated Bayesian polishing on three publicly available data sets that cover a range of particle sizes: the *Plasmodium falciparum* cytoplasmic ribosome (EMPIAR 10028), *Escherichia coli* β-galactosidase (EMPIAR 10061) and human γ-secretase (EMPIAR 10194). For all three cases our group has previously published structures calculated using the original particle-polishing approach (Wong *et al.*, 2014; Kimanius *et al.*, 2016; Bai *et al.*, 2015). We used the same particles and masks for both polishing and the final high-resolution refinement as were used in those papers. Further information on these data sets is shown in Table 1.

The experiments were set up as follows. Firstly, the input movies were aligned and dose-filtered using *MotionCor*2 (Zheng *et al.*, 2017). From these aligned micrographs, particles were extracted and an initial reference reconstruction was computed using the three-dimensional auto-refinement procedure in *RELION* (Scheres, 2012). Using this reference map, the three parameters that describe the statistics of motion (σ_V, σ_D and σ_A) were determined for each data set, and the Bayesian polishing algorithm was run on the original, unaligned micrographs. One set of *B* factors were estimated for an entire data set, assigning one *B*-factor value to each frame index. Using these, a set of motion-corrected and *B*-factor-weighted particle images were computed, called *shiny* particles in *RELION*, which were then used for a second round of three-dimensional auto-refinement to produce a final map.

Since the official UCSF implementation of *MotionCor*2 does not output motion that can be easily interpolated at the positions of the individual particles, we have written our own

Table 1
Properties of the three data sets.

The two entries in the 'No. of particles' column refer to the numbers used during motion estimation and refinement, respectively.

	Mass	Frame dose (e^- Å^{-2})	F	Average defocus (µm)	No. of particles	Pixel size (Å)	Box size (pixels)
Ribosome	3.2 MDa	1.00	16	2.0	105248/105248	1.340	360
β-Galactosidase	464 kDa	1.18	38	1.0	120516/108210	0.637	384
γ-Secretase	140 kDa	2.00	20	1.9	412275/159550	1.400	180

Table 2
Optimal parameter values used for motion estimation.

The values of σ_V and σ_A are normalized by fractional dose (measured in e^- Å^{-2}), so they are given in Å/(e^- Å^{-2}). The values of σ_D are given in Å.

	σ_V	σ_D	σ_A
Ribosome	1.17	28650	1.6
β-Galactosidase	0.66	3300	1.5
γ-Secretase	0.57	10710	3.0

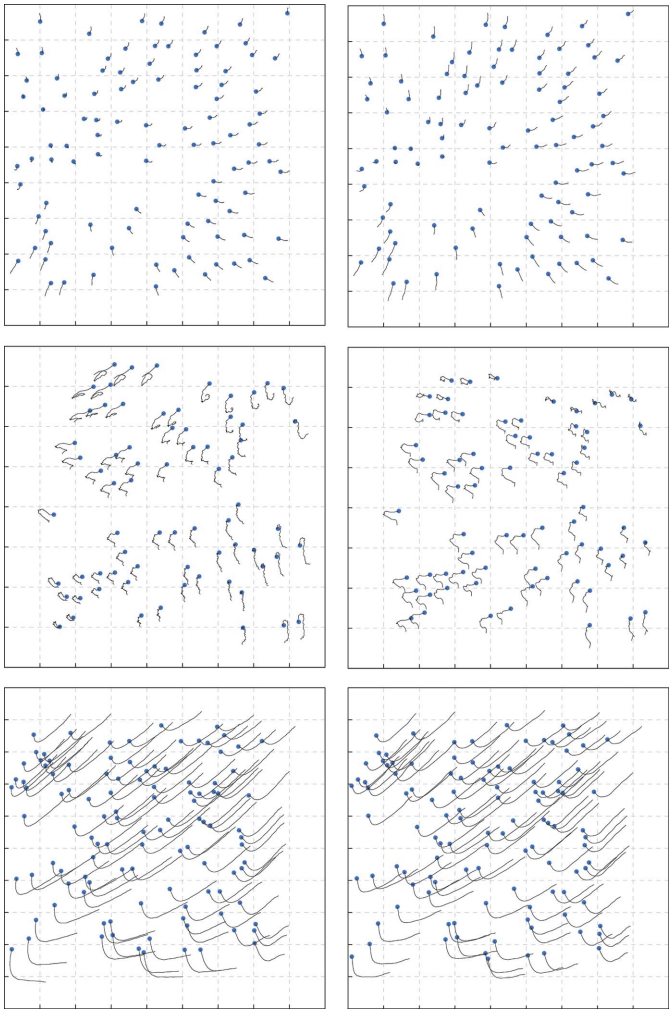


Figure 3
Example trajectories using our own version of *MotionCor*2 (left) and Bayesian polishing (right) for the ribosome (top), β-galactosidase (centre) and γ-secretase (bottom). Particle motion is scaled by a factor of 30. The blue dot indicates the start of the trajectory.

version of *MotionCor*2. The two implementations are not completely identical. Specifically, our version lacks the fallback mechanism of considering larger tiles if the signal in a tile is insufficient, and it only estimates one set of polynomial coefficients for the entire frame range, while the UCSF implementation always estimates two. In Section 3.4, we will show direct comparisons of the FSCs resulting from the two versions to confirm that they give similar resolutions of the final reconstructions.

The particle trajectories for the Bayesian polishing were initialized with the motion estimated by our version of *MotionCor*2. This initialization does not appear to be strictly necessary, however, since in most cases the Bayesian polishing algorithm converged to the same optima if initialized with an unregularized global trajectory. On the β-galactosidase data set, for example, 90% of the final particle positions showed a difference of less than 10^{-4} pixels as a result of initialization.

The resulting maps were compared with those obtained from both versions of *MotionCor*2 and with the previously published results. Since the resolution of the resulting maps is influenced by many different factors beyond particle motion, we assume that the estimated relative B factors reflect the efficacy of motion estimation more reliably than the resolution alone. For this reason, we have also compared the estimated B factors with those obtained from our version of *MotionCor*2 and with the previously published B factors. A B-factor comparison with the UCSF version of *MotionCor*2 is not possible, since the particle trajectories are not readily available.

3. Results and discussion

3.1. Motion parameters

The motion parameters were estimated as described in Section 2.2. The results are shown in Table 2. We used 25 000 randomly selected particles to estimate the parameters. Performing these calculations multiple times showed that the random subset of micrographs that was used to select the 25 000 particles did affect the outcome of the actual values. Specifically, subsets containing micrographs that exhibited a large amount of stage drift would produce a simultaneous increase in the values of σ_V and σ_D, *i.e.* stronger and spatially smoother motion. Nevertheless, the choice among different such parameter triplets did not have a measurable impact on the resolution of the resulting reconstructions (results not shown). We assume that stage drift is also the most important reason behind the difference in parameter values among the three data sets, although other reasons might include the size of the molecule and the thickness of the ice.

3.2. Motion

Using the motion parameters from Table 2, we estimated the motion trajectories for all particles in the three data sets.

These calculations took 128 CPU hours for the ribosome and 778 CPU hours for β-galactosidase on 3.0 GHz Intel Xeon cores, and 1464 CPU hours for γ-secretase on 2.9 GHz Intel Xeon cores. This is comparable to the computational cost of the existing movie-refinement implementation in *RELION*. Examples of trajectories estimated by Bayesian polishing and our implementation of *MotionCor*2 are shown in Fig. 3. A qualitative comparison suggests that they describe the same motion, although they differ in the details. The difference is the most pronounced for β-galactosidase, where the motion statistics correspond to very incoherent motion (*i.e.* a low σ_D). In addition, the trajectories from Bayesian polishing are smoother than the trajectories from *MotionCor*2. This is owing to the fact that the global component of the motion is not regularized in *MotionCor*2. The latter has probably no real impact on the resolution of the reconstruction, since the irregularities are far smaller than one pixel. However, quantitative statements about the quality of motion estimation can only be made once a full reconstruction has been computed. This will be performed in Section 3.4.

3.3. *B* factors

From the particle trajectories estimated by both Bayesian polishing and our implementation of *MotionCor*2, we computed the FCCs as defined in equation (20), and from these the B_f, C_f and D_κ factors. Since the three sums in (20) are updated after the alignment of each micrograph, once all of them have been aligned, the computation of the *B* factors only takes fractions of a second. In the previous particle-polishing implementation, this step would take up multiple days of additional CPU time to calculate two half-set reconstructions for each movie frame. A comparison between the *B* factors obtained by the two methods are shown in Fig. 4. A comparison with the previously published *B* factors is shown on the left-hand side of Fig. 5.

Generally, a set of relative *B* factors can be shifted by a constant offset without altering the resulting pixel weights. Such a shift corresponds to multiplying the D_κ factors by a Gaussian over κ, and it cancels out when the division in (22) is performed. In order to make a meaningful comparison between the *B* factors for motion estimated by Bayesian polishing and *MotionCor*2, we have estimated both sets of *B* factors with the same D_κ factors. This is equivalent to treating the movie frames aligned using Bayesian polishing and those aligned using our implementation of the *MotionCor*2 algorithm as a movie of twice the length. As can be seen in Fig. 4, the *B* factors from Bayesian polishing are better over all frames for all three cases. The average improvement in *B* factor over all frames is 9 Å^2 for the ribosome, 26 Å^2 for β-galactosidase and 15 Å^2 for γ-secretase. These increases suggest that more high-resolution signal is present, and hence that Bayesian polishing models motion more accurately than the *MotionCor*2 algorithm. We will confirm this in the following section.

To confirm that our new technique of estimating *B* factors does not yield systematically different *B* factors from the original method (Scheres, 2014), we also calculated the *B* factors using the original method but with the trajectories from Bayesian polishing for comparison. These plots are shown on the left-hand side of Fig. 5 and they indicate that the new technique produces values that are close to those obtained through the old technique. The similarity between the two curves is especially striking for the ribosome data set (top left in Fig. 5), where the image contrast is the strongest. The greater smoothness of the curve obtained through the new technique in the β-galactosidase plot (centre left in Fig. 5) indicates that the new technique is more robust than the old technique. This is to be expected, since the linear Guinier fit applied by the old technique (Scheres, 2014) has to rely on the frequency range in which the FSC is sufficiently large, and this range can become very small in later frames.

3.4. Resolution

Finally, the gold-standard FSCs are compared with those from the two *MotionCor*2 implementations in Fig. 6 and with

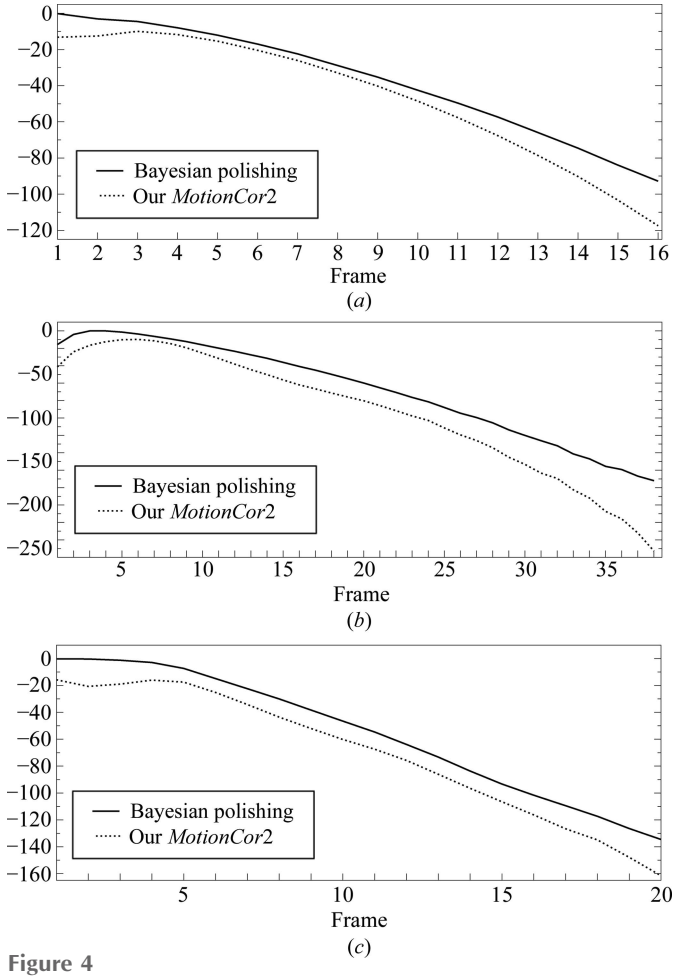


Figure 4
Relative *B* factors for the ribosome (top), β-galactosidase (centre) and γ-secretase (bottom). The two sets of *B* factors share the same D_κ factors, making their relative vertical position meaningful. The observation that the *B* factors from the Bayesian polishing are higher than those from our *MotionCor*2 implementation suggest that Bayesian polishing models the motion more accurately.

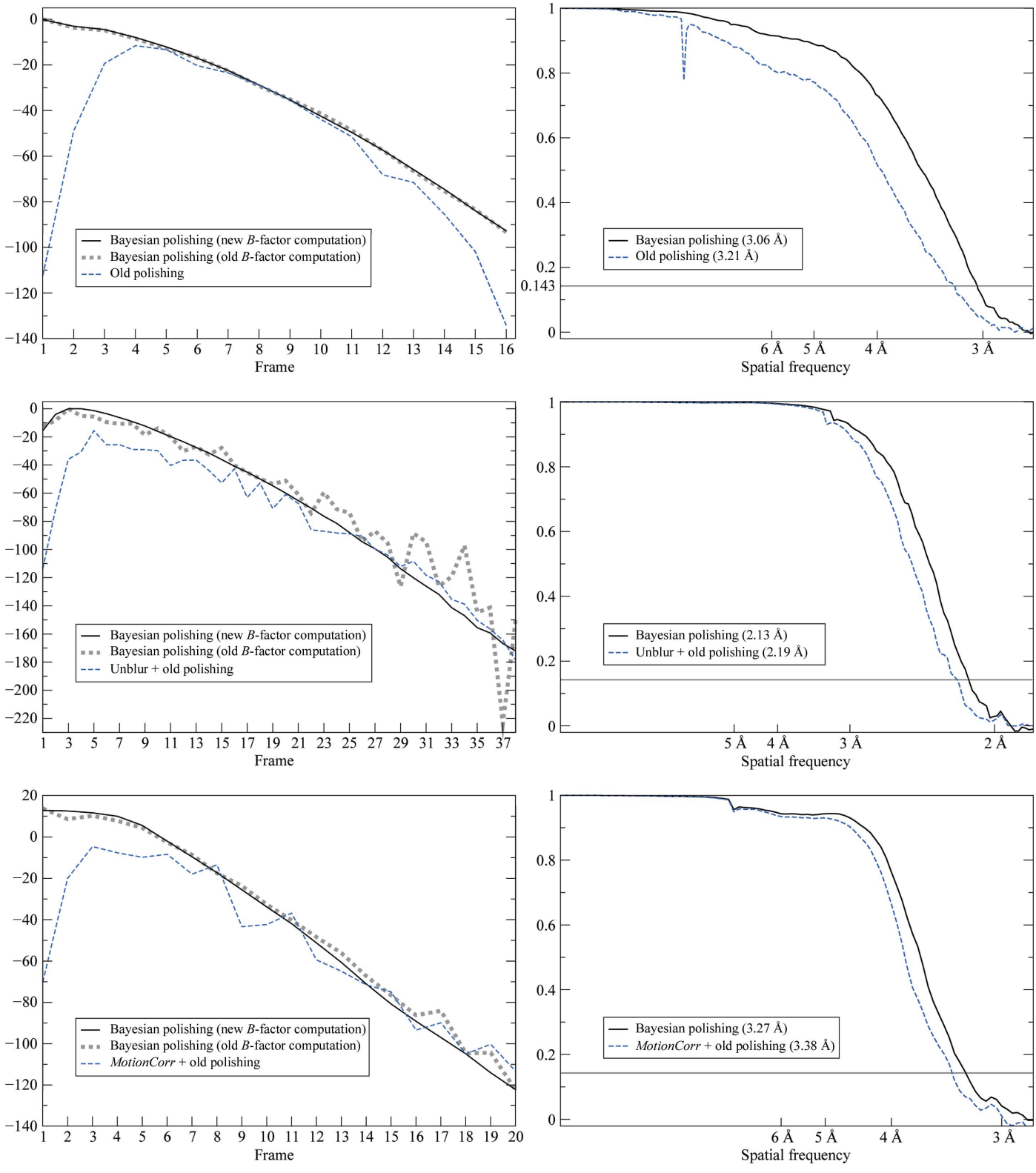


Figure 5
Comparison to previously published results for the ribosome (top), β-galactosidase (centre) and γ-secretase (bottom). Left: relative B factors. Unlike in Fig. 4, the vertical positions of these curves are arbitrary: only their shapes hold any meaning. The continuous black and dotted grey lines correspond to the same motion estimate, but they have been determined using the new and the old B-factor estimation techniques, respectively. Their similarity indicates that the new technique estimates the same B factors as the old technique, albeit in a more robust way. The dashed blue line corresponds to previously published B factors. Note the stark improvement at the beginning of the sequence. Right: FSC curves comparing the new results with the previously published results. Note that the old polishing approach estimated the motion as superimposed over that estimated by another, reference-free method, while Bayesian polishing always works on the raw unaligned micrographs and aims to model the entire motion by itself.

the previously published results on the right-hand side of Fig. 5.

The FSCs were measured under the same solvent mask as had been used in the three previous publications, and the effects of mask-induced correlation were corrected for through phase randomization (Chen *et al.*, 2013) using the post-processing program in *RELION*. To further improve their precision, the resolutions indicated in the figures were measured as the resolutions at which the linearly interpolated FSCs cross the 0.143 threshold.

As can be seen in the FSC plots, Bayesian polishing leads to an increase in resolution over both *MotionCor*2 and the previously published results in all three cases. The increase over *MotionCor*2 is the greatest for the β-galactosidase data set. We assume that this is because this data set extends to higher resolution than the other two data sets, and Bayesian polishing makes more efficient use of the high spatial frequencies by comparing the noisy movie frames with high-resolution reference projections. This assumption is further supported by the fact that β-galactosidase is also the only data set on which traditional polishing applied after *Unblur* produces a better reconstruction than running *MotionCor*2 alone. Compared with our previously published results, the increase in resolution is highest for the ribosome data set. We assume that this is because of the high molecular weight of these particles, which allows precise modelling of the motion tracks. The γ-secretase data set yields the smallest increases in resolution in comparison with both *MotionCor*2 and our previous results. Possible reasons for this will be discussed in the following.

We have also analysed the resolution of the resulting reconstructions as a function of the number of particles, as proposed by Rosenthal & Henderson (2003). These plots are shown for both our results and those obtained from the UCSF

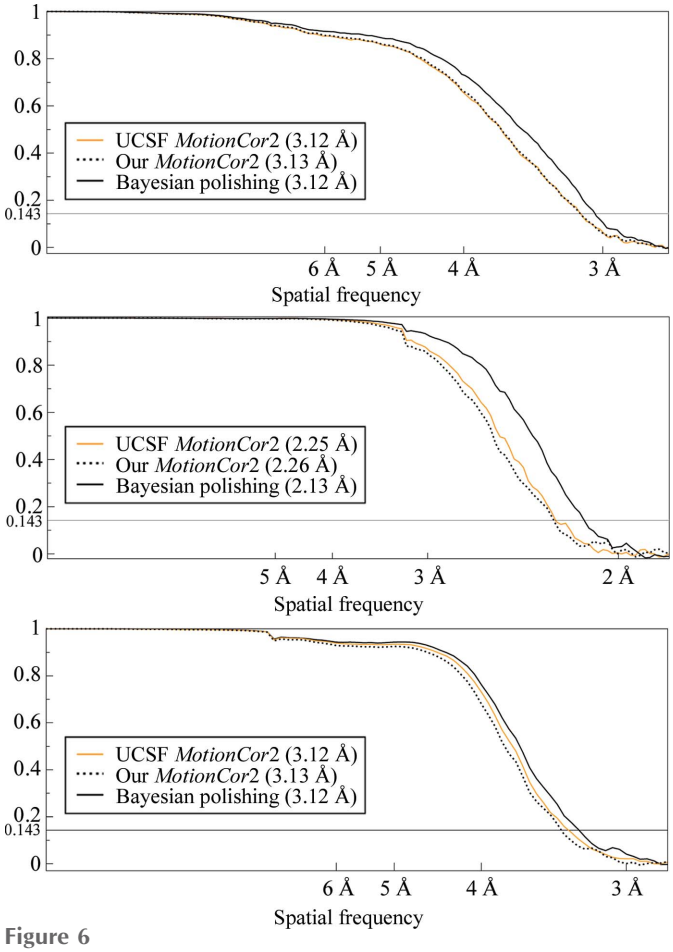


Figure 6
Gold-standard FSC plots for the ribosome (top), β-galactosidase (centre) and γ-secretase (bottom). The values in parentheses indicate the 0.143 FSC resolution. The continuous orange line results from the official UCSF implementation of *MotionCor*2 and the dotted black line from our own implementation.

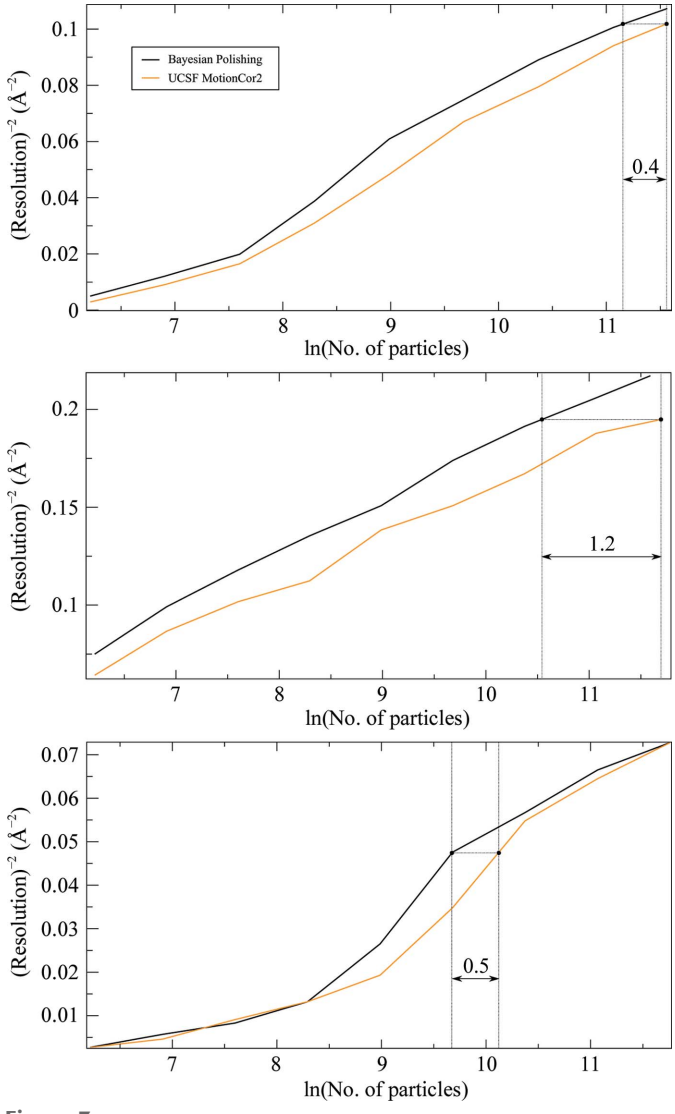


Figure 7
Plot of the inverse-squared resolution as a function of the number of particles, as proposed by Rosenthal & Henderson (2003), for the ribosome (top), β-galactosidase (centre) and γ-secretase (bottom). The horizontal distance between the curves describes the fraction of the number of particles required to obtain the same resolution with Bayesian polishing as with the UCSF implementation of *MotionCor*2. The indicated distances correspond to 66%, 30% and 60% of the particles, respectively. Note that the horizontal distance shrinks to zero at the right-hand edge of the γ-secretase plot. This implies that the γ-secretase data set is limited by additional effects at high resolutions.

implementation of *MotionCor*2 in Fig. 7. They indicate that in order to reach the same maximum resolution with Bayesian polishing as with the UCSF implementation of *MotionCor*2, only 66% of the particles are needed for the ribosome and as few as 30% of the particles for β-galactosidase. For γ-secretase, only 60% of the particles are needed to reach the same *intermediate* resolutions, although the same numbers of particles are required to obtain the maximum resolution. This suggests that at high resolutions, γ-secretase is limited by additional effects beyond the experimental noise and the uncertainty in particle alignment. Such effects could include molecular heterogeneity, anisotropic magnification, an insufficient particle-box size or variations in microscope parameters across the data set. The latter is especially likely since this data set was collected in six different sessions over a time span of half a year.

4. Conclusions

We have presented Bayesian polishing, a new method for the estimation of particle motion and of the corresponding per-frame relative *B* factors. We have compared our method with *MotionCor*2 and with the previously existing particle-polishing method in *RELION* on three publicly available data sets. In all three cases, Bayesian polishing led to an increase in resolution over both alternatives. Since the FSC-based resolution estimates are influenced by many other factors besides particle motion, the accuracy of motion estimation was also measured by comparing the estimated relative *B* factors. We have shown that Bayesian polishing produces better *B* factors than our implementation of *MotionCor*2 for all frames of all data sets, with an average improvement over all three data sets of 16 Å^2, while the achieved resolution after refinement shows that our implementation of *MotionCor*2 is comparable to the official UCSF implementation. The comparison of the shapes of our new *B*-factor curves with our previously published curves suggests that Bayesian polishing captures significantly more of the initial motion than the existing particle-polishing method in *RELION*. This allows the use of almost as much high-resolution data from the first few movie frames as from the intermediate movie frames, thereby obviating the need for the practice of discarding the first few movie frames (Li *et al.*, 2013). Finally, we have shown that the new FCC-based technique of estimating *B* factors measures the same *B* factors as the existing particle-polishing method, but much faster and more robustly.

We have also presented a method that enables the user to determine the optimal parameters governing the statistics of motion. Since the final resolution of the resulting reconstructions appears to be relatively insensitive to these parameters, and the parameter hyper-optimization algorithm requires considerable amounts of memory, we do not necessarily recommend estimating new parameters for each data set. Instead, we expect that use of the default values should produce similar results, unless the data set has been collected under unusual conditions. For example, re-estimating the motion parameters may be necessary for data sets that exhibit a much smaller fractional electron dose or significantly thinner or thicker ice, or if special grids are used that are designed to minimize beam-induced motion.

Bayesian polishing has been implemented as part of the open-source release of *RELION*-3.0. The new implementation no longer requires the storage of aligned micrograph movies or movie particles, and is capable of performing on-the-fly gain correction on movies stored in compressed TIFF format. Thus, the new implementation strongly reduces the storage requirements of performing particle polishing in *RELION*. Because the new method has outperformed the previously existing particle polishing in all tests performed, the new approach replaces the old one in the graphical user interface (GUI) of *RELION*-3.0.

Acknowledgements

We thank Jake Grimmett and Toby Darling for assistance with high-performance computing, and Christopher Russo and Richard Henderson for critical reading of the manuscript.

References

Abrishami, V., Vargas, J., Li, X., Cheng, Y., Marabini, R., Sorzano, C. Ó. S. & Carazo, J. M. (2015). *J. Struct. Biol.* **189**, 163–176.
Bai, X.-C., Fernandez, I. S., McMullan, G. & Scheres, S. H. W. (2013). *Elife*, **2**, e00461.
Bai, X.-C., Yan, C., Yang, G., Lu, P., Ma, D., Sun, L., Zhou, R., Scheres, S. H. W. & Shi, Y. (2015). *Nature (London)*, **525**, 212–217.
Baker, L. A. & Rubinstein, J. L. (2010). *Methods Enzymol.* **481**, 371–388.
Brilot, A. F., Chen, J. Z., Cheng, A., Pan, J., Harrison, S. C., Potter, C. S., Carragher, B., Henderson, R. & Grigorieff, N. (2012). *J. Struct. Biol.* **177**, 630–637.
Campbell, M. G., Cheng, A., Brilot, A. F., Moeller, A., Lyumkis, D., Veesler, D., Pan, J., Harrison, S. C., Potter, C. S., Carragher, B. & Grigorieff, N. (2012). *Structure*, **20**, 1823–1828.
Chen, S., McMullan, G., Faruqi, A. R., Murshudov, G. N., Short, J. M., Scheres, S. H. W. & Henderson, R. (2013). *Ultramicroscopy*, **135**, 24–35.
Grant, T. & Grigorieff, N. (2015). *Elife*, **4**, e06980.
Hayward, S. B. & Glaeser, R. M. (1979). *Ultramicroscopy*, **4**, 201–210.
Kimanius, D., Forsberg, B. O., Scheres, S. H. W. & Lindahl, E. (2016). *Elife*, **5**, e18722.
Li, X., Mooney, P., Zheng, S., Booth, C. R., Braunfeld, M. B., Gubbens, S., Agard, D. A. & Cheng, Y. (2013). *Nat. Methods*, **10**, 584–590.
Liu, D. C. & Nocedal, J. (1989). *Math. Program.* **45**, 503–528.
Lucas, B. D. & Kanade, T. (1981). *Proceedings of the DARPA Image Understanding Workshop*, pp. 121–130.
Lüthi, M., Gerig, T. & Vetter, T. (2018). *IEEE Trans. Pattern Anal. Mach. Intell.* **40**, 1860–1873.
McLeod, R. A., Kowal, J., Ringler, P. & Stahlberg, H. (2017). *J. Struct. Biol.* **197**, 279–293.
Nelder, J. A. & Mead, R. (1965). *Comput. J.* **7**, 308–313.
Nguyen-Tuong, D., Seeger, M. & Peters, J. (2009). *Adv. Robot.* **23**, 2015–2034.
Rasmussen, C. E. (2004). *Advanced Lectures on Machine Learning*, edited by O. Bousquet, U. von Luxburg & G. Rätsch, pp. 63–71. Berlin, Heidelberg: Springer.
Rosenthal, P. B. & Henderson, R. (2003). *J. Mol. Biol.* **333**, 721–745.

Rubinstein, J. L. & Brubaker, M. A. (2015). *J. Struct. Biol.* **192**, 188–195.
Scheres, S. H. W. (2012). *J. Struct. Biol.* **180**, 519–530.
Scheres, S. H. W. (2014). *Elife*, **3**, e03665.
Scheres, S. H. W. & Chen, S. (2012). *Nat. Methods*, **9**, 853–854.
Unwin, P. N. T. & Henderson, R. (1975). *J. Mol. Biol.* **94**, 425–440.
Wang, J. M., Fleet, D. J. & Hertzmann, A. (2008). *IEEE Trans. Pattern Anal. Mach. Intell.* **30**, 283–298.
Wong, W., Bai, X.-C., Brown, A., Fernandez, I. S., Hanssen, E., Condron, M., Tan, Y. H., Baum, J. & Scheres, S. H. W. (2014). *Elife*, **3**, e03080.
Zheng, S. Q., Palovcak, E., Armache, J.-P., Verba, K. A., Cheng, Y. & Agard, D. A. (2017). *Nat. Methods*, **14**, 331–332.