arXiv:1710.09791v1 [stat.AP] 23 Oct 2017

Structural Variability from Noisy Tomographic Projections[*]

Joakim Andén[†] and Amit Singer[‡]

Abstract. In cryo-electron microscopy, the 3D electric potentials of an ensemble of molecules are projected along arbitrary viewing directions to yield noisy 2D images. The volume maps representing these potentials typically exhibit a great deal of structural variability, which is described by their 3D covariance matrix. Typically, this covariance matrix is approximately low-rank and can be used to cluster the volumes or estimate the intrinsic geometry of the conformation space. We formulate the estimation of this covariance matrix as a linear inverse problem, yielding a consistent least-squares estimator. For n images of size N-by-N pixels, we propose an algorithm for calculating this covariance estimator with computational complexity $\mathcal{O}(nN^4 + \sqrt{\kappa}N^6 \log N)$, where the condition number κ is empirically in the range 10–200. Its efficiency relies on the observation that the normal equations are equivalent to a deconvolution problem in 6D. This is then solved by the conjugate gradient method with an appropriate circulant preconditioner. The result is the first computationally efficient algorithm for consistent estimation of 3D covariance from noisy projections. It also compares favorably in runtime with respect to previously proposed non-consistent estimators. Motivated by the recent success of eigenvalue shrinkage procedures for high-dimensional covariance matrices, we introduce a shrinkage procedure that improves accuracy at lower signal-to-noise ratios. We evaluate our methods on simulated datasets and achieve classification results comparable to state-of-the-art methods in shorter running time. We also present results on clustering volumes in an experimental dataset, illustrating the power of the proposed algorithm for practical determination of structural variability.

Key words. cryo-electron microscopy, heterogeneity, single-particle reconstruction, principal component analysis, deconvolution, Toeplitz matrices, shift invariance, conjugate gradient method

AMS subject classifications. 92C55, 68U10, 44A12, 65R32, 62G05, 62H30, 62J10, 62J07

1. Introduction. A single biological macromolecule often exists in a variety of three-dimensional configurations. These can be due to deformations of the molecular structure, known as conformational variability, or smaller molecules being added or removed, known as compositional variability. Since molecular structure dictates biological function, properly resolving these different configurations is of great importance in structural biology. In some cases, it is possible to separate the different structures experimentally and subsequently image each separately in order to reconstruct its three-dimensional structure. Often, however, this is not possible due to the similarity in shape and size of the various configurations. In this case, the particles are imaged in a single heterogeneous sample, and their structural variability must be taken into account at the reconstruction stage.

Traditional methods such as X-ray crystallography and nuclear magnetic resonance (NMR)

[†]Center for Computational Biology, Flatiron Institute, New York, NY (janden@flatironinstitute.org).
[‡]Department of Mathematics and Program in Applied and Computational Mathematics, Princeton University, NJ (amits@math.princeton.edu).
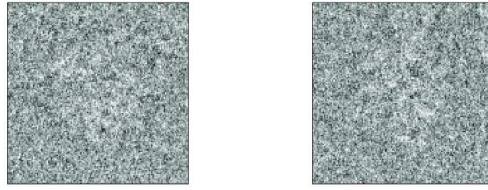
Figure 1. *Two sample cryo-EM images from a 10000-image dataset depicting the 70S ribosome complex in E. Coli [43]. Each image measures 130-by-130 with a pixel size of 2.82 Å. The images depict two similar molecular structures projected in approximately the same viewing direction, but the high noise level makes it difficult to distinguish the difference in structure.*

imaging are not well suited for this task, since both rely on aggregate measurements from the whole sample. In contrast, methods such as single-particle cryo-electron microscopy (cryo-EM) image each particle separately, and can thus potentially recover the structural variability in the sample. Unlike X-ray crystallography, cryo-EM does not require the crystallization of the sample and can handle larger molecules compared to NMR (and so far as small as 64 kDa [37]), making it a more flexible method. In recent years, new sample preparation techniques and better detectors have yielded reconstructions at near-atomic resolution and the method's popularity has been steadily on the rise [40, 2, 45]. In 2017, the Nobel Prize in Chemistry was awarded to three pioneers of cryo-EM and the technique was named Method of the Year in 2015 by Nature Methods [21].

To image a set of particles using single-particle cryo-EM, the sample is frozen in a thin layer of ice and exposed to an electron beam. The transmitted electrons are then recorded, forming a set of noisy projection images, one for each particle. Images are modeled as the integral of the particle's electric potential along a particular viewing direction, followed by convolution with a point spread function and addition of noise [24]. Damaging ionization effects are reduced by limiting electron dose, resulting in images dominated by noise (see Figure 1).

The traditional reconstruction task in single-particle cryo-EM assumes that all particles have identical structure and attempts to reconstruct that structure. As mentioned above, however, this is not always the case. The task of reconstructing the variability of molecular structure of a heterogeneous population is known as the heterogeneity problem. Typically, structural variability is assumed to be discrete or continuous. In the case of discrete variability, each particle takes on a finite number of possible molecular configurations. For continuous variability, the molecular structures vary continuously, forming a smooth manifold on which each point corresponds to a distinct configuration.

The single-particle reconstruction problem in cryo-EM has been approached from many directions, bringing together ideas from statistics and tomography [24, 12, 50, 78, 6]. For heterogeneous data, the popular RELION software implements a regularized maximum-likelihood estimator using expectation-maximization [63]. It fits a parametric model of discrete variability, so the number of molecular structures needs to be specified in advance. In addition, a high-quality initialization is often required for successful reconstruction. Moreover, there is no global convergence guarantee. Lastly, most of these algorithms require a significant amount

of computational resources, although this has recently been partly mitigated by exploiting advances in hardware using graphics processing units (GPUs) [38, 57].

Suppose the electric potentials are represented on an N-by-N-by-N voxel grid, which we will refer to as volume vectors. A promising approach, first outlined by Penczek et al. [55], is to estimate the covariance of the volume vectors, known as the 3D covariance matrix. The covariance will typically be approximately low-rank, since addition or removal of a substructure is captured by a single volume vector, while deformations are often limited in spatial extent and therefore well-approximated using a small number of vectors. The top eigenvectors, or "eigenvolumes," of the 3D covariance thus describe the dominant modes of variability in the volumes. Projecting these eigenvolumes in the viewing direction of each image and calculating the least-squares fit yields a set of coordinates for that image. The low dimensionality of the space significantly reduces the noise in the coordinates compared to the original images. Using the coordinates, the images are then clustered and each cluster is then used to reconstruct a volume using standard tomographic inversion techniques. An additional advantage is that the number of clusters need not be known prior to running the algorithm. If there are C volume states, the 3D covariance has rank at most $C-1$, so one plus the number of dominant eigenvalues bounds the number of clusters.

To estimate the 3D covariance matrix, Penczek et al. use a heuristic bootstrap-like estimator which unfortunately does not come with consistency guarantees. To remedy this, an alternate approach was proposed by Katsevich et al. [35], where the 3D covariance estimation problem is formulated as a linear inverse problem and a least-squares estimator was derived. While this estimator is consistent, its calculation involves solving a large-scale linear system, which is prohibitively expensive to invert directly for typical problem sizes. The authors therefore propose a block-diagonal approximation to the linear system in the large-sample limit which can be solved efficiently, but this is only valid for a uniform distribution of viewing angles and a fixed microscope point spread function, and is therefore of limited applicability in experimental datasets. A new approach was proposed by Andén et al. [3], where the same linear system is solved using the conjugate gradient (CG) method [31]. However, the method requires many iterations to converge due to ill-conditioning of the system. In addition, each iteration requires a separate pass through the entire dataset, resulting in long run times.

In this paper, we propose an improved version of the method of Andén et al. [3] for efficient and accurate estimation of the 3D covariance matrix. Our method exploits the fact that projection followed by its dual (backprojection) is a convolution operator [20], also known as a Toeplitz operator. This has already resulted in efficient reconstruction techniques in MRI [82, 23, 29] and cryo-EM [79, 83]. The 3D covariance least-squares estimator has a similar structure, letting us pose it as a deconvolution problem in six dimensions. As a result, only one pass through the dataset is required to calculate the convolution kernel, allowing each CG iteration to be computed quickly. To reduce the number of iterations required for convergence, we employ a circulant preconditioner [76] to improve the conditioning of the system. Our method makes mild assumptions on the distribution of viewing angles and handles image-dependent point spread functions, providing a flexible method for covariance estimation. It is a consistent estimator of the 3D covariance, but unlike the methods of Katsevich et al. [35] and Andén et al. [3], it can be applied efficiently to a wide range of data.

The proposed algorithm has computational complexity $\mathcal{O}(nN^4 + \sqrt{\kappa}N^6 \log N)$ for n images, where κ is the condition number of the preconditioned convolution operator and is typically in the range 1–200. This outperforms the algorithm of Katsevich et al. [35], which has computational complexity of $\mathcal{O}(nN^6 + N^{9.5})$ [35]. It similarly outperforms the method of Andén et al. [3], which has a complexity of $\mathcal{O}(\sqrt{\kappa'}nN^7)$, whose condition number κ' is of the order of 5000. The computational complexity is also lower compared to the covariance matrix estimation method introduced by Liao et al. [44], which uses a block Kaczmarz method. Although the paper does not provide an explicit computational complexity of the algorithm, it can be verified to have a complexity of at least $\mathcal{O}(TN^{10})$, where T is the number of iterations (typically around 20).

We also introduce a modified covariance estimator based on eigenvalue shrinkage, which lowers the estimation error in the high-dimensional regime. This technique is based on prior work for high-dimensional covariance estimation, where eigenvalue shrinkage methods have been shown to consistently outperform other approaches [18, 25, 17]. These ideas are most relevant when dealing with vectors of dimensionality comparable to the number of samples, which is often the case for cryo-EM. As a result, we can accurately estimate the 3D covariance at lower signal-to-noise ratios than is possible for the conventional least-squares estimator.

We evaluate the proposed algorithms on simulated datasets, showing their ability to deal with high noise levels, non-uniform distribution of viewing angles, and optical aberrations. In particular, we find that the number of images n necessary to obtain a given covariance estimation accuracy scales inversely with the square of the signal-to-noise ratio, with a phase transition occurring at a critical noise level for a fixed n. We also compare our algorithm to the state-of-the-art RELION software [63], obtaining superior accuracy in shorter computation time without using an initial reference structure. The eigenvalue shrinkage variant outperforms the standard estimator, achieving the same accuracy for noise levels up to a factor of two higher. Finally, we evaluate the algorithms on several experimental datasets, where we obtain state-of-the-art reconstruction results. GNU Octave/MATLAB code to reproduce the experiments and figures in this paper are provided by functions located in the heterogeneity folder of the ASPIRE toolbox available at http://spr.math.princeton.edu/.

The remainder of the paper is organized as follows. Section 2 presents the heterogeneity problem in cryo-EM, while some background and existing approaches are described in Section 3. Section 4 describes the least-squares estimators for the volume mean and covariance and proposes an eigenvalue shrinkage estimator to improve accuracy in the high-dimensional regime. The proposed algorithms for efficient calculation of these estimators are presented in Section 5. Once the mean and covariance have been estimated, we describe their use for image clustering in Section 6, while simulation and experimental results are provided in Sections 7 and 8, respectively. Possible directions for future work are discussed in Section 9.

2. Problem setup. To model the cryo-EM imaging process, we equate molecular structure with its electric potentials in three dimensions. These potential maps, referred to as volumes, exist in a variety of states, characterizing the structural variability of the molecule. We consider the volume function $\mathcal{X} : \mathbb{R}^3 \to \mathbb{R}$ to be a random field of unknown distribution such that $\mathcal{X} \in L^1(\mathbb{R}^3)$. If the molecule takes on C distinct configurations, $\mathcal{X}$ is modeled as a discrete random variable with C states, each of which is a function in $L^1(\mathbb{R}^3)$.

The electron microscope sends a stream of electrons through the particle represented by $\mathcal{X}$, which scatters the electrons. The result is a distorted tomographic projection of each volume, which can be modeled in the weak-phase approximation by an integral along a certain viewing angle followed by a convolution of the resulting image with a microscope-dependent point spread function [24]. Due to the freezing process, each particle in the ice layer has a different orientation. We denote the rotation, or viewing direction, of the particle with respect to some reference frame by the 3-by-3 rotation matrix R, which we assume is drawn from some distribution over the rotation group SO(3). We then define the projection of $\mathcal{X}$ along R to be the image

$$(1) \qquad \mathcal{Z}(\boldsymbol{u}) := \int_{\mathbb{R}} \mathcal{X}(R^{\mathrm{T}}[\boldsymbol{u}; z]) dz,$$

where $\boldsymbol{u} = \mathbb{R}^2$ and $[\boldsymbol{u}; z] \in \mathbb{R}^3$ is the concatenation of $\boldsymbol{u}$ with z. This mapping is also known as the X-ray transform of $\mathcal{X}$ along R [53]. In addition to the tomographic projection, the configuration of the microscope induces a certain amount of optical aberration, which is modeled by a convolution with some point spread function $h \in L^1(\mathbb{R}^2)$ [81, 22]. Again, we assume that this is drawn from some (typically discrete) distribution over $L^1(\mathbb{R}^2)$. The convolution is defined by

$$(2) \qquad \mathcal{Y}(\boldsymbol{u}) := \int_{\mathbb{R}^2} \mathcal{Z}(\boldsymbol{v} - \boldsymbol{u}) h(\boldsymbol{v}) d\boldsymbol{v},$$

for $\boldsymbol{u} \in \mathbb{R}^2$. Combining both operations, we have the projection mapping $\mathcal{P} : L^1(\mathbb{R}^3) \to L^1(\mathbb{R}^2)$

$$(3) \qquad \mathcal{P}\mathcal{X}(\boldsymbol{u}) := \int_{\mathbb{R}^2} \left(\int_{\mathbb{R}} \mathcal{X}(R^{\mathrm{T}}[\boldsymbol{u} - \boldsymbol{v}; z]) dz \right) h(\boldsymbol{v}) d\boldsymbol{v}.$$

We can now state our forward model for the cryo-EM imaging process, which takes a volume $\mathcal{X}$ and gives the image

$$(4) \qquad \mathcal{Y} = \mathcal{P}\mathcal{X} + \mathcal{E},$$

where $\mathcal{E} : \mathbb{R}^2 \to \mathbb{R}$ is a white Gaussian random field of variance σ^2. The noise represents error introduced into the image due to non-interacting electrons, inelastic scattering, and quantum noise [80, 7, 56]. While these error sources follow a Poisson distribution, the counts are typically high enough for this to be well-approximated by a Gaussian distribution. The noise is also rarely white, but images are prewhitened prior to any analysis and can therefore be assumed to have approximately white noise. Due to changing experimental conditions, the intensities of the various noise sources vary from image to image. To account for this, we employ a noise estimation algorithm which exploits such low-rank variability [4].

It is useful to consider the above mappings in the Fourier domain. Let us define the d-dimensional Fourier transform $\mathcal{F}\mathcal{G}$ of some function $\mathcal{G} \in L^1(\mathbb{R}^d)$ by

$$(5) \qquad \mathcal{F}\mathcal{G}(\boldsymbol{\omega}) := \int_{\mathbb{R}^d} \mathcal{G}(\boldsymbol{u}) \mathrm{e}^{-2\pi \mathrm{i} \langle \boldsymbol{\omega}, \boldsymbol{u} \rangle} d\boldsymbol{u},$$

for any frequency $\boldsymbol{\omega} \in \mathbb{R}^d$. In this case, the tomographic projection mapping $\mathcal{P}$ satisfies

$$(6) \qquad \mathcal{F}\mathcal{P}\mathcal{X}(\boldsymbol{\omega}) = \mathcal{F}\mathcal{X}\left(R^{\mathrm{T}}[\boldsymbol{\omega};0]\right) \cdot \mathcal{F}h(\boldsymbol{\omega}),$$

for any $\boldsymbol{\omega} \in \mathbb{R}^2$, which is known as the Fourier Slice Theorem [53]. In other words, projection in the spatial domain corresponds to restriction (or "slicing") to a plane in the Fourier domain and multiplication by a transfer function $\mathcal{F}h$. The Fourier transform $\mathcal{F}h$ of the point spread function h is known as the contrast transfer function (CTF).

The model presented above describes continuous images $\mathcal{Y}$ obtained from continuous volume densities $\mathcal{X}$. While an accurate model of the physical process, it is not compatible with the output of an electron microscope, which is in the form of discrete images y with values on an N-by-N pixel grid, where N typically ranges from 100 to 500. The images are therefore limited in resolution which imposes a similar limit on the resolution of the reconstructed volumes.

To discretize the images, we define the N-point grid M_N as

$$(7) \qquad M_N \coloneqq \{-\lfloor N/2 \rfloor, \ldots, \lceil N/2 - 1 \rceil\}.$$

An image $\mathcal{P}$ is then represented by sampling evenly over the square $[-1, +1]^2$ at points $2M_N^2/N$, yielding a function $y : M_N^2 \to \mathbb{R}$. We treat this function as an N^2-dimensional vector in $\mathbb{R}^{N^2}$.

There is more choice in representing the volumes. One popular approach is to consider the voxel samples of $\mathcal{X}$ at points $2M_N^3/N$ in the cube $[-1, +1]^3$, yielding a vector of dimension N^3 [62, 79, 83]. While this basis has computational advantages, it is not always well suited to representing volumes of interest. As such, we will use a different basis described in Section 5.5 and convert between this and the voxel basis. Let us denote by Q the matrix whose columns corresponding to the p basis vectors of size N^3, each corresponding to an N-by-N-by-N volume. Here $p = \mathcal{O}(N^3)$ and Q is of size N^3-by-p. We will assume that expansion and evaluation in this basis is fast, that is both Q and its transpose Q^{T} can be applied in $\mathcal{O}(N^3 \log N)$. To simplify expressions, we introduce the notation $v = Qx$.

An important question is then how to properly discretize the forward model (4). One approach is embed the discretized volumes $v \in \mathbb{R}^{N^3}$ into $L^1(\mathbb{R}^3)$ using a sinc basis, apply $\mathcal{P}$ and fit the result to a sinc basis expansion in $L^1(\mathbb{R}^2)$ using least-squares. While this provides a matrix representation of $\mathcal{P}$ that is accurate in a least-squares sense and converges to $\mathcal{P}$ as $N \to \infty$, the mapping is computationally inefficient, requiring a full N^2-by-N^3 matrix multiplication to apply.

Another approach is to mimic the Fourier slice structure of $\mathcal{P}$ described in (6), enabling speedups associated with fast Fourier transforms (FFTs) [14]. First, let us define the discrete Fourier transform of some function $g : M_N^d \to \mathbb{R}$ in d dimensions

$$(8) \qquad \mathcal{F}g(\boldsymbol{k}) \coloneqq \sum_{\boldsymbol{i} \in M_N^d} g(\boldsymbol{i}) \mathrm{e}^{-2\pi \mathrm{i} \langle \boldsymbol{i}, \boldsymbol{k} \rangle / N}.$$

Here, we have abused notation slightly by having $\mathcal{F}$ signify both the continuous and discrete Fourier transforms. The nature of the mapping should be clear from context. Note that this

transform is defined for any frequency vector $\boldsymbol{k} \in \mathbb{R}^d$, although they are traditionally restricted to the grid M_N^d.

We now introduce the mapping I, which transforms the voxel volume v into the image Iv through

$$(9) \qquad Iv(\boldsymbol{i}) = \frac{1}{N^3} \sum_{\boldsymbol{j} \in M_N^3} v(\boldsymbol{j}) \sum_{\boldsymbol{k} \in M_{2\lceil N/2 \rceil - 1}^2} \mathcal{F}h(\boldsymbol{k}) \mathrm{e}^{-2\pi \mathrm{i}(\langle R^{\mathrm{T}}[\boldsymbol{k};0], \boldsymbol{j} \rangle - \langle \boldsymbol{k}, \boldsymbol{i} \rangle)/N} \quad \text{for all } \boldsymbol{i} \in M_N^2.$$

Computing its discrete Fourier transform, we obtain

$$(10) \qquad \mathcal{F}Iv(\boldsymbol{k}) = \begin{cases} \dfrac{1}{N} \mathcal{F}v\left(R^{\mathrm{T}}[\boldsymbol{k};0]\right) \cdot \mathcal{F}h(\boldsymbol{k}), & \boldsymbol{k} \in M_{2\lceil N/2 \rceil - 1}^2 \\ 0, & \text{otherwise,} \end{cases}$$

for any $\boldsymbol{k} \in M_N^2$. The operator I therefore satisfies a discrete version of the Fourier Slice Theorem (6). Note that in the case of even N, the Nyquist frequencies at $-N/2$ are set to zero to ensure a real image IQx. Enforcing this one-to-one mapping of frequencies allows us to derive the convolutional formulations in Sections 5.1 and 5.2. The entire mapping, from x to $v = Qx$ to $Iv = IQx$ is denoted by $P = IQ$ and is called the volume projection (or imaging) mapping.

To project a volume $x \in \mathbb{R}^{N^3}$, first evaluate it on the $2M_N^3/N$ voxel grid to obtain $v = Qx$, then calculate the discrete Fourier transform $\mathcal{F}v$ using (8) on the grid defined by $R^{\mathrm{T}}[\boldsymbol{k};0]$. We then multiply the Fourier transform pointwise by the contrast transfer function $\mathcal{F}h(\boldsymbol{k})$, set Nyquist frequencies to zero, and apply the inverse discrete Fourier transform, which gives $Px \in \mathbb{R}^{N^2}$. As mentioned earlier, the basis evaluation matrix Q can be applied in $\mathcal{O}(N^3 \log N)$ time. The first discrete Fourier transform $\mathcal{F}v$ is computed using a non-uniform fast Fourier transform (NUFFT), which has a computational complexity of $\mathcal{O}(N^3 \log N)$ [20, 27], while pointwise multiplication and the 2D inverse FFT require $\mathcal{O}(N^2)$ and $\mathcal{O}(N^2 \log N)$, respectively. The overall computational complexity is therefore $\mathcal{O}(N^3 \log N)$, which is a significant improvement over the direct matrix multiplication approach, which has a complexity of $\mathcal{O}(N^5)$. Another important advantage is that calculating multiple projections of the same volume x, the overall complexity scales as $\mathcal{O}(N^3 \log N + nN^2)$, where n is the number of projection images.

With the projection mapping P, we can now formulate our discrete forward model as

$$(11) \qquad y := Px + e,$$

where $e \in \mathbb{R}^{N^2}$ is a standard white Gaussian noise image. In a single-particle cryo-EM experiment, however, we have more than one image, with multiple copies of the same molecule being imaged separately, each with a different structural configuration, projected at a different viewing angle, subjected to convolution by a different point spread function, and degraded by a different realization of noise. As a result, we consider identically distributed, independent copies $x_1, \ldots, x_n$ of x. Similarly, we have rotations $R_1, \ldots, R_n$, point spread functions $h_1, \ldots, h_n$ and noise vectors $e_1, \ldots, e_n$ which are independent and identically distributed copies of R, h, and e, respectively. These give rise to the images

$$(12) \qquad y_s := P_s x_s + e_s$$

for $s = 1, \ldots, n$, where P_s is the projection mapping corresponding to the viewing direction R_s and contrast transfer function $\mathcal{F}h_s$. Typically, the contrast transfer functions $\mathcal{F}h_1, \ldots, \mathcal{F}h_n$ are not all distinct, but can be grouped into a small number of distinct functions (ranging from 10 to 500).

We note that the P_s mappings are not known for experimental data, but must be estimated. While several methods exist to estimate the CTFs $\mathcal{F}h_1, \ldots, \mathcal{F}h_n$ [51], estimating the molecular orientation R_s is non-trivial when x exhibits a large amount of variability. When x does not vary too much around its mean, traditional methods for orientation estimation can be applied [84, 63]. Indeed, previous works have demonstrated that this is a feasible approach in different situations [55, 44]. In this work, we assume that this is the case and so $P_1, \ldots, P_n$ are known to a certain accuracy.[1]

The heterogeneity problem can now be stated more formally. From projection images $y_1, \ldots, y_n$, we would like to characterize the distribution of x and reconstruct the underlying volumes $x_1, \ldots, x_n$ corresponding to each image. Since the projection mappings P_s are not invertible, this cannot be done directly, so more sophisticated methods are needed.

3. Related work. Due to the importance of determining structural variability from cryo-EM projections, much work has been focused on resolving this heterogeneity problem. Although various methods have been introduced that have some degree of experimental success, they do not possess any accuracy guarantees, so it is sometimes difficult to validate the reconstructions. In addition, they also often rely on good initializations, which can significantly bias the final result. Finally, the computational complexity of these methods is typically quite high, requiring a large amount of computational resources.

3.1. Maximum likelihood. One popular method for solving the heterogeneity problem has been to set up a probabilistic model for image formation and maximizing the likelihood function with respect to the model parameters given the data. This was first considered for class averaging in the space of images by Sigworth [68], where the probability density of the images y was modeled as a mixture of Gaussians, with each component center constituting a distinct image class. These centers, along with other parameters such as shifts were estimated by maximizing the likelihood using an expectation-maximization algorithm [16].

The maximum likelihood method was subsequently extended by Scheres, who modeled the underlying volume vector x as a mixture of Gaussians and regularized the likelihood function using Bayesian priors on the parameters, which includes the viewing directions $R_1, \ldots, R_n$ [63]. The resulting algorithm, implemented in the RELION software package, has seen significant success and provides generally satisfactory volume estimates [62]. However, since the algorithm attempts to optimize a non-convex function, there is no guarantee that a globally optimal solution is obtained. The algorithm also needs to be initialized with single reference structure that is similar to the molecule being imaged, which can significantly bias the result if not chosen carefully. Similarly, the number of clusters is part of the model and needs to be specified in advance, limiting the method's flexibility. Finally, the optimization typically

---

[1]In the case of high variability in x, the orientations must be estimated simultaneously with the clustering of the images, a more challenging problem recently studied in terms of non-unique games by Lederman and Singer [42].

takes a long time to converge, although this has been remedied recently by a new GPU-based implementation [38].

3.2. Common lines. Another approach proposed by Shatsky et al. [65] clusters the projection images by defining a similarity measure between all pairs and applying a spectral clustering method. From the Fourier Slice Theorem (6), the Fourier transform $\mathcal{F}y$ of a clean projection image y corresponds to the restriction of the volume Fourier transform $\mathcal{F}x$ to a plane and multiplied by a contrast transfer function $\mathcal{F}h$. Two images y_s, y_t thus occupy two central planes of the volume Fourier transform and intersect along a common line. The Fourier transforms $\mathcal{F}y_s, \mathcal{F}y_t$ of two noiseless projections of the same molecular structure should therefore coincide along this common line (up to differing contrast transfer functions), so their cross-correlation along this line provides a good similarity measure.

Using this common-lines affinity, the authors applied spectral clustering to group the images according to their underlying molecular structure. Unfortunately, the Fourier transforms of two projections of the same volume will not coincide exactly due to the image noise. In most cryo-EM experiments, the images are dominated by noise, making this particular approach unfeasible without some amount of denoising. Denoising images in cryo-EM is traditionally achieved by class averaging, where images that represent similar views are averaged together. However, for heterogeneous data this may break down since images belonging to different molecular structures could be averaged together.

3.3. Covariance estimation. Instead of directly clustering the images, Penczek et al. [55] proposed to first estimate the 3D covariance matrix $\text{Cov}[x]$ of the volumes. In that work, the authors introduce a heuristic bootstrap-like method, where different subsets of the images $y_1, \ldots, y_n$ are selected and a volume is reconstructed from each subset. The sample mean and sample covariance were then computed from the resulting set of volumes. Since the distribution of molecular structures differs slightly between subsets, the idea is that this will capture the 3D variability of the volumes.

The top eigenvectors, or eigenvolumes, of the estimated 3D covariance matrix are then used to reconstruct the volumes in the sample. Indeed, as discussed above, this covariance matrix is typically approximated by a low-rank matrix, so its top eigenvolumes together with the mean volume form an affine space containing most of the volumes in the dataset. Projecting the mean volume and the eigenvolumes, the authors find the coordinates of each image in this affine space through a least-squares fit. Clustering the images using these coordinates, each cluster is then used to reconstruct a different molecular structure. Unfortunately, the heuristic nature of the covariance estimation does not provide any accuracy guarantees. A maximum-likelihood approach has also been proposed to estimate the top eigenvolumes [72], but this suffers from the same initialization problems and lack of guarantees as other non-convex approaches (see Section 3.1).

To address these problems, Katsevich et al. [35] formulated the 3D covariance estimation problem as a linear inverse problem and proposed a least-squares estimator with proven consistency results. Unfortunately, direct calculation of the estimator proved computationally intractable, so the authors introduced an approximation which relied on uniform distribution of viewing directions and a single fixed contrast transfer function. These conditions are rarely satisfied in experimental data, so practical use of this method was limited. A more flexible ap-

proach has been proposed, based on calculating a related estimator using the block Kaczmarz method, but unfortunately this suffers from slow convergence and reduced accuracy [44].

An improved version of the method of Katsevich et al. [35] was introduced in Andén et al. [3], where the exact linear system was solved iteratively using the CG method. This had the advantage of allowing for non-uniform distributions of viewing directions and varying contrast transfer functions. However, the method required a pass through the dataset at each iteration and a large number of iterations was needed to reach convergence. As a result, this method proved unfeasible for large datasets.

4. Mean and Covariance Estimators. As discussed in the previous section, the 3D covariance approach of Penczek et al. [55] is powerful, but its heuristic nature does not provide any performance guarantees. In the following, we describe the least-squares estimators for both volume mean and covariance previously introduced by Katsevich et al. [35]. The estimators take as input the projection images along with estimates of the viewing directions and CTFs and provide as output estimates of the mean and covariance of the volumes. These estimators have theoretical guarantees, ensuring that for a fixed noise level, the mean and covariance estimates converge to their population values as the number of images increases. We also introduce a modified variant of the original covariance estimator, where eigenvalue shrinkage is used to reduce error in the regime of high dimension.

4.1. Least-Squares Estimators. To estimate the mean $\mathbb{E}[x]$ of the volume vectors, we fix the projection mapping P_s and consider the expectations of the images. That is, we fix the viewing angle R_s and the CTF $\mathcal{F}h_s$ while letting the volume structure x_s and the noise e_s vary. Let us denote the expectation with respect to some variable a conditioned on some other variable b by $\mathbb{E}_{a|b}$. From the forward model (11), we obtain

$$(13) \qquad\qquad \mathbb{E}_{x_s,e_s|P_s}[y_s] = P_s\mathbb{E}[x_s] = P_s\mathbb{E}[x],$$

for $s = 1,\ldots,n$, since $x_1,\ldots,x_n$ are all identically distributed and $\mathbb{E}[e_1] = \cdots = \mathbb{E}[e_n] = 0$. The above equation provides a constraint on $\mathbb{E}[x]$ for each $s = 1,\ldots,n$. We could therefore solve for $\mathbb{E}[x]$ if we were given the left-hand side expectations $\mathbb{E}_{x_s,e_s|P_s}[y_s]$, but these are unavailable to us.

However, the image itself y_s is an unbiased estimator of $\mathbb{E}_{x_s,e_s|P_s}[y_s]$, albeit one with significant variance. Substituting $y_s \approx \mathbb{E}_{x_s,e_s|P_s}[y_s]$ into (13) gives

$$(14) \qquad\qquad y_s \approx P_s\mathbb{E}[x],$$

for all $s = 1,\ldots,n$. Combining these approximate constraints into a least-squares objective, we obtain the following estimator μ_n for $\mathbb{E}[x]$:

$$(15) \qquad\qquad \mu_n := \operatorname*{argmin}_{\mu\in\mathbb{R}^p}\frac{1}{n}\sum_{s=1}^{n}\|y_s - P_s\mu\|^2.$$

This estimator minimizes the average square distance of the projected mean $P_s\mu_n$ to each of the images y_s. When there is no structural variability, that is when $\operatorname{Cov}[x] = 0$, μ_n is also the maximum-likelihood estimator for $\mathbb{E}[x]$.

We can construct a similar estimator for the covariance matrix of the volumes, $\mathrm{Cov}[x]$. Specifically, for each $s = 1, \ldots, n$, computing the covariance of (11) conditioned on P_s gives

$$(16) \qquad \mathrm{Cov}_{x_s, e_s | P_s}[y_s] = P_s \mathrm{Cov}[x] P_s^{\mathrm{T}} + \sigma^2 \mathrm{I}_{N^2},$$

where $\mathrm{Cov}[e] = \sigma^2 \mathrm{I}_{N^2}$. The left-hand side is again unknown, but we can estimate it using

$$(17) \qquad (y_s - P_s \mu_n)(y_s - P_s \mu_n)^{\mathrm{T}} \approx \mathrm{Cov}_{x_s, e_s | P_s}[y_s].$$

Plugging this into (16), we obtain

$$(18) \qquad (y_s - P_s \mu_n)(y_s - P_s \mu_n)^{\mathrm{T}} \approx P_s \mathrm{Cov}[x] P_s^{\mathrm{T}} + \sigma^2 \mathrm{I}_{N^2}.$$

Combining these constraints for all $s = 1, \ldots, n$ gives the least-squares estimator Σ_n of $\mathrm{Cov}[x]$ defined by

$$(19) \qquad \Sigma_n := \underset{\Sigma}{\mathrm{argmin}} \frac{1}{n} \sum_{s=1}^{n} \left\| (y_s - P_s \mu_n)(y_s - P_s \mu_n)^{\mathrm{T}} - (P_s \Sigma P_s^{\mathrm{T}} + \sigma^2 \mathrm{I}_{N^2}) \right\|_{\mathrm{F}}^2.$$

Similar to μ_n, this estimator finds the covariance matrix that, when projected, minimizes the average square Frobenius distance to the outer product of the mean-subtracted images.

Both (15) and (19) can be augmented by adding regularization terms, such as a Tikhonov regularizer, to enforce smoothness and other properties. We also note that Σ_n is not constrained to be positive semi-definite and so may not qualify as a covariance matrix. However, since the population covariance $\mathrm{Cov}[x]$ is approximately low-rank, we are only interested in the dominant eigenvectors, so imposing this condition would not appreciably alter our results.

The constraints (14) and (18) are by necessity loose due to the high amount of noise in the data and the limited sample size. Still, aggregating these over a large number of images n results in estimators μ_n and Σ_n that are asymptotically unbiased and consistent, as discussed in the next section.

4.2. Normal Equations. In order to calculate these estimators, we form the normal equations by differentiating the objectives and setting the derivative to zero. For the mean estimator μ_n, the objective in (15) gives

$$(20) \qquad A_n \mu_n = b_n,$$

where

$$(21) \qquad A_n := \frac{1}{n} \sum_{s=1}^{n} P_s^{\mathrm{T}} P_s,$$

$$(22) \qquad b_n := \frac{1}{n} \sum_{s=1}^{n} P_s^{\mathrm{T}} y_s.$$

The right-hand side b_n is the average of the backprojected images $P_s^{\mathrm{T}} y_s$, while A_n is the projection-backprojection operator corresponding to the set of viewing directions $R_1, \ldots, R_n$ and CTFs $\mathcal{F} h_1, \ldots, \mathcal{F} h_n$.

This least-squares estimator is a popular choice for estimating the molecular structure in cryo-EM samples when heterogeneity is not assumed to be present. Indeed, if x has no variability, $x_1 = \cdots = x_n = \mathbb{E}[x]$ then μ_n estimates the single volume present in the sample. When there is heterogeneity, it will reconstruct the average volume.

The Fourier Slice Theorem (6) is particularly useful in interpreting (20)–(22). Let us restrict to the case of no CTF, that is $\mathcal{F}h_1, \ldots, \mathcal{F}h_n = 1$. In this case P_s is pure projection and acts as restriction to a plane in the Fourier domain. Its adjoint, known as the backprojection mapping P_s^{T}, therefore inserts a two-dimensional Fourier transform into that plane, with remaining frequencies set to zero. The right-hand side b_n therefore takes the Fourier transform $\mathcal{F}y_s$ of each image, places it onto the proper plane in three-dimensional Fourier space, and averages across all images.

If the set of all planes cover enough of the three-dimensional Fourier domain, this will "reconstruct" the average volume $\mathbb{E}[x]$ up to a frequency-dependent reweighting that depends on the distribution of viewing directions. Indeed, even in the case of uniform distribution of viewing directions R over $\mathrm{SO}(3)$, frequencies close to the origin will be oversampled compared to frequencies farther away, and this must be compensated for. This reweighting is encoded in A_n through the average of the projection-backprojection operators $P_s^{\mathrm{T}}P_s$. Again, (6) tells us that $P_s^{\mathrm{T}}P_s$ in the Fourier domain is restriction followed by insertion, which is equivalent to multiplication by the indicator function of the plane corresponding to R_s. Adding all of these indicator functions together yields the reweighting A_n necessary to map μ_n to b_n. Inverting A_n thus lets us go the other direction.

We note here that in the case of uniform distribution of viewing directions, the inverse of $A = \lim_{n \to \infty} A_n$ is a ramp filter so inverting it is particularly straightforward. In this case, $A^{-1}(b_n)$ is known as filtered backprojection [30, 53, 58], a popular estimator for reconstruction in computerized tomography (CT) and related fields.

In the case of non-trivial CTFs $\mathcal{F}h_1, \ldots, \mathcal{F}h_n$, the same ideas hold, except that backprojection P_s^{T} and projection-backprojection includes multiplication by the CTF $\mathcal{F}h_s$ and the squared CTF $|\mathcal{F}h_s|^2$, respectively.

For the covariance estimator Σ_n, we can similarly differentiate its objective in (19) and set it to zero, which gives

$$(23) \qquad\qquad\qquad\qquad L_n(\Sigma_n) = B_n,$$

where

$$(24) \qquad\qquad L_n(\Sigma) := \frac{1}{n}\sum_{s=1}^{n} P_s^{\mathrm{T}} P_s \Sigma P_s^{\mathrm{T}} P_s,$$

$$(25) \qquad\qquad B_n := \frac{1}{n}\sum_{s=1}^{n} P_s^{\mathrm{T}}(y_s - P_s\mu_n)(y_s - P_s\mu_n)^{\mathrm{T}} P_s - \sigma^2 P_s^{\mathrm{T}} P_s.$$

In the same spirit as b_n, the right-hand side matrix B_n averages the backprojected outer product covariance estimators for each image with a noise term correction. The covariance projection-backprojection operator L_n plays the same role as A_n by describing the reweighting of each frequency pair according to the distribution of viewing directions and CTFs.

To interpret these estimators, we again turn to the Fourier Slice Theorem (6). Applying this to a the continuous covariance matrix $\mathcal{C} : \mathbb{R}^3 \times \mathbb{R}^3 \longrightarrow \mathbb{R}$ such that $\mathcal{C} \in L^1(\mathbb{R}^3 \times \mathbb{R}^3)$, we obtain

$$(26) \qquad (\mathcal{F} \times \mathcal{F})(\mathcal{P} \times \mathcal{P})\mathcal{C}(\boldsymbol{\omega}_1, \boldsymbol{\omega}_2) = (\mathcal{F} \times \mathcal{F})\mathcal{C}(R^{\mathrm{T}}[\boldsymbol{\omega}_1; 0], R^{\mathrm{T}}[\boldsymbol{\omega}_2; 0])\mathcal{F}h(\boldsymbol{\omega}_1)\mathcal{F}h(\boldsymbol{\omega}_2),$$

where $(\mathcal{A} \times \mathcal{B})$ denotes the mapping that applies $\mathcal{A}$ along the first variable and $\mathcal{B}$ along the second variable. In the Fourier domain, projecting a covariance matrix along its columns and rows therefore corresponds to restriction to frequency pairs belonging to a certain plane defined by R and multiplication by the CTF.

The dual formulation of (26) says that backprojecting a two-dimensional covariance matrix, that is applying $(\mathcal{P}^{\mathrm{T}} \times \mathcal{P}^{\mathrm{T}})$, corresponds to inserting its Fourier transform into a three-dimensional Fourier transform along pairs of frequencies both belonging to a certain plane. The right-hand side matrix B_n therefore takes the two-dimensional matrix estimates $(y_s - P_s\mu_s)(y_s - P_s\mu_s)^{\mathrm{T}} - \sigma^2 \mathrm{I}_{N^2}$, places their Fourier transform along the correct plane in three-dimensional covariance Fourier space, and averages across all images. Slice by slice, this then provides a "reconstruction" of the three-dimensional covariance matrix.

Much like the case for mean estimation, however, this reconstruction by backprojection needs to be reweighted in order to obtain an accurate covariance estimate. The weighting is encoded by the covariance projection-backprojection operator L_n and depends on the distribution of viewing directions and CTFs. In the case of uniform distribution of viewing angles, the same consideration of A_n applies, where frequencies closer to the origin are weighted higher with respect to frequencies farther away. The covariance variant, however, also needs to take into account the relationship between pairs of frequencies. Indeed, for a given pair of frequencies, its weight depends on how many central planes, that is images, pass through both frequencies. Since frequencies that are nearly co-linear have more central planes passing through them, this results in higher weights compared to other pairs. By inverting L_n, we renormalize the backprojected covariance estimate B_n with respect to this density. For a more detailed discussion of this phenomenon, see Katsevich et al. [35].

The above estimators μ_n and Σ_n were previously introduced by Katsevich et al. [35], who also studied their convergence properties. Specifically, they showed that the estimators are consistent and that

$$(27) \qquad \qquad \|\mu_n - \mathbb{E}[x]\| = \mathcal{O}\left(\frac{1}{\sqrt{n}}\right),$$

$$(28) \qquad \qquad \|\Sigma_n - \mathrm{Cov}[x]\| = \mathcal{O}\left(\frac{\log n}{\sqrt{n}}\right),$$

with high probability as $n \to \infty$, provided that x is bounded and A_n and L_n are invertible when $n > n_0$ for some n_0. Informally, this invertibility condition is satisfied if the central planes corresponding to the viewing directions $R_1, \ldots, R_n$ cover all frequency pairs in the Fourier domain. This is the case, for example, when the viewing directions are sampled uniformly from SO(3), but also holds more generally.

In addition, Katsevich et al. [35] defined a volume basis based on spherical harmonics in which $L = \lim_{n \to \infty} L_n$ is a block diagonal matrix with sparse blocks under certain conditions.
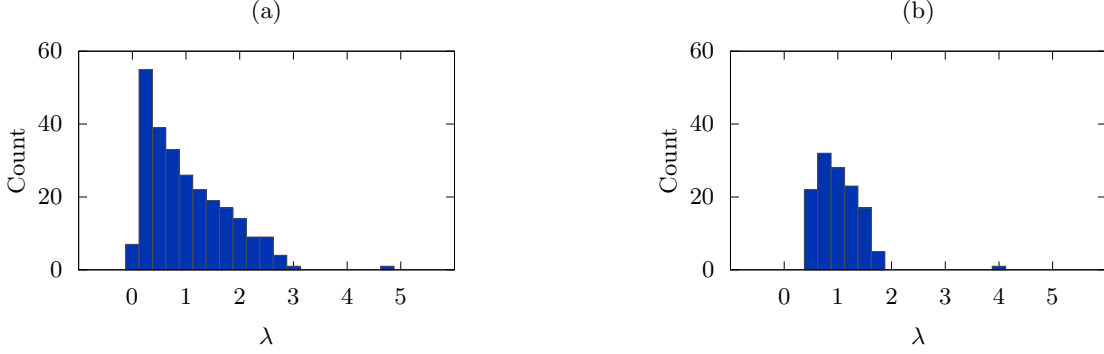
Figure 2. *The eigenvalue distribution of the sample covariance matrix for (a) $p = 256$, $n = 512$ and (b) $p = 128$, $n = 1024$ with $\sigma = 1$ and $\ell = 3$ in both regimes. For (a), we have $\gamma = 1/2$ and the spiked covariance model predicts a maximum noise eigenvalue at $(1 + \sqrt{1/2})^2 \approx 2.91$ and a signal eigenvalue at $\lambda(3, 1/2) \approx 4.67$, while for (b), $\gamma = 1/8$ gives $(1 + \sqrt{1/8})^2 \approx 1.83$ and $\lambda(3, 1/8) \approx 4.17$.*

Unfortunately, the approximation is only valid if R is uniformly distributed on $\mathrm{SO}(3)$ and the CTF is fixed. These conditions typically do not hold for experimental data. In addition, the approximation of L_n by L only holds as $n \to \infty$, and is therefore not appropriate for smaller datasets. In Section 5, we propose a method for inverting L_n accurately using an iterative algorithm.

4.3. High-Dimensional PCA. The above consistency results show that Σ_n converges to $\mathrm{Cov}[x]$ as $n \to \infty$ and in particular when $n \gg N^3$. However, in many applications, while n may be large, it is not necessarily large with respect to N^3. Indeed, while n is of the order of 10^3–10^6, N is in the range 10–400, resulting in N^3 of the order of 10^3–10^8.

In this case, a more appropriate setting is to consider the behavior of Σ_n as n and N both tend to infinity, but at potentially different rates. To illustrate covariance estimation in this regime, we consider the sample covariance of a set of independently sampled Gaussian noise vectors $w_1, \ldots, w_n$

$$(29) \qquad \frac{1}{n} \sum_{s=1}^{n} w_s w_s^{\mathrm{T}}.$$

where $\mathrm{Cov}[w_1] = \ldots = \mathrm{Cov}[w_n] = \sigma^2 \mathrm{I}_p$ for some dimension $p > 1$.

In the low-dimensional regime where $n \gg p$, all eigenvalues of this sample covariance are concentrated around the single population eigenvalue σ^2. However, for $n, p \to \infty$ where $p/n \to \gamma < 1$, the spectrum will instead spread between $\sigma^2(1 - \sqrt{\gamma})^2$ and $\sigma^2(1 + \sqrt{\gamma})^2$, following the Marčenko-Pastur distribution [48].

We model the clean signal as $a_s = \sqrt{\ell} v z_s$ for $s = 1, \ldots, n$, where v is a unit vector, $z_1, \ldots, z_n$ are i.i.d., zero-mean and unit variance random variables, while ℓ is the signal strength. The covariance of a_s is then equal to $\ell v v^{\mathrm{T}}$ and has rank one. Adding noise, we then have the measurements

$$(30) \qquad d_s = a_s + w_s,$$

for all $s = 1, \ldots, n$. As before, the sample covariance is

$$(31) \qquad \frac{1}{n} \sum_{s=1}^{n} d_s d_s^{\mathrm{T}}.$$

When $n \gg p$, its spectrum converges to $\{\ell + \sigma^2, \sigma^2, \ldots, \sigma^2\}$, with the dominant eigenvector equal to v. However, when $n, p \to \infty$ and $p/n \to \gamma < 1$, there are two possible scenarios. If $\ell/\sigma^2 < \sqrt{\gamma}$, the spectrum will be the same as the pure noise case—the signal is lost in the noise. If instead $\ell/\sigma^2 \geq \sqrt{\gamma}$, all eigenvalues will follow the Marčenko-Pastur distribution except one [54]. This signal eigenvalue will "pop out" at

$$(32) \qquad \lambda(\ell, \gamma) = (\sigma^2 + \ell)(1 + \gamma\sigma^2/\ell).$$

These distributions are illustrated for two regimes $\gamma = 1/2$ and $\gamma = 1/8$ in Figure 2. As $\ell/(\sigma^2\sqrt{\gamma})$ increases, the dominant eigenvector converges to v [54]. Specifically, the square correlation $|\langle v, u \rangle|^2$ between v and the dominant eigenvector u of the sample covariance (31) tends to

$$(33) \qquad c(\ell, \gamma) = \frac{1 - \gamma\sigma^4/\ell^2}{1 + \gamma\sigma^2/\ell}.$$

The above model is known as the "spiked covariance model" [34] and suggests a solution for estimating v from the noisy observations $d_1, \ldots, d_n$. For signal covariance estimation, the eigenvalues below $\sigma^2(1 + \sqrt{\gamma})^2$ are set to zero while those above are shrunk by an appropriate amount. A first approach would be to invert (32) to obtain

$$(34) \qquad \ell(\lambda, \gamma) = \frac{1}{2}\left(\lambda + \sigma^2(1 - \gamma) + \sqrt{(\lambda + \sigma^2(1 - \gamma))^2 - 4\sigma^2\lambda}\right) - \sigma^2$$

and replace a given eigenvalues λ by $\ell(\lambda, \gamma)$. However, this does not necessarily account for the error introduced by the inaccurate estimation of the eigenvectors as described by (33). A more appropriate approach was introduced by Donoho et al. [18] and succeeds quite well in recovering covariance matrices in high-dimensional data. Given a sample covariance matrix, we calculate its eigendecomposition

$$(35) \qquad \frac{1}{n} \sum_{s=1}^{n} d_s d_s^{\mathrm{T}} = \sum_{m=1}^{p} \lambda_m u_m u_m^{\mathrm{T}},$$

where $\{v_1, \ldots, v_p\}$ form an orthonormal basis. The eigenvalues $\{\lambda_1, \ldots, \lambda_p\}$ are transformed using a shrinker ρ into $\{\rho(\lambda_1), \ldots, \rho(\lambda_p)\}$. Putting everything back together gives us

$$(36) \qquad \sum_{m=1}^{p} \rho(\lambda_m) u_m u_m^{\mathrm{T}},$$

which is an estimator for the population covariance $\mathrm{Cov}[a]$.

Donoho et al. showed how to choose the shrinker ρ to optimize the error with respect to some loss function on the covariance estimator. The shrinker which achieves the lowest expected loss in the Frobenius norm as $n, p \to \infty$ is given by

$$(37) \qquad \rho(\lambda) = \ell(\lambda, \gamma)c(\ell(\lambda, \gamma), \gamma).$$

Among all shrinkage estimators of the form (36), the shrinker ρ given by (37) provides the lowest expected loss in the Frobenius norm [18] as $n, p \to \infty$. The expected loss with respect to the operator norm is minimized by $\rho(\lambda) = \ell(\lambda, \gamma)$. The authors consider a variety of norms, each of which is assigned a corresponding optimal shrinker [18]. Since our least-squares objective is formulated with respect to the Frobenius norm, we shall use the corresponding shrinker (37) in the following. Note that this estimator is not restricted to rank-one signals a but is optimal for arbitrary rank.

By slight abuse of notation, we extend the action of ρ from scalars to symmetric matrices by its action on the eigenvalues, so that

$$(38) \qquad \rho(A) := \rho\left(\sum_{m=1}^{p} \lambda_m v_m v_m^{\mathrm{T}}\right) = \sum_{i=1}^{p} \rho(\lambda_m)v_m v_m^{\mathrm{T}},$$

provided $\sum_{m=1}^{p} \lambda_m v_m v_m^{\mathrm{T}}$ is an eigendecomposition of A.

4.4. Shrinkage of B_n. We can apply the above ideas to obtain a better estimate for the right-hand side B_n. This is the first major improvement over previous work on least-squares estimators for 3D covariance matrices in cryo-EM [35, 3]. Specifically, we will replace B_n by

$$(39) \qquad B_n^{(\mathrm{s})} = A_n^{1/2}\rho(A_n^{-1/2}B_n A_n^{-1/2} + \sigma^2 \mathrm{I}_p)A_n^{1/2},$$

where A_n is given by (21).

In the clean data case, $\sigma = 0$ so $y_s = P_s x_s$. According to the definition of B_n (25), the clean right hand side equals

$$(40) \qquad B_n^{(\mathrm{c})} = \frac{1}{n}\sum_{s=1}^{n} P_s^{\mathrm{T}}(y_s - P_s\mu_n)(y_s - P_s\mu_n)^{\mathrm{T}}P_s,$$

Plugging in our forward model (12) for y_s, we get

$$(41) \qquad B_n^{(\mathrm{c})} = \frac{1}{n}\sum_{s=1}^{n} P_s^{\mathrm{T}}P_s(x_s - \mu_n)(x_s - \mu_n)^{\mathrm{T}}P_s^{\mathrm{T}}P_s,$$

which is the sample covariance of the mean-subtracted and projected-backprojected volumes $P_s^{\mathrm{T}}P_s x_s$ for $s = 1, \ldots, n$.

We would now like to obtain something close to (41) also in the noisy case. Mean-subtracting and backprojecting noisy images y_s, we have

$$(42) \qquad P_s^{\mathrm{T}}(y_s - P_s\mu_n) = P_s^{\mathrm{T}}P_s(x_s - \mu_n) + P_s^{\mathrm{T}}e_s.$$
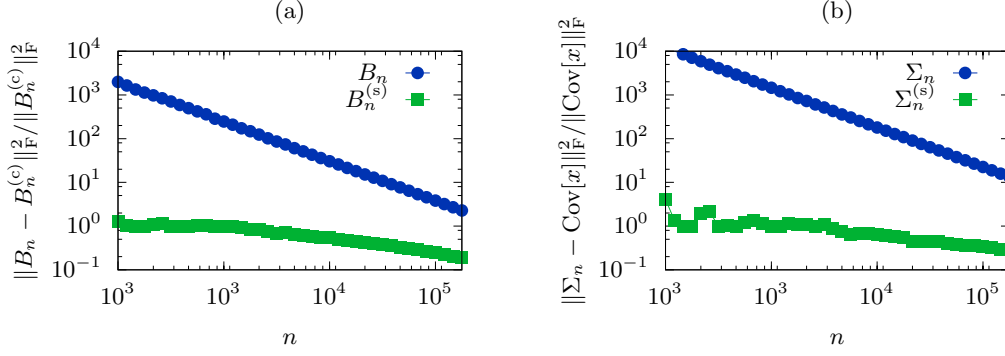
Figure 3. *Effect of shrinkage on the relative error of (a) B_n and (b) Σ_n for simulated data with $N = 16$, $C = 2$ and 7 distinct CTFs.*

This is similar to the spiked covariance model (30), except the noise term is not white but has covariance $\mathbb{E}[P^{\mathrm{T}}P]$. We can approximate the noise covariance using

$$(43) \qquad A_n = \frac{1}{n} \sum_{s=1}^{n} P_s^{\mathrm{T}} P_s,$$

since $\mathbb{E}[A_n] = \mathbb{E}[P^{\mathrm{T}}P]$ and the law of large numbers guarantees that A_n converges to its expectation as $n \to \infty$. Note that the high-dimensional phenomena studied in the previous section do not appear here since P is sampled from $\mathrm{SO}(3)$, which is of fixed dimension.

Multiplying the backprojected images by $A_n^{-1/2}$ whitens the noise, and we define

$$(44) \qquad z_s = A_n^{-1/2} P_s^{\mathrm{T}} (y_s - P_s \mu_n) = A_n^{-1/2} P_s^{\mathrm{T}} P_s (x_s - \mu_n) + A_n^{-1/2} P_s^{\mathrm{T}} e_s.$$

We now apply the standard shrinkage operator ρ defined in (37) and obtain

$$(45) \qquad \rho \left(\frac{1}{n} \sum_{s=1}^{n} z_s z_s^{\mathrm{T}} \right) = \rho \left(A_n^{-1/2} B_n A_n^{-1/2} + \sigma^2 \mathrm{I}_p \right).$$

Conjugating the shrunken covariance by $A_n^{1/2}$, we obtain a shrinkage variant $B_n^{(\mathrm{s})}$ of B_n as

$$(46) \qquad B_n^{(\mathrm{s})} := A_n^{1/2} \rho \left(A_n^{-1/2} B_n A_n^{-1/2} + \sigma^2 \mathrm{I}_p \right) A_n^{1/2},$$

providing an estimator of the clean right-hand side $B_n^{(\mathrm{c})}$. Note that the loss is minimized with respect to the Frobenius norm on $A_n^{-1/2} B_n^{(\mathrm{s})} A_n^{-1/2}$, not $B_n^{(\mathrm{s})}$, so there still might be some room for improvement. This is the subject of future work.

Replacing B_n by $B_n^{(\mathrm{s})}$ in (23) yields a more accurate estimator $\Sigma_n^{(\mathrm{s})} = L_n^{-1}(B_n^{(\mathrm{s})})$ since $B_n^{(\mathrm{s})}$ is a more accurate estimate of $B_n^{(\mathrm{c})}$ compared to B_n. We will refer to $\Sigma_n^{(\mathrm{s})}$ as the shrinkage covariance estimator, in contrast to the standard Σ_n least-squares covariance estimator. To evaluate the effect of the shrinkage on estimation accuracy, we plot the error of B_n and $B_n^{(\mathrm{s})}$ with respect to the clean $B_n^{(\mathrm{c})}$ as a function of n in Figure 3(a). For the values of n considered, shrinkage introduces a significant reduction in error. The same effect is found for the resulting estimators Σ_n and $\Sigma_n^{(\mathrm{s})}$ when compared with $\mathrm{Cov}[x]$ in Figure 3(b).

5. Efficient Computation. The covariance estimators as formulated in the previous section are computed by solving the corresponding normal equations. However, direct matrix inversion is intractable for typical problem sizes. We therefore consider an iterative solution based on the conjugate gradient method applied to a convolutional formulation of the normal equations. We first illustrate this for the mean least-squares estimator μ_n and then generalize this technique to the covariance estimator Σ_n. To speed up convergence, we employ circulant preconditioners for both A_n and L_n. The use of the conjugate gradient method to estimate the covariance was previously considered in Andén et al. [3], but this work lacked the convolutional formulation and appropriate preconditioners necessary for rapid convergence.

5.1. Mean deconvolution. The normal equations (20) for the mean estimator μ_n can be solved by calculating the matrix A_n, the right-hand side b_n, and solving for μ_n in $A_n \mu_n = b_n$. Solving this linear system is prohibitively expensive for typical problem sizes, so a more sophisticated approach is needed.

The computational complexity of calculating μ_n can be reduced by exploiting the convolutional structure of A_n. This observation has been exploited in several related applications previously [82, 23, 29, 79]. We shall focus on its incarnation in the Fourier-based iterative reconstruction method (FIRM) introduced by Wang et al. [83]. This section will rederive that method with the goal of applying these ideas to the computation of Σ_n.

We first note that the projection-backprojection operator $P_s^{\mathrm{T}} P_s$ is factored into $Q^{\mathrm{T}} I_s^{\mathrm{T}} I_s Q$, where $I_s : \mathbb{R}^{N^3} \to \mathbb{R}^{N^2}$ is the voxel projection mapping corresponding to P_s. In the voxel domain, $I_s^{\mathrm{T}} I_s$ is a convolution. Indeed, in the continuous case, projection integrates along a certain viewing direction, while backprojection "fills up" a volume along a certain viewing angle using an image. The resulting volume is then constant along that viewing direction. Projection followed by backprojection is therefore a low-pass filter of infinite support along that viewing direction. In the frequency domain, the Fourier slice theorem tells the same story. Projection is restriction of the volume Fourier transform to a plane while backprojection inserts a two-dimensional Fourier transform into that plane in a a three-dimensional Fourier transform, filling the rest with zeroes. The result is a multiplication of the original volume by a Dirac delta function supported on that plane.

Since we have engineered our voxel discretization I_s of the projection operator to satisfy a discrete Fourier slice theorem (10), these properties carry over to the discrete case. In particular, we have that

$$(47) \qquad I_s^{\mathrm{T}} I_s x(\boldsymbol{i}) = x * k_s(\boldsymbol{i}) = \sum_{\boldsymbol{j} \in M_{2N-1}^3} x(\boldsymbol{i} - \boldsymbol{j}) k_s(\boldsymbol{j}),$$

where $*$ denotes convolution and

$$(48) \qquad k_s(\boldsymbol{i}) := \frac{1}{N^4} \sum_{\boldsymbol{l} \in M_{N-1}^2} |\mathcal{F} h_s(\boldsymbol{l})|^2 \mathrm{e}^{\frac{2\pi \mathrm{i}}{N} \langle \boldsymbol{i}, R_s^{\mathrm{T}}[\boldsymbol{l};0] \rangle},$$

for all $\boldsymbol{i} \in M_{2N-1}^3$. This follows from calculating the dual I_s^{T} of the voxel projection matrix (9) and applying it to the matrix I_s itself. In other words, $I_s^{\mathrm{T}} I_s$ is a 3-Toeplitz matrix. The projection-backprojection operator $P_s^{\mathrm{T}} P_s$ is thus factored into a basis evaluation Q, a convolution $I_s^{\mathrm{T}} I_s$, and a basis expansion Q^{T}.

In the definition (21) of A_n, plugging in (47) now gives

$$(49) \qquad A_n x = \frac{1}{n} \sum_{s=1}^{n} P_s^{\mathrm{T}} P_s x = \frac{1}{n} \sum_{s=1}^{n} Q^{\mathrm{T}}(Qx * k_s) = Q^{\mathrm{T}} \left(Qx * \frac{1}{n} \sum_{s=1}^{n} k_s \right) = Q^{\mathrm{T}}(Qx * f_n),$$

where

$$(50) \qquad f_n := \frac{1}{n} \sum_{s=1}^{n} k_s.$$

The matrix A_n is therefore factorized as basis evaluation Q, followed by application of a 3-Toeplitz matrix, then a basis expansion Q^{T}.

The convolution kernel f_n can be calculated in one pass over the dataset using NUFFTs with complexity $\mathcal{O}(N^3 \log N + nN^2)$. Once this has been calculated, however, each application of A_n using the convolutional formulation of (49) is achieved in $\mathcal{O}(N^3 \log N)$ time using FFTs, independent of the number of images.

We can exploit this fast application of A_n to solve the system $A_n \mu_n = b_n$. Specifically, we apply the conjugate gradient (CG) method, which is an iterative algorithm for solving linear systems [31]. It computes an approximate solution at each iteration through a single application of A_n, so its efficiency depends on being able to apply this operator fast, which is the case for Toeplitz operators, as seen above [9]. To reach a given accuracy $\mathcal{O}(\sqrt{\kappa(A_n)})$ iterations are needed, where $\kappa(A_n)$ is the condition number of A_n. As we shall see, $\kappa(A_n)$ can be reduced by the circulant preconditioner described in Section 5.3.

The above algorithm therefore consists of two steps. First, we precalculate the kernel f_n and the right-hand side b_n. These are both achieved in $\mathcal{O}(N^3 \log N + nN^2)$ using NUFFTs. Second, $\mathcal{O}(\sqrt{\kappa(A_n)})$ iterations CG are performed, each at a cost of $\mathcal{O}(N^3 \log N)$. The overall complexity is then $\mathcal{O}(\sqrt{\kappa(A_n)} N^3 \log N + nN^2)$. Note that this method is nearly optimal in the sense that simply reading the images requires $\mathcal{O}(nN^2)$ operations, while the reconstructed volume takes up $\mathcal{O}(N^3)$ in memory.

5.2. Covariance deconvolution. As discussed in the previous section, directly solving the normal equations for μ_n can be computationally expensive. This is also the case for the covariance estimator Σ_n, which scales worse in N. Indeed, volume vectors x are of size N^3 so the covariance estimate Σ_n is of size N^3-by-N^3 and thus contains N^6 elements. Since L_n maps covariance matrices to covariance matrices, the matrix representation of L_n requires $(N^6)^2 = N^{12}$ elements, which stored at single precision occupies 256 GB for $N = 8$. Direct inversion of this matrix would have computational complexity $\mathcal{O}(N^{18})$.

A more practical approach is to apply the ideas from the mean estimation algorithm described in the previous section. Note that L_n is defined in (24) as a sum of linear matrix operators

$$(51) \qquad \Sigma \mapsto P_s^{\mathrm{T}} P_s \Sigma P_s^{\mathrm{T}} P_s.$$

Since $P_s^{\mathrm{T}} P_s$ can be factored into basis evaluation/expansion and convolution in three dimensions, this mapping enjoys a similar factorization

$$(52) \qquad P_s^{\mathrm{T}} P_s \Sigma P_s^{\mathrm{T}} P_s = Q^{\mathrm{T}} \left(I_s^{\mathrm{T}} I_s \left(Q \Sigma Q^{\mathrm{T}} \right) I_s^{\mathrm{T}} I_s \right) Q.$$

The conjugation by $I_s^{\mathrm{T}} I_s$ convolves both the rows and the columns of the matrix by k_s, which can be written as a convolution in six dimensions by the outer product of k_s with itself. Specifically, we have

$$I_s^{\mathrm{T}} I_s Z I_s^{\mathrm{T}} I_s = Z * K_s, \tag{53}$$

for any matrix $Z \in \mathbb{R}^{N^3 \times N^3}$, where

$$K_s[\boldsymbol{i}_1, \boldsymbol{i}_2] \coloneqq k_s[\boldsymbol{i}_1] k_s[\boldsymbol{i}_2], \tag{54}$$

for all $(\boldsymbol{i}_1, \boldsymbol{i}_2) \in M_{2N-1}^6$. Consequently

$$P_s^{\mathrm{T}} P_s \Sigma P_s^{\mathrm{T}} P_s = Q^{\mathrm{T}} \left(Q \Sigma Q^{\mathrm{T}} * K_s \right) Q. \tag{55}$$

One advantage of this formulation is that we average the convolutional kernels over the whole dataset to obtain a convolutional representation for L_n. This gives

$$L_n(\Sigma) = Q^{\mathrm{T}} \left(Q \Sigma Q^{\mathrm{T}} * F_n \right) Q, \tag{56}$$

where

$$F_n \coloneqq \frac{1}{n} \sum_{s=1}^{n} K_s. \tag{57}$$

Similar to A_n, L_n is factored into basis evaluations/expansions and a 6-Toeplitz matrix operator, allowing for rapid calculation of $L_n(\Sigma)$.

The kernel F_n is calculated using an NUFFT at a computational cost of $\mathcal{O}(N^6 \log N + n N^4)$ by rewriting (57) as the six-dimensional non-uniform discrete Fourier transform of the N^2-by-N^2-by-n array formed by the outer products of $\mathcal{F} h_s$ sampled on a two-dimensional M_{N-1}^2 grid. Once this kernel is computed, applying L_n using the convolution formulation (56) costs $\mathcal{O}(N^6 \log N)$. This follows because the basis evaluations/expansions each have complexity $\mathcal{O}(N^3 \log N)$ and there are $\mathcal{O}(N^3)$ rows and columns in the matrix, while the six-dimensional convolution achieves complexity $\mathcal{O}(N^6 \log N)$ using FFTs.

The right-hand side matrix B_n can also be computed as an NUFFT with complexity $\mathcal{O}(N^6 \log N + n N^4)$. As a result, the overall complexity for solving the normal equations (23) using CG is $\mathcal{O}(\sqrt{\kappa(L_n)} N^6 \log N + n N^4)$. We again note that this complexity is nearly optimal, since with respect to storing the $\mathcal{O}(N^6)$ size covariance matrix, we only lose a condition number and logarithmic factor, while we only require a N^2 factor increase with respect to storing the input images of size $\mathcal{O}(n N^2)$.

5.3. Circulant Preconditioners for A_n and L_n. The number of iterations required for CG to converge scales with the square root of the condition number of the linear system [61, 5, 73]. As we shall see, both A_n and L_n are badly conditioned due to geometric considerations and the influence of the CTFs. However, by properly preconditioning these operators, we can reduce the condition number to a manageable regime.

For a uniform distribution of viewing directions R over SO(3) and with no microscope aberration, that is $\mathcal{F} h(\omega) = 1$, the continuous version of A_n is approximated in the Fourier

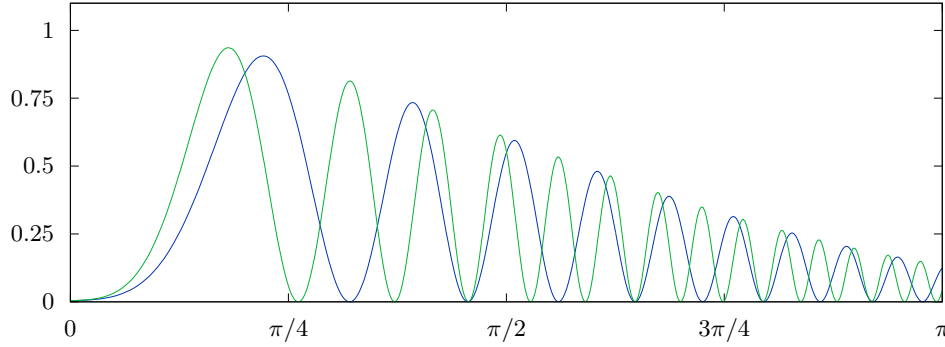**Figure 4.** *The square magnitudes $|\mathcal{F}h(\boldsymbol{\omega})|^2$ of two sample CTFs. Their sum forms a bandpass filter, worsening the conditioning of the least-squares estimators.*

domain by the filter $2/\|\boldsymbol{\omega}\|$ as $n \to \infty$ (e.g., see [53, 35]). Qualitatively, this is the limit of the Fourier transform $\mathcal{F}f_n$ of f_n as $N, n \to \infty$. As $\mathcal{F}f_n$ approaches this limit, the $1/\|\boldsymbol{\omega}\|$ decay results in worse conditioning of A_n. The influence of the CTFs do not improve this situation. Indeed, the square magnitudes of the CTFs $|\mathcal{F}h_s(\boldsymbol{\omega})|^2$ form a bandpass filter (see Figure 4), attenuating low and high frequencies, which is replicated in the Fourier transform of f_n, further worsening the conditioning of A_n.

Similar results hold for L_n. Indeed, for uniform distribution of viewing directions and $\mathcal{F}h(\omega) = 1$, it was shown by Katsevich et al. [35] that its continuous version acts as multiplication in the Fourier domain by $2/\|\boldsymbol{\omega}_1 \times \boldsymbol{\omega}_2\|$ as $n \to \infty$. Not only does this decay as $1/\|\boldsymbol{\omega}_1\|\|\boldsymbol{\omega}_2\|$, but the kernel is singular for $\boldsymbol{\omega}_1$ parallel to $\boldsymbol{\omega}_2$. Again, adding CTFs results in attenuation for low and high frequencies $\|\boldsymbol{\omega}_1\|$ and $\|\boldsymbol{\omega}_2\|$. As a result, the Fourier transform $\mathcal{F}F_n$ of the kernel F_n representing L_n is close to singular and decays rapidly, resulting in poor conditioning of L_n.

In addition to the above, we note that if the distribution of viewing directions is non-uniform, the condition number will be even larger. As a consequence of the above phenomena, a large number of iterations will be required in order to achieve convergence, which is not a desirable situation.

To remedy this, we precondition A_n and L_n. In other words, we define operators C_n and D_n that can be easily inverted and such that $C_n^{-1}A_n$ and $D_n^{-1}L_n$ are close to the identity. As a result, $\kappa(C_n^{-1}A_n)$ and $\kappa(D_n^{-1}L_n)$ are small, allowing us to reformulate the linear systems to achieve better conditioning. This is the idea of the preconditioned CG method, which converges in $\mathcal{O}(\sqrt{\kappa(C_n^{-1}A_n)})$ and $\mathcal{O}(\sqrt{\kappa(D_n^{-1}L_n)})$ steps for the mean and covariance estimators, respectively [61, 5, 73]. Note that the overall conditioning of the problem is still the same, it is only the CG convergence rate that changes.

A variety of preconditioners have been developed to improve convergence of the CG method. Popular alternatives include diagonal, or Jacobi, preconditioners and incomplete Cholesky or LU factorizations [61, 5]. As mentioned above, a good preconditioner is one whose inverse closely approximates the operator itself. This is commonly achieved by exploiting the structure of the operator. For example, a Toeplitz operator is well approximated by

a circulant operator. The inverse of a circulant operator can in turn be calculated efficiently using FFTs [71].

In the current work, we therefore consider circulant approximations to the convolution factors of A_n and L_n as preconditioners. A circulant operator C operating on some d-dimensional vector w defined on M_N^d is given by a circular convolution

$$(58) \qquad Cw(\boldsymbol{i}) = \sum_{\boldsymbol{j} \in M_N^d} w^{\mathrm{per}}(\boldsymbol{i} - \boldsymbol{j})g(\boldsymbol{j}),$$

where w^{per} is the periodized extension of w

$$(59) \qquad w^{\mathrm{per}}(\boldsymbol{i}) = w(\boldsymbol{i} + N\boldsymbol{m}) \quad \text{for } \boldsymbol{m} \text{ such that } \boldsymbol{i} + N\boldsymbol{m} \in M_N^d,$$

and g is a convolution kernel defined on M_N^d. The difference with the standard convolution encountered before in A_n and L_n (see (49) and (56)) is that the functions are periodized at the boundary instead of padded with zeroes.

An important advantage of circulant operators is that they are diagonalized by the Fourier basis, and we can therefore write the action of C in the Fourier domain as

$$(60) \qquad \mathcal{F}Cw(\boldsymbol{k}) = \mathcal{F}w(\boldsymbol{k})\mathcal{F}g(\boldsymbol{k}).$$

This makes circulant operators fast to apply, but also fast to invert, since

$$(61) \qquad \mathcal{F}C^{-1}w(\boldsymbol{k}) = \mathcal{F}w(\boldsymbol{k})\mathcal{F}g(\boldsymbol{k})^{-1},$$

provided $\mathcal{F}g(\boldsymbol{k}) \neq 0$ for all $\boldsymbol{k} \in M_N^d$. Since standard convolutions and circular convolutions differ principally at the boundary, they are similar when both functions w and g are concentrated around the origin.

Circulant operators are therefore good preconditioners to standard convolutions, provided we can calculate them efficiently. Let us define C_n to be a circulant operator with associated kernel g_n such that

$$(62) \qquad \widetilde{C}_n := \operatorname*{argmin}_{\widetilde{C}_n \text{ circ.}} \|\widetilde{A}_n - \widetilde{C}_n\|_{\mathrm{F}},$$

and

$$(63) \qquad \widetilde{A}_n = \frac{1}{n} \sum_{s=1}^{n} I_s^{\mathrm{T}} I_s,$$

is the voxel version of A_n such that $\widetilde{A}_n v = v * f_n$. Such approximations have been previously studied by Tyrtyshnikov [76], from whose results we derive the formula

$$(64) \qquad g_n(\boldsymbol{i}) := \frac{1}{N^3} \sum_{\substack{\boldsymbol{m} \in M_{2N-1}^3 \\ \boldsymbol{m} = \boldsymbol{i} \ (\mathrm{mod}\ N)}} (N - |m_1|)(N - |m_2|)(N - |m_3|)f_n(\boldsymbol{m}),$$

for all $\boldsymbol{i} \in M_N^3$. This periodizes the original kernel f_n with periodicity N and weights by a multiplier that attenuates points far from the origin.

This particular circulant approximation $\widetilde{C}_n$ of the Toeplitz operator $\widetilde{A}_n$ has the advantage of being computed with low computational complexity. Indeed, calculating g_n using (64) has complexity $\mathcal{O}(N^3)$. Furthermore, it is optimal in the sense of minimizing the distance in Frobenius norm to $\widetilde{A}_n$ and it preserves the positive semidefiniteness and invertibility of $\widetilde{A}_n$, which is not the case for other circulant preconditioners, such as those proposed by Strang and Chan [71, 10].

Since the circulant approximation $\widetilde{C}_n$ can be inverted easily using three-dimensional FFTs, we use $C_n^{-1} := Q^{\mathrm{T}} \widetilde{C}_n^{-1} Q$ as a preconditioner in the CG method when solving $A_n \mu_n = b_n$. In this case, the effective condition number of the preconditioned linear system is equal to $\kappa(C_n^{-1} A_n)$, which is small if the approximation is accurate. As we shall see in Section 7.1, numerical experiments indicate that this preconditioner brings the condition number down to 1–50.

The same type of circulant approximation can also be found for the convolution factor in L_n, with the circulant operator

$$
(65) \qquad\qquad \widetilde{D}_n := \operatorname*{argmin}_{\widetilde{D}_n \text{ circ.}} \|\widetilde{L}_n - \widetilde{D}_n\|_{\mathrm{F}},
$$

where

$$
(66) \qquad\qquad \widetilde{L}_n(Z) = \frac{1}{n} \sum_{s=1}^{n} I_s^{\mathrm{T}} I_s Z I_s^{\mathrm{T}} I_s = Z * F_n
$$

is the voxel version of L_n. The kernel G_n of $\widetilde{D}_n$ can be found to equal

$$
(67) \qquad\qquad G_n(\boldsymbol{i}) := \frac{1}{N^6} \sum_{\substack{\boldsymbol{m} \in M_{2N-1}^6 \\ \boldsymbol{m} = \boldsymbol{i} \ (\mathrm{mod}\ N)}} (N - |m_1|) \cdots (N - |m_6|) F_n(\boldsymbol{m}),
$$

for all $\boldsymbol{i} \in M_N^6$. Again, G_n is a weighted periodization of F_n. The computational complexity of calculating G_n is $\mathcal{O}(N^6)$.

The circulant operator $\widetilde{D}_n$ can also be inverted quickly using six-dimensional FFTs, so we use $\Sigma \mapsto D_n^{-1}(\Sigma) := Q^{\mathrm{T}} \widetilde{D}_n^{-1}(Q\Sigma Q^{\mathrm{T}})Q$ to precondition the normal equations $L_n(\Sigma_n) = B_n$ of the least-squares covariance estimator or $L_n(\Sigma_n) = B_n^{(\mathrm{s})}$ for the shrinkage variant. For the same reasons as in the mean estimation case, the condition number $\kappa(D_n^{-1} L_n)$ is expected to be small. Again, numerical simulations in Section 7.1 indicate that the condition number of this operator also stays in the regime 1–200 when we apply the basis projection described in the next section.

5.4. Conjugate Gradient & Thresholding. We are now ready to formulate the algorithms for estimating the volume mean and covariance matrix given the input images $y_1, \ldots, y_n$ and projection mappings $P_1, \ldots, P_n$ (or equivalently, the viewing directions $R_1, \ldots, R_n$ and CTFs $\mathcal{F}h_1, \ldots \mathcal{F}h_n$).

Algorithm 1 The least-squares mean estimator μ_n

> **function** MEANESTIMATION($\{R_s\}_{s=1}^n$, $\{h_s\}_{s=1}^n$, $\{y_s\}_{s=1}^n$, Q)
>
> Set $f_n[\boldsymbol{i}] \leftarrow \frac{1}{nN^4} \sum_{s=1}^n \sum_{\boldsymbol{l} \in M_{N-1}^2} |\mathcal{F}h_s(\boldsymbol{l})|^2 \mathrm{e}^{\frac{2\pi \mathrm{i}}{N}\langle \boldsymbol{i}, R_s^{\mathrm{T}}[\boldsymbol{l};0]\rangle}$
>
> Set $g_n[\boldsymbol{i}] \leftarrow \frac{1}{N^3} \sum_{\substack{\boldsymbol{m} \in M_{2N-1}^3 \\ \boldsymbol{m} = \boldsymbol{i} \,(\mathrm{mod}\ N)}} (N - |m_1|)(N - |m_2|)(N - |m_3|)f_n(\boldsymbol{m}) \quad \forall \boldsymbol{i} \in M_N^3$
>
> Calculate $\overline{g}_n$ such that $\mathcal{F}\overline{g}_n = (\mathcal{F}g_n)^{-1}$
>
> Set $b_n \leftarrow \frac{1}{n} \sum_{s=1}^n P_s^{\mathrm{T}} y_s$
>
> Apply CG to $Q^{\mathrm{T}}(Q\mu_n * f_n) = b_n$ with preconditioner $x \mapsto Q^{\mathrm{T}}(Qx * \overline{g}_n)$
>
> **return** μ_n
>
> **end function**

The mean estimation algorithm is given in Algorithm 1. First, the convolutional kernel f_n associated with A_n and the right-hand side b_n are computed at a cost of $\mathcal{O}(N^3 \log N + nN^2)$. The circulant approximation kernel g_n is then calculated from f_n, which takes $\mathcal{O}(N^3)$. We then apply CG to (20), with each iteration requiring application of A_n and C_n^{-1} which are obtained by multiplications by Q, Q^{T} and convolutions by f_n and $\overline{g}_n$, all of which have computational complexity of $\mathcal{O}(N^3 \log N)$. After $\mathcal{O}(\sqrt{\kappa(C_n^{-1}A_n)})$ iterations, we have μ_n. The overall computational complexity of Algorithm 1 is then

$$(68) \qquad O\left(\sqrt{\kappa(C_n^{-1}A_n)}N^3 \log N + nN^2\right),$$

where $\kappa(C_n^{-1}A_n)$ is typically in the range 1–50.

Algorithm 2 The covariance estimators Σ_n (*do_shrink* = false) and $\Sigma_n^{(\mathrm{s})}$ (*do_shrink* = true)

> **function** COVARIANCEESTIMATION($\{R_s\}_{s=1}^n$, $\{h_s\}_{s=1}^n$, $\{y_s\}_{s=1}^n$, μ_n, σ, Q, *do_shrink*)
>
> Set $f_n[\boldsymbol{i}_1, \boldsymbol{i}_2] \leftarrow \frac{1}{nN^8} \sum_{s=1}^n \sum_{\boldsymbol{l}_1, \boldsymbol{l}_2 \in M_{N-1}^2} |\mathcal{F}h_s(\boldsymbol{l}_1)|^2 |\mathcal{F}h_s(\boldsymbol{l}_2)|^2 \mathrm{e}^{\frac{2\pi \mathrm{i}}{N}\left(\langle \boldsymbol{i}_1, R_s^{\mathrm{T}}[\boldsymbol{l}_1;0]\rangle - \langle \boldsymbol{i}_2, R_s^{\mathrm{T}}[\boldsymbol{l}_2;0]\rangle\right)}$
>
> Set $G_n[\boldsymbol{i}] \leftarrow \frac{1}{N^6} \sum_{\substack{\boldsymbol{m} \in M_{2N-1}^6 \\ \boldsymbol{m} = \boldsymbol{i} \,(\mathrm{mod}\ N)}} (N - |m_1|) \cdots (N - |m_6|)F_n(\boldsymbol{m}) \quad \forall \boldsymbol{i} \in M_N^6$
>
> Calculate $\overline{G}_n$ such that $\mathcal{F}\overline{G}_n = (\mathcal{F}G_n)^{-1}$
>
> Set $B_n \leftarrow \frac{1}{n} \sum_{s=1}^n P_s^{\mathrm{T}}(y_s - P_s\mu_n)(y_s - P_s\mu_n)^{\mathrm{T}} P_s - \sigma^2 P_s^{\mathrm{T}} P_s$
>
> **if** *do_shrink* **then**
>
> Set $B_n \leftarrow B_n^{(\mathrm{s})} = A_n^{1/2}\rho(A_n^{-1/2}B_nA_n^{-1/2} + \sigma^2 \mathrm{I}_p)A_n^{1/2}$
>
> **end if**
>
> Apply CG to $Q^{\mathrm{T}}(Q\Sigma_nQ^{\mathrm{T}} * F_n) = B_n$ with preconditioner $X \mapsto Q^{\mathrm{T}}(QXQ^{\mathrm{T}} * \overline{G}_n)Q^{\mathrm{T}}$
>
> **return** Σ_n
>
> **end function**

The covariance estimation method listed in Algorithm 2 is qualitatively similar. The convolutional kernel F_n associated with L_n and the right-hand side matrix B_n are computed with complexity $\mathcal{O}(N^6 \log N + nN^4)$. The circulant kernel G_n is obtained at cost $\mathcal{O}(N^6)$. Applying L_n and D_n now involves multiplication by Q and Q^{T} as well as convolution with F_n

and G_n^{-1}, each of which has computational complexity $\mathcal{O}(N^6 \log N)$. Now $\mathcal{O}(\sqrt{\kappa(D_n^{-1}L_n)})$ iterations are needed to obtain Σ_n. The overall computational complexity of Algorithm 2 is then

$$
(69) \qquad O\left(\sqrt{\kappa(D_n^{-1}L_n)} N^6 \log N + n N^4 \right),
$$

where $\kappa(D_n^{-1}L_n)$ is in the range 1–200.

To obtain the shrinkage variant of the estimator, $\Sigma_n^{(s)}$, the additional step of calculating $B_n^{(s)}$ from B_n is added before the CG step in Algorithm 2. This is done using (46), where B_n is whitened by conjugation with $A_n^{-1/2}$, the whitened matrix is shrunk using ρ, and the result is unwhitened by conjugation with $A_n^{1/2}$. The number of top eigenvalues of $A_n^{-1/2}B_nA_n^{-1/2}$ which exceed the Marĕnko-Pastur threshold is typically small, so we can exploit Lanczos method for finding the top eigenvalues and eigenvectors. For this, we need to apply $A_n^{-1/2}B_nA_n^{-1/2}$ fast. Since A_n can be applied fast using its convolution kernel, applying its inverse square root $A_n^{-1/2}$ to a volume can be approximated iteratively using Krylov subspace methods [19, 60, 32]. The number of iterations needed is typically small, so we take its complexity to be $\mathcal{O}(N^3 \log N)$. As a result, applying $A_n^{-1/2}B_nA_n^{-1/2}$ has complexity $O(N^6)$, since matrix multiplication by B_n takes $\mathcal{O}(N^6)$. The overall eigendecomposition calculation therefore has complexity $\mathcal{O}(N^6)$, where we have assumed that the number of non-trivial eigenvectors is taken to be constant.

We note that an alternative to Krylov subspace methods for approximating $A_n^{-1/2}$ is to exploit the Toeplitz structure in A_n and use this to calculate its Cholesky factors, which have similar properties to the matrix square root but can be inverted efficiently. In one dimension, this can be done in $\mathcal{O}(N^2)$ using the Schur algorithm [64, 52, 1]. While this has been generalized to matrices of block Toeplitz structure [1], these do not take into account 2-Toeplitz structure, also known as Toeplitz-block-Toeplitz, and so have complexity $O(N^5)$ instead of the desired $O(N^4)$. Designing an appropriate generalization of the Schur algorithm for d-Toeplitz operators where $d \geq 2$ is the subject of future work.

Both in the case of the standard estimator and the shrinkage variant, the estimated covariance matrix Σ_n will contain a considerable amount of error in the form of a bulk noise distribution similar to that observed in the spiked covariance model. A final step of selecting the dominant eigenvectors is therefore necessary to extract the relevant part of the covariance matrix structure. Since we expect the population covariance matrix to be of low rank, it must have a small number of non-zero eigenvectors. This number can be estimated by looking for a "knee" in the spectrum of Σ_n where the dominant eigenvalues separate from the bulk noise distribution. In the case of a discrete distribution of molecular structures, this is at most one minus the number of resolved structures in the dataset. However, this determination has to be done manually. A heuristic method for validating this choice could be to inspect the corresponding eigenvectors and determine how "noise-like" they appear, using a suitable prior. Future work will focus on enabling the algorithm to perform this selection automatically. The computational complexity of calculating the leading $r = O(1)$ eigenvalues and eigenvectors of Σ_n is $O(rN^6)$.

5.5. Choice of Basis. To represent a volume x, we can store its values on the voxel grid M_N^3/N. We will call this a decomposition in the voxel basis. The problem with this basis is that the electric potential of a molecule is supported in the central ball $\{\|\boldsymbol{u}\| < 1\}$, with no energy in the "corners" of the cube $[-1,+1]^3$. Indeed, any energy in this region will be captured by projections along a subset of viewing angles and will not be well reconstructed. We can therefore safely assume that the support is contained in the central ball. The same holds in the frequency domain, where frequency samples outside the Nyquist ball $\{\|\boldsymbol{k}\| < N/4\}$ are expected to be negligible. In addition, the low sampling density of these frequencies leads to ill-conditioning of the reconstruction problem, which we would like to avoid.

To solve this, we will use different bases which are concentrated on $\{\|\boldsymbol{u}\| < 1\}$ in space and within $\{\|\boldsymbol{k}\| < N/4\}$ in frequency. One solution to this spectral concentration is given by the 3D Slepian functions [70], but their implementation is quite complicated. Instead, we will focus on an alternative basis with similar properties, the spherical Fourier-Bessel basis. It consists of functions given in spherical coordinates (r, θ, ϕ) by

$$(70) \qquad \phi_{\ell,k,m}(r,\theta,\phi) = \begin{cases} C_{\ell,k} j_\ell(rz_{\ell,k}) Y_{\ell,m}(\theta,\phi) & 0 \le r < 1 \\ 0 & 1 \le r \end{cases}$$

where j_ℓ is the spherical Bessel function of order ℓ, $z_{\ell,k}$ is the kth zero of j_ℓ, and $Y_{\ell,m}$ is the spherical harmonic function of order ℓ and degree m, and $C_{\ell,k} = \sqrt{2}|j_{\ell+1}(z_{\ell,k})|^{-1}$. The indices m and k satisfy $|m| \le \ell$ and $k \le k_{\max}(\ell)$, where $k_{\max}(\ell)$ is the largest integer such that $z_{\ell,k_{\max}(\ell)+1} < N\pi/4$. This is the same sampling criterion used in Bhamre et al. [8] and Cheng [11], which generalizes similar criteria for the 2D Fourier-Bessel basis [39, 85]. This condition on k ensures that $\mathcal{F}\phi$ is concentrated within the Nyquist ball since this function is concentrated around a ring centered at $\|\boldsymbol{k}\| = z_{\ell,k}/\pi$. Finally, the constant $C_{\ell,k}$ ensures that the basis functions have unit norm. For ℓ up to some $\ell_{\max}$, we therefore have the basis $\{\phi_{\ell,k,m}\}_{\ell \le \ell_{\max}, k \le k_{\max}(\ell), |m| \le \ell}$ which we use to decompose x.

However, as discussed in Section 2, the standard voxel basis allows for fast projection through using NUFFTs. To take advantage of this, we need a fast change-of-basis mapping between the voxel basis and the spherical Fourier-Bessel basis. For this, we can use NUFFTs and separation of variables to evaluate the basis at voxel grid points in $\mathcal{O}(N^4)$ complexity. Using fast spherical harmonic transforms [59, 74, 75, 41] and fast Fourier-Bessel transforms [49], we can reduce this further to $\mathcal{O}(N^3 \log N)$.

A simpler alternative is provided by the truncated Fourier basis

$$(71) \qquad \phi_{\boldsymbol{k}}(x) = \begin{cases} C\mathrm{e}^{2\pi i \langle x, \boldsymbol{k} \rangle} & x \in M_{N-2}^3/N \\ 0 & \text{otherwise} \end{cases}$$

for $\boldsymbol{k} \in M_{N-2}^3 \cap \{\|\boldsymbol{\omega}\| < (N-2)/2\}$. Again, C is chosen so that $\phi_{\boldsymbol{k}}$ has unit norm for all $\boldsymbol{k}$. The functions are zero outside a central box M_{N-2}, providing a padding of one voxel in each direction, and only consists of frequencies inside the Nyquist ball. While less concentrated compared to the spherical Fourier-Bessel basis, it has the advantage of providing efficient change-of-basis mappings through standard FFTs.

In the following, we will use the spherical Fourier-Bessel basis since it enjoys better concentration properties. We note, however, that for large values of N, it may be more computationally efficient to use the truncated Fourier basis. This is due to the fact that the constant

associated with FFTs is considerably smaller than those related to fast spherical harmonic and Fourier-Bessel transforms.

6. Reconstruction of States. Having estimates μ_n and Σ_n of the mean $\mathbb{E}[x]$ and covariance $\text{Cov}[x]$ provides us with partial information on the distribution of the volumes $x_1, \ldots, x_n$. However, unless the distribution of x is Gaussian, this is not enough to fully characterize it. To do so, more information has to be extracted from the images $y_1, \ldots, y_n$. We shall consider two types of singular distributions: those supported on a finite number of points and those supported on a continuous low-dimensional manifold.

6.1. Wiener Filter. For a fixed viewing direction R, the variability in the volume x encoded by the 3D covariance $\text{Cov}[x]$ induces variability in the clean images Px through the 2D covariance $P\text{Cov}[x]P^{\text{T}}$, that is, the projected 3D covariance. Classical Wiener filtering leverages this 2D covariance to denoise images or estimate the underlying volume corresponding to each image.

Recall that we have the image formation model (12), restated here as

$$y_s = P_s x_s + e_s. \tag{72}$$

As we saw earlier, this induces the relation

$$\text{Cov}_{x_s|P_s}[P_s x_s] = P_s \text{Cov}[x] P_s^{\text{T}}, \tag{73}$$

for the signal term, which allows us to estimate the image covariance as

$$\text{Cov}_{x_s|P_s}[P_s x_s] \approx P_s \Sigma_n P_s^{\text{T}}. \tag{74}$$

The noise covariance $\text{Cov}[e_s]$ is assumed to be $\sigma^2 \text{I}_{N^2}$.

We now use the estimated mean and covariance to define a Wiener filter estimator [47, 36]

$$\widehat{x}_s \coloneqq H_s(y_s - P_s\mu_n) + \mu_n, \tag{75}$$

of x_s where

$$H_s \coloneqq \Sigma_{n,r} P_s^{\text{T}} (P_s \Sigma_{n,r} P_s^{\text{T}} + \sigma^2 \text{I}_{N^2})^{-1}. \tag{76}$$

If $\Sigma_{n,r} = \text{Cov}[x]$ and $\mu_n = \mathbb{E}[x]$, this linear filter minimizes the expected mean-squared error

$$\mathbb{E}_{x_s,e_s|P_s} \|\widehat{x}_s - x_s\|^2. \tag{77}$$

In the finite-sample case, this no longer holds, but we can expect the Wiener filter to perform better than a direct pseudo-inverse calculation as our mean and covariance estimates become more accurate.

As discussed in the previous section, Σ_n is often of low rank r following the final thresholding step, giving $\Sigma_{n,r}$. We can use this to significantly reduce the complexity of calculating $\widehat{x}_s$, which through direct evaluation of (76) takes $\mathcal{O}(N^6 \log N)$ operations since it involves calculating $P_s Q \Sigma_n Q^{\text{T}} P_s^{\text{T}}$. Let

$$O_s U_s = P_s V_{n,r} \tag{78}$$

be the "thin" QR decomposition of $P_s V_{n,r}$, where O_s is an N^2-by-r orthonormal matrix and U_s is an r-by-r upper triangular matrix [73, 26]. Using these matrices, we rewrite $\widehat{x}_s$ as

$$(79) \qquad \widehat{x}_s = V_{n,r}\Lambda_{n,r}U_s^{\mathrm{T}}(U_s\Lambda_{n,r}U_s^{\mathrm{T}} + \sigma^2 \mathrm{I}_r)^{-1}O_s^{\mathrm{T}}(y_s - P_s\mu_n) + \mu_n,$$

for $s = 1, \ldots, n$. This shows that $\widehat{x}_s$ equals μ_n plus a linear combination of the vectors $V_{n,r}$. A more compact, but isometrically equivalent, representation is therefore to keep only the coordinates

$$(80) \qquad \widehat{\alpha}_s := V_{n,r}^{\mathrm{T}}(\widehat{x}_s - \mu_n) = \Lambda_{n,r}U_s^{\mathrm{T}}(U_s\Lambda_{n,r}U_s^{\mathrm{T}} + \sigma^2 \mathrm{I}_r)^{-1}O_s^{\mathrm{T}}(y_s - P_s\mu_n).$$

These coordinates can be calculated in $\mathcal{O}(rN^3 \log N + nr^2 N^2)$, since the projections $P_s V_{n,r}$ and $P_s \mu_n$ require $\mathcal{O}(rN^3 \log N + nrN^2)$, the QR decompositions have computational complexity $\mathcal{O}(nr^2 N^2)$ and inverting n r-by-r matrices takes $\mathcal{O}(nr^3)$, where we assume that $N^2 \gg r$. We note that this is close to optimal, since r eigenvolumes require $\mathcal{O}(rN^3)$ in storage, while the images are stored in $\mathcal{O}(nN^2)$, so we only lose a factor of $\log N$ and r^2, respectively.

The Wiener filter estimate of the volumes is now

$$(81) \qquad \widehat{x}_s = V_{n,r}\widehat{\alpha}_s + \mu_n.$$

The traditional denoising Wiener filter of the 2D images is obtained by projecting these volume estimates. Specifically, we define

$$(82) \qquad \widehat{y}_s := P_s\widehat{x}_s = P_s(V_{n,r}\widehat{\alpha}_s + \mu_n).$$

This is the same estimator obtained by minimizing the expected loss $\mathbb{E}_{x_s, e_s | P_s}[\|\widehat{y}_s - y_s\|^2]$ of a linear estimator $\widehat{y}_s$ and substituting our estimates for the volume mean and covariance.

6.2. Volume Distance Measures. Given the images $y_1, \ldots, y_n$ together with our mean and covariance estimates μ_n and Σ_n, we can also define distance measures on the underlying volumes. This will allow us to cluster them using methods described in Section 6.3 or to describe their manifold structure using the manifold learning techniques in Section 6.4.

The simplest distance is the Euclidean norm on the volume estimates $\widehat{x}_1, \ldots, \widehat{x}_n$. Specifically,

$$(83) \qquad d_{st}^{(\text{eucl})} = \|\widehat{x}_s - \widehat{x}_t\|,$$

for $s, t = 1, \ldots, n$.

Unfortunately, this distance measure weights all directions equally regardless of their accuracy for a given pair. To see why this is a problem, consider the fact that the columns of $P_s V_{n,r}$ have different norms which depend on s. For example, a volume which is highly oscillatory along one axis will project to almost zero for viewing directions along that axis. Since these vectors are used to estimate $\widehat{x}_s$, this means that the power of the noise is different for each coordinate. A distance measure that takes this into account would therefore be more robust than $d_{st}^{(\text{eucl})}$.

One way to do this is to instead consider distances on the denoised images $\widehat{y}_1, \ldots, \widehat{y}_n$. While we still have the problem of low-energy basis vectors, these do not have a large energy

once reprojected, so the situation is better. We then use the common-lines distance between the images [65]. From the Fourier Slice Theorem (6), we see that the Fourier transforms of two images y_s and y_t occupy the planes orthogonal to $R_s^{(3)}$ and $R_t^{(3)}$, respectively, where $R_s^{(i)}$ denotes the ith row of R_s. As such, they intersect along the line defined by the unit vector

$$(84) \qquad \frac{R_s^{(3)} \times R_t^{(3)}}{\|R_s^{(3)} \times R_t^{(3)}\|}.$$

We therefore define the common-lines vector $\boldsymbol{c}_{st} \in \mathbb{R}^2$ for image s with respect to image t by rotating this vector into the image coordinates

$$(85) \qquad \boldsymbol{c}_{st} = \left[R_s^{(1)}; R_s^{(2)}\right] \left(\frac{R_s^{(3)} \times R_t^{(3)}}{\|R_s^{(3)} \times R_t^{(3)}\|}\right).$$

The common lines of $\widehat{y}_s$ and $\widehat{y}_t$ with respect to one another are then $\mathcal{F}\widehat{y}_s(k\boldsymbol{c}_{st})$ and $\mathcal{F}\widehat{y}_t(k\boldsymbol{c}_{ts})$, respectively, where $k \in M_N$. If y_s and y_t are projections of the same molecular structure, that is $x_s = x_t$, we expect that these common lines should be close since they are restrictions of the same volume Fourier transform along the same line. A useful distance is therefore

$$(86) \qquad d_{st}^{(\mathrm{cl})} = \sum_{k \in M_N} |\mathcal{F}\widehat{y}_s(k\boldsymbol{c}_{st}) - \mathcal{F}\widehat{y}_t(k\boldsymbol{c}_{ts})|^2,$$

for $s, t = 1, \ldots, n$, which we call the common-lines distance.

6.3. Clustering. In the previous sections, we estimated the 3D covariance matrix and used it to calculate estimates $\widehat{x}_1, \ldots, \widehat{x}_n$ of the volumes $x_1, \ldots, x_n$, or more specifically, their low-dimensional coordinate vectors $\widehat{\alpha}_1, \ldots, \widehat{\alpha}_n$. These were then used to define distances $d^{(\mathrm{eucl})}$ and $d^{(\mathrm{cl})}$ between the volumes. Without any additional assumptions, we cannot extract more information. For this, we need prior information on the distribution of the volume x.

For example, if x is a discrete random variable, we can fit a discrete distribution by clustering the volume estimates $\widehat{x}_1, \ldots, \widehat{x}_n$. Such a model is realistic for many molecules, where the majority of their time is spent in a small number of states, briefly transitioning from one to another.

Instead of clustering the volume estimates $\widehat{x}_1, \ldots, \widehat{x}_n$ themselves, we work on the coordinates $\widehat{\alpha}_1, \ldots, \widehat{\alpha}_n$, since these are lower-dimensional but isometrically parametrize the volumes. One clustering approach is the k-means vector quantization algorithm [46]. While widely used for clustering, k-means has several problems, one of which is that it favors partitions that distribute points uniformly between clusters. This can be partly mitigated by modeling x using a Gaussian mixture model (GMM), which can be estimated using the expectation-maximization algorithm [16].

Given the distances $d^{(\mathrm{eucl})}$ or $d^{(\mathrm{cl})}$ instead of coordinates, we can use standard graph clustering algorithms such as normalized cut [66]. These algorithms partition the points into subsets that optimize certain criteria, the goal being to minimize the distances within clusters while maximizing distances between clusters.

Regardless of the clustering mechanism, we obtain a cluster assignment associated with each image. We then average the corresponding volume estimates to obtain an reconstruction

of that class. However, as we shall see, the algorithm will typically be applied to downsampled images, so this reconstruction is by necessity of low resolution. A more accurate reconstruction is obtained by partitioning the dataset according to the cluster assignments and reconstructing each subset separately at full resolution using tools such as RELION [62], FREALIGN [28], or ASPIRE [86, 67, 69].

We note that performing full-resolution reconstruction for each subset provides refined estimates of the viewing directions $R_1, \ldots, R_n$. While these are assumed given for our algorithm, they are not necessarily very accurate, since they must be estimated from the average molecular structure as discussed in Section 2. Since these subsets of particles given by clustering should be more homogeneous, we expect the estimates to be more accurate. Using these, the covariance estimation and clustering steps can be repeated to achieve better results. This approach, known as iterative refinement, has proved useful for other cryo-EM problems [77, 24] and can provide more detailed reconstruction results at the cost of additional computational time.

6.4. Continuous Variability and Diffusion Maps. Certain molecules do not spend most of their time in a small number of states, but have a continuously varying structure. In this case, the clustering approach outlined above fails since the volumes x do not follow a discrete distribution. However, due to physical constraints on the molecular dynamics, this continuum of states can often be described by a small number of dominant flexible motions. In this section, we describe a method for analyzing this low-dimensional manifold using the diffusion maps framework [13].

These tools have previously been applied to study the continuous variability of molecular structure by calculating diffusion maps for images in each viewing direction and "patching" these together to yield a diffusion map for the whole volume [15]. Unfortunately, this is a heuristic method which is not guaranteed to yield an accurate description of the low-dimensional conformation manifold. We propose to use the mean and covariance estimates, together with the derived distance measures $d^{(\mathrm{eucl})}$ or $d^{(\mathrm{cl})}$ to calculate a diffusion map embedding that more closely captures the underlying structural variability.

The first step is to form a similarity matrix W whose entries are

$$(87) \qquad W_{st} := \exp\left(-\frac{d_{st}^2}{\epsilon}\right),$$

for $s, t = 1, \ldots, n$, where d_{st} is a distance between volumes s and t while ϵ is a scale parameter that depends on the smoothness of the manifold. The distance d_{st} can be one of $d^{(\mathrm{eucl})}$ or $d^{(\mathrm{cl})}$. In the following, we shall use $d^{(\mathrm{eucl})}$.

To normalize W, we sum the entries along each row and define the diagonal matrix D with entries

$$(88) \qquad D_{ss} := \sum_{t=1}^{n} W_{st},$$

for $s = 1, \ldots, n$. Renormalizing W by D, we obtain the row-stochastic Markov transition matrix $A := D^{-1}W$. This defines a random walk on the graph of volume estimates, with the

transition probability between two points proportional to their similarity. Let us calculate the eigenvalues and eigenvectors of A. We then have

$$(89) \qquad\qquad A\phi_i = \lambda_i \phi_i,$$

for $i = 1, \ldots, n$. These eigenvalues satisfy $|\lambda_i| \leq 1$ for all i and there is at least one eigenvalue equal to 1 with the corresponding eigenvector parallel to the all-ones vector (this follows from the fact that A is row-stochastic). We thus order the eigenvalues as $1 = \lambda_1 \geq |\lambda_2| \geq \ldots \geq |\lambda_n|$.

The diffusion coordinates at diffusion time τ for the sth volume are now

$$(90) \qquad\qquad \widehat{\beta}_s^{(\tau)} := [\lambda_2^\tau \phi_{2,s}, \ldots, \lambda_n^\tau \phi_{n,s}],$$

where $\phi_{i,s}$ is the sth element of the ith eigenvector ϕ_i [13]. The diffusion time τ specifies the scale of the diffusion embedding. Specifically, the distance $\|\widehat{\beta}_s^{(\tau)} - \widehat{\beta}_t^{(\tau)}\|$ between two diffusion map coordinates $\widehat{\beta}_s^{(\tau)}$ and $\widehat{\beta}_t^{(\tau)}$ approximates the distance between the probability distributions obtained from random walks on the volume graph starting at $\widehat{x}_s$ and $\widehat{x}_t$ after τ steps. As τ increases, these probability distributions overlap more and more for points close together on the manifold. By increasing τ, we obtain a set of coordinates $\widehat{\beta}_1^{(\tau)}, \ldots, \widehat{\beta}_n^{(\tau)}$ that are more robust to noise compared to $\widehat{\alpha}_1, \ldots, \widehat{\alpha}_n$ at the expense of smoothing out the fine-scale manifold structure.

Restricting $\widehat{\beta}_s$ to the first two or three coordinates, we obtain a two- or three-dimensional embedding of the estimated volumes, which provides a helpful visualization to determine the global geometric structure of the continuous manifold of conformations. We shall see some examples of this in Section 7.5.

7. Simulation results. We evaluate the performance of the covariance estimation algorithm using simulated data. These are obtained by applying the forward model (11) to different configurations of volumes, projection mappings, and noise sources. The resulting images are then given as input to the covariance estimation algorithms outlined in Sections 4–6 to study their computational efficiency and accuracy.

7.1. Conditioning & Convergence Results. To reach a given accuracy the conjugate gradient method requires $O(\sqrt{\kappa(Z)})$ iterations to invert the operator Z, where $\kappa(Z)$ is its condition number [61, 5, 73]. As such, we would like to compare this quantity for the operators A_n and $C_n^{-1}A_n$, as well as L_n and $D_n^{-1}L_n$ in order to evaluate the effect of the strategy outlined in Section 5.3. In addition, we examine its effect on the convergence rate of the CG method.

To estimate the condition numbers, we generate a dataset with $n = 16384$ projections of size $N = 16$. We evaluate the effect of preconditioning in three different scenarios. First, we have no CTF (that is $\mathcal{F}h_s = 1$ for all $s = 1, \ldots, n$) and uniform distribution of viewing angles. This gives $\kappa(L_n) \approx 650$ and $\kappa(D_n^{-1}L_n) \approx 170$. Adding three distinct CTFs gives $\kappa(L_n) \approx 1400$ and $\kappa(D_n^{-1}L_n) \approx 230$, resulting in slightly worse conditioning due to the zeros of the CTFs. Finally, we generate the viewing angles using a non-uniform distribution over SO(3), increasing the condition numbers to $\kappa(L_n) \approx 4800$ and $\kappa(D_n^{-1}L_n) \approx 720$. While the preconditioned operator still increases in condition number, it gains a factor of 6 over the non-preconditioned variant. Since the number of iterations required for convergence scales
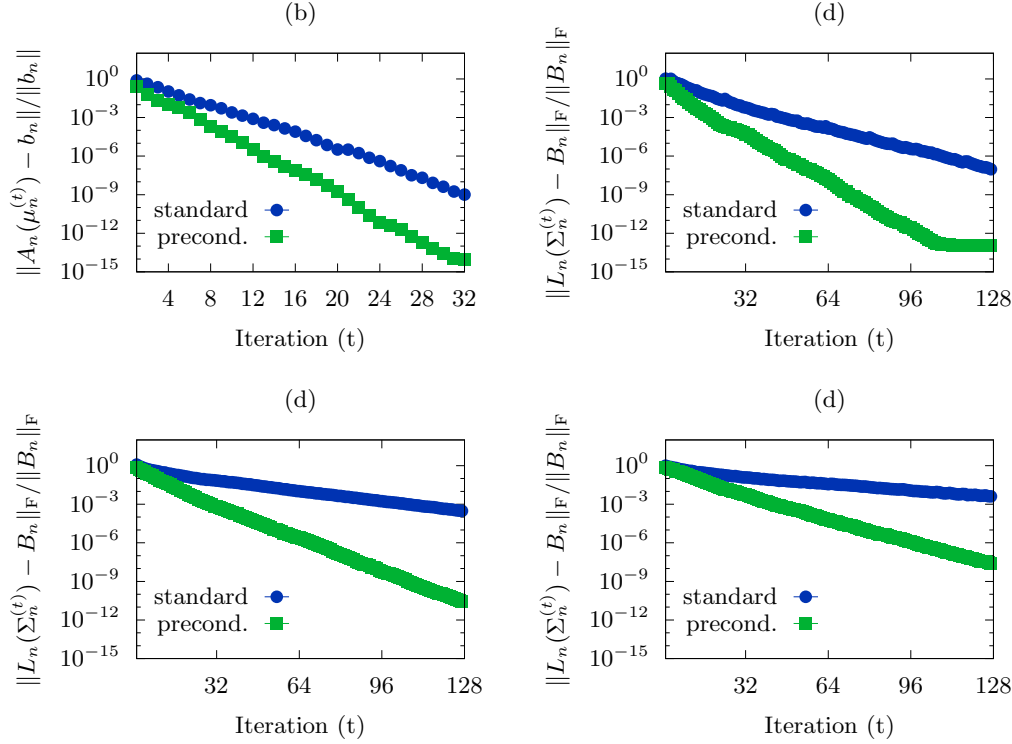
Figure 5. (a) The relative residuals for each iteration of CG applied to $A_n \mu_n = b_n$, denoted by $\mu_n^{(t)}$, with no CTF and uniform distribution of viewing angles. (b-d) The relative residuals for each CG iterate $\Sigma_n^{(t)}$ of $L_n(\Sigma_n) = B_n$ with (b) no CTF and uniform distribution of viewing angles, (c) three distinct CTFs and uniform distribution of viewing angles, and (d) three distinct CTFs and non-uniform distribution of viewing angles. For all plots, the residuals of the standard (unpreconditioned) CG method is compared with using a circulant preconditioner. All methods were applied to $n = 16384$ images of size $N = 16$.

with the square root of the condition number, this should reduce the number of iterations by a factor of 2.5.

We can observe the effects of preconditioning on the convergence rate of the CG method as shown in Figure 5. In Figure 5(a) we have the CG residuals at different iterations for $A_n \mu_n = b_n$, with and without preconditioning, for the baseline case of no CTF and uniform viewing angles. Preconditioning allows CG to converge much faster, reaching a relative residual of 10^{-3} in 8 iterations instead of 13.

Similar plots are obtained for CG applied to $L_n(\Sigma_n) = \Sigma_n$ in Figure 5(b), which similarly has no CTF and uniform distribution of viewing angles. Reaching relative residual 10^{-3} takes 20 iterations compared to 47 for the non-preconditioned algorithm. In Figure 5(c), we see that adding CTFs slows convergence for both methods but with the preconditioned algorithm showing a similar improvement. It reduces the number of iterations required to attain a relative residual of 10^{-3} from 107 to 31, resulting in a speedup of at least 3x. Finally, making the distribution of viewing angles non-uniform gives an even worse convergence rate as shown in Figure 5(d). However, the preconditioned algorithm only requires 44 iterations to converge
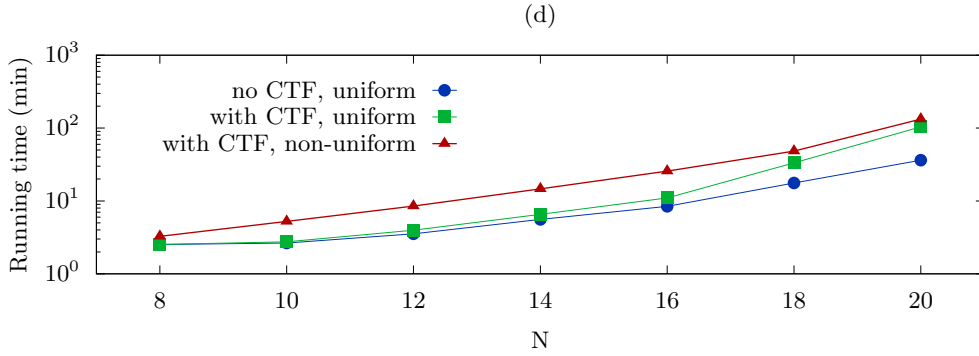
(d)

Figure 6. *Running times for the whole covariance estimation algorithm applied to a dataset of size $n = 16384$ with varying image size N. Three scenarios are considered: no CTF with uniform distribution of viewing angles, three distinct CTFs with uniform distribution of viewing angles, and three CTFs with non-uniform distribution of viewing angles.*

while without preconditioning, more than 128 iterations are required.

We note that applying $C_n^{-1}L_n$ requires more time than L_n. However, C_n^{-1} only involves 6D FFTs of size N, while the 6D FFTs in L_n have size $2N$, that is 64 times larger. The additional time required is therefore expected to be small. Indeed, one preconditioned iteration of CG requires, on average, 15% more time compared to the non-preconditioned iteration.

7.2. Running Time Results. To evaluate the efficiency of our method, we measure its running time on the datasets outlined in the previous section. That is, we have $n = 16384$ images in three different configurations. The first has no CTF and uniform distribution of viewing angles, whereas the second introduces three distinct CTFs into the imaging model. Finally, the third set of experiments were performed with CTFs and non-uniform distribution of viewing angles. The running times are reported in Figure 6 as a function of the image size N. The experiments were run on a dual 14-core Intel Xeon CPU with clock frequency 2.4 GHz.

We see that in the relatively well-conditioned case of uniform distribution of viewing angles, the algorithm terminates quite quickly. Specifically, it takes around 8 minutes when no CTFs are present and 12 minutes with CTFs. However, when we no longer have a uniform distribution of viewing angles, we have a running time of 25 minutes. In all of these scenarios, the dominant step is the conjugate gradient method, since each iteration requires a 6D FFT of size $2N$ and a large number of iterations can be required when the system is ill-conditioned. That being said, the results compare favorably with other methods for CPU-based heterogeneity, such as RELION, which can take up to 800 minutes to run on the same dataset.

7.3. Covariance Estimation Results. The experiments in this section and the following consider the performance of the estimation algorithms with respect to the signal-to-noise ratio (SNR) of the images. Since the task is to extract the heterogeneity structure of the data, a standard SNR measure comparing the average signal power to that of the noise is insufficient.
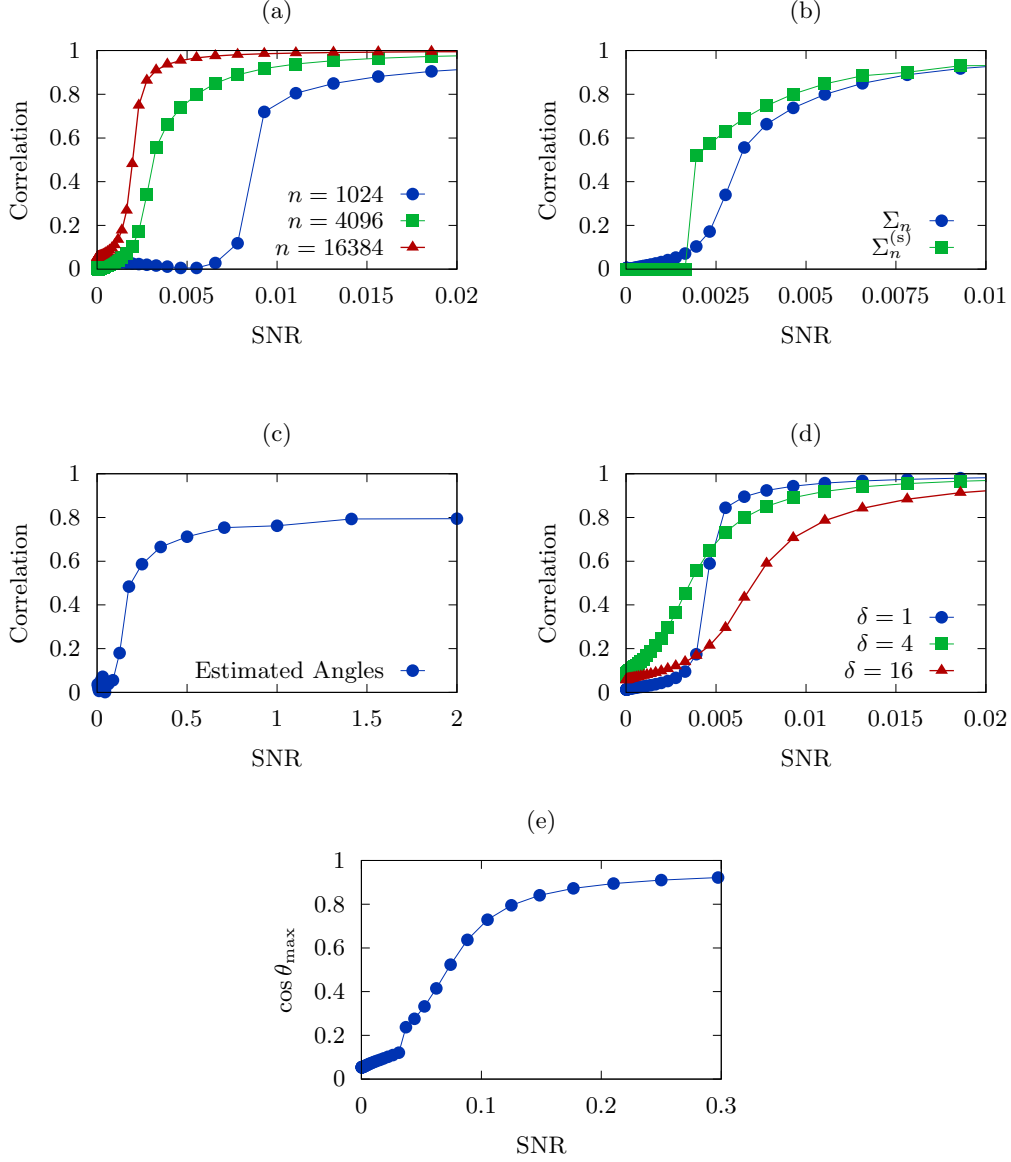
Figure 7. *Covariance estimation results for different simulations with $N = 8$, $C = 2$, three distinct CTFs, and unless otherwise noted, $n = 1024$ and uniform distribution of orientations. (a) The correlation of the top eigenvector of Σ_n with that of $\mathrm{Cov}[x]$ as a function of SNR for $n = 1024$, $n = 4096$, $n = 16384$. (b) Top eigenvector correlations for Σ_n and $\Sigma_n^{(s)}$. (c) Top eigenvector correlations for Σ_n with orientations estimated using the ASPIRE toolbox. (d) Top eigenvector correlations for Σ_n with different orientation distributions over $\mathrm{SO}(3)$ described by (92). (e) The cosine of the maximum principal angle between the top three eigenvectors of Σ_n and those of $\mathrm{Cov}[x]$ for a simulation with $C = 4$ classes, as a function of SNR.*

Instead, we consider the heterogeneous SNR, given by

$$\text{(91)} \qquad \text{SNR} = \frac{\sum_{s=1}^{n} \|P_s(x_s - \mathbb{E}[x])\|^2}{nN^2\sigma^2}.$$

That is, we center the clean images $P_s x_s$ by subtracting the projection of the mean volume $\mathbb{E}[x]$ and compute the square norm of these coefficients before dividing by the noise power σ^2. As a result, for a fixed σ^2, a dataset with low variability will yield a lower SNR compared to a dataset with higher variability. This is the same notion of SNR used by Katsevich et al. [35].

We first test the least-squares estimators μ_n and Σ_n on a simulated dataset consisting of two 70S ribosome configurations at different noise levels with $n = 1024$ images and $N = 8$. Since $C = 2$, the population covariance $\text{Cov}[x]$ is of rank one, so we are only interested in reconstructing its top eigenvector. Figure 7 therefore plots the correlation of the top eigenvector of Σ_n with that of $\text{Cov}[x]$ for different SNRs, where a high correlation implies a good reconstruction. Below a certain SNR, the correlation drops to zero while above a critical threshold the eigenvector correlation approaches one. This behavior is typical of the high-dimensional PCA model discussed in Section 4.3 and the critical SNR threshold can similarly be observed to vary proportionally to the inverse square root of the number of images $n^{-1/2}$. This is illustrated in Figure 7(a), where the correlation as a function of SNR is shown for $n = 1024$, $n = 4096$ and $n = 16384$.

Having established a baseline performance, we study the effect of replacing B_n with its shrinkage variant $B_n^{(\text{s})}$ in the estimator. Figure 7(b) plots the eigenvector correlation as a function of SNR for both Σ_n and $\Sigma_n^{(\text{s})}$, where $\Sigma_n^{(\text{s})} = L_n^{-1}(B_n^{(\text{s})})$. We see that the shrinkage makes a difference for SNRs between 0.001 and 0.01. Above 0.01, while the shrinkage provides a good estimate of the covariance, the top eigenvector is already well-correlated even without shrinkage, so there is little difference in performance. Below 0.001, the signal eigenvalues are absorbed by the noise bulk, so there is no possibility of extracting accurate eigenvectors and both variants perform badly. Between these values, however, shrinkage makes a difference, obtaining an eigenvector correlation of 0.6 for an SNR half that of the standard estimator.

We also study the robustness of the estimation algorithm with respect to errors in viewing angle estimation. Running the standard orientation estimation algorithms in the ASPIRE toolbox, which relies on class averaging [86] followed by a common-lines based synchronization [67, 69], we can apply the covariance estimation method using the estimated viewing angles. The results are shown for a range of SNRs in Figure 7(c). We see that for this particular set of molecular structures, we recover the viewing angles with enough accuracy to allow good covariance estimation.

So far, the distribution of viewing angles has been uniform over SO(3), although it is not required for the covariance estimation algorithms. To demonstrate the robustness to non-uniform distributions, we draw the rotations R_s from a family of distributions on SO(3) indexed by a parameter δ which determines the skew of the distribution towards the identity rotation I_3. Representing the rotation matrices using Euler angles (α, β, γ) in the relative z-y-z convention, we consider the distributions

$$\text{(92)} \qquad \begin{aligned} \alpha &\sim U[0, 2\pi] \\ \beta &\sim \cos^{-1}(2U[0,1]^\delta - 1) \ , \\ \gamma &\sim U[0, 2\pi] \end{aligned}$$

where $\delta = 1$ is a uniform distribution over SO(3) and higher values of δ concentrate the distribution closer to I_3. The resulting eigenvector correlations are shown in Figure 7(d) as a function of SNR for different values of δ. We see that as long as the distribution is not completely skewed in one direction, we are able to recover the covariance structure quite well at low signal-to-noise ratios.

Increasing the number of classes to $C = 4$, we see that the method handles this type of variability just as well as for two classes. Since $C = 4$, the population covariance is of rank 3, so instead of evaluating the top eigenvector of Σ_n, we need to consider the top 3 eigenvectors. A simple correlation will not do, and we instead consider the maximum principal angles between the subspace spanned by the top three eigenvectors of Σ_n and those of $\text{Cov}[x]$. The cosine $\cos\theta$ of the maximum principal angle θ between two subspaces U and V is given by

$$(93) \qquad \cos\theta = \min_{u \in U, v \in V} \frac{|\langle u, v \rangle|}{\|u\|\|v\|}$$

Given two orthogonal matrices Q_U and Q_V whose columns span r-dimensional subspaces U and V, respectively, $\cos\theta$ is obtained as the rth singular value of $Q_U^{\mathrm{T}} Q_V$. This value is plotted as a function of SNR in Figure 7(e).

7.4. Clustering Results. We now consider the clustering and reconstruction steps for the baseline estimator at SNR $= 0.05$, where the top eigenvector correlation equals 0.99. Figure 8(a) shows the spectrum of the covariance estimate Σ_n. Since $C = 2$, we expect there to be one dominant eigenvalue since the population covariance is of rank one. Indeed, there is one eigenvalue that stands out from the bulk noise distribution, so we form $\Sigma_{n,1}$ by extracting the dominant eigenvector and eigenvalue. The Wiener filter described in Section 6.1 gives us a set of scalar coordinate estimates $\{\widehat{\alpha}_1^{(1)}, \ldots, \widehat{\alpha}_n^{(1)}\}$. Their histogram is shown in Figure 8(b). A clear bimodal distribution suggests that we do indeed have two molecular structures present in the data. Clustering the coordinates using k-means as described in Section 6.3 and comparing with the ground truth assignments, we achieve 94.5% accuracy.

To study the behavior of the complete pipeline at different signal-to-noise ratios, we plot the clustering accuracy with respect to SNR in Figure 8(c). Similarly, the reconstruction error with respect to SNR is plotted in Figure 8(d). As expected, we observe a phase transition phenomenon similar to that of the top eigenvector correlations. Once a certain threshold is passed, we classify well and obtain high-quality reconstructions. Below the threshold, however, the estimated eigenvectors correlate badly with the population eigenvectors so we are unable to identify the important directions of variability in the molecules. As a result, the subsequent clustering and reconstruction steps fail.

7.5. Manifold Learning Results. To simulate continuous variability, we deform a ribosome molecule in two regions by rotating two parts of it independently of one another. This yields a two-parameter family of molecular structures with the two parameters being the two angles of rotations. Specifically, the manifold described by these molecular structures is the two-dimensional torus, which can be embedded in three dimensions. As a result, we need at least three eigenvectors to properly capture the manifold. The population covariance of the volumes is not strictly low-rank, but 83% of the variance is concentrated in the leading 4 eigenvectors. If we can recover this eigenspace using our covariance estimation method, we
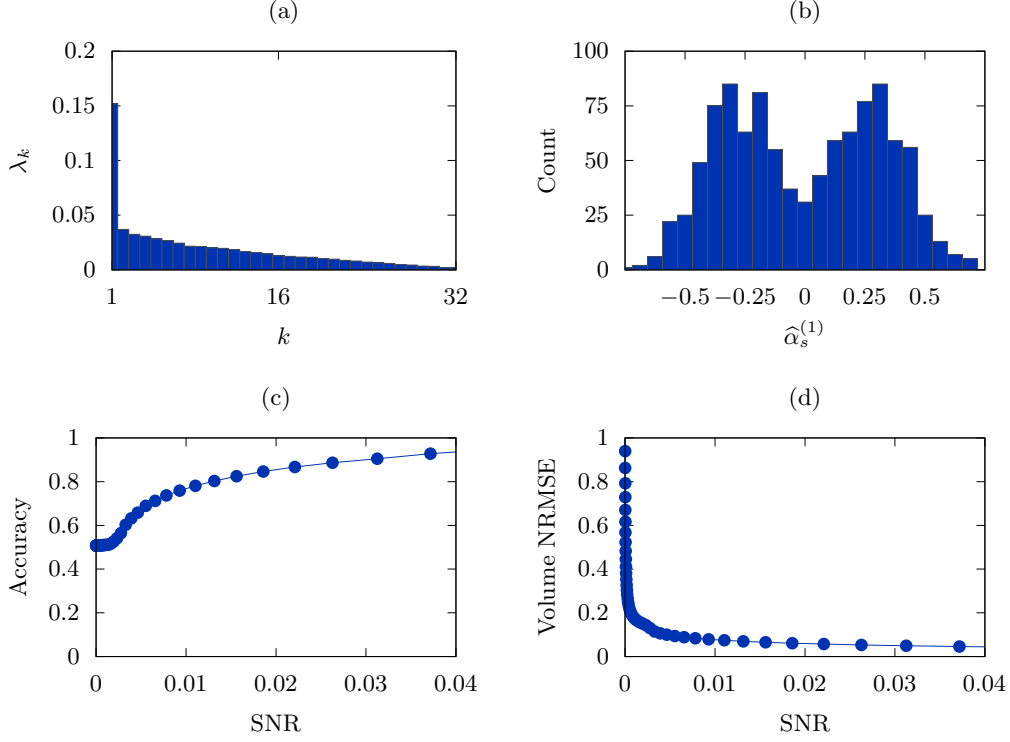
Figure 8. *Clustering results for discrete variability with $C = 2$ classes imaged using $n = 1024$ images with resolution $N = 8$ for uniform distribution of viewing angles and three distinct CTFs. (a) The top 32 eigenvalues of Σ_n obtained at $\mathrm{SNR} = 0.05$. (b) A histogram of the coordinates $\widehat{\alpha}_s^{(1)}$ corresponding to the images y_s for $s = 1, \ldots, n$ subject to the same SNR. (c) The fraction of images classified correctly as a function of SNR. (d) The normalized root mean square error (NRMSE) of the reconstructed volumes as a function of SNR.*

should be able to estimate the manifold structure of the continuous variability. The cosine of the maximum principal angle between the top four population eigenvectors and those of the estimated covariance Σ_n is plotted in Figure 9(a) as a function of SNR. For an SNR above 0.05 these top eigenvectors are well-estimated, with the cosine of the maximum principal angle in excess of 0.90.

Fixing the SNR at 0.125, we calculate the coordinates $\{\widehat{\alpha}_1, \ldots, \widehat{\alpha}_n\}$ of the images using the mean estimate μ_n and the top four eigenvectors of Σ_n. From these we compute a diffusion map embedding. Figure 9(b) plots first three embedding coordinates: first colored according to one rotation angle, second colored according to the other. The embedding successfully reproduces these angles, indicating that the procedure captures the two-parameter structure quite well. Comparing the estimated volumes $\widehat{x}_s$ with the ground truth volumes x_s for $s = 1, \ldots, 4096$, we obtain an NRMSE of 0.31. From this, the recovered range of molecular structure appears to be quite accurate. Plotting the NRMSE as a function of the diffusion map coordinates in Figure 9(c), we see that in areas with high sampling density, the reconstruction is more accurate compared to areas with more sparse sampling. Since the neighborhoods in the sparsely sampled regions have fewer images, this loss of accuracy is expected.
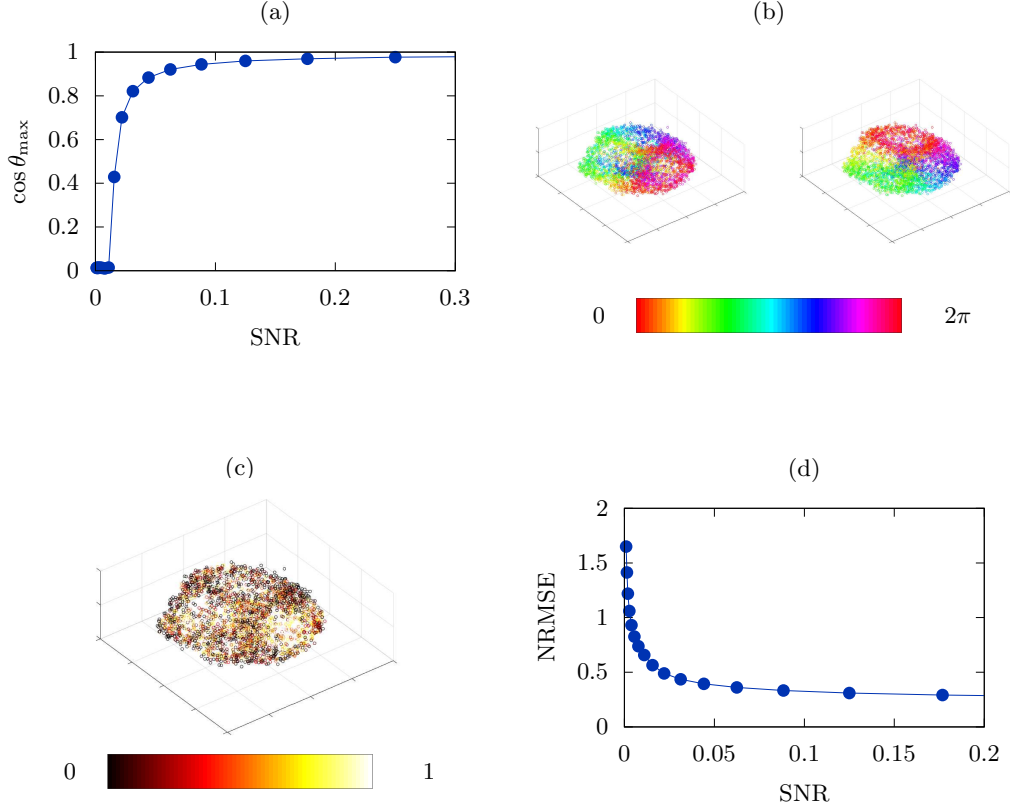
Figure 9. *Manifold learning results for continuous variability. (a) The cosine of the maximum principal angle between the top four population eigenvectors and those of Σ_n. (b) Three-dimensional diffusion map embedding coordinates of the volume coordinates $\{\widehat{\alpha}_1, \ldots, \widehat{\alpha}_n\}$, colored according to the first and second rotation angles. (c) The NRMSE of each volume estimate as a function of its diffusion map coordinate. (d) The NRMSE of the reconstructed volumes as a function of SNR.*

We also study the performance of the manifold learning reconstruction as a function of SNR in Figure 9(d). As in the discrete case, we see that for fixed n, higher SNR results in more accurate reconstructions.

8. Experimental results. Although simulations are useful to understand the workings of an algorithm, they do not suffice to demonstrate the practical usefulness of a tool. We therefore investigate our covariance estimation approach on an experimental dataset obtained from real cryo-EM samples. This is a standard dataset published by Joachim Frank's lab [43] consisting of projections of a 70S ribosome in two distinct conformations. The dataset comes with a labeling of the images as coming from one of the two states. This labeling was obtained using supervised classification.

For the 70S ribosome dataset, we have 10000 images of size 130-by-130. We first run the RELION software [62] with the number of classes set to one. This provides an estimate of the
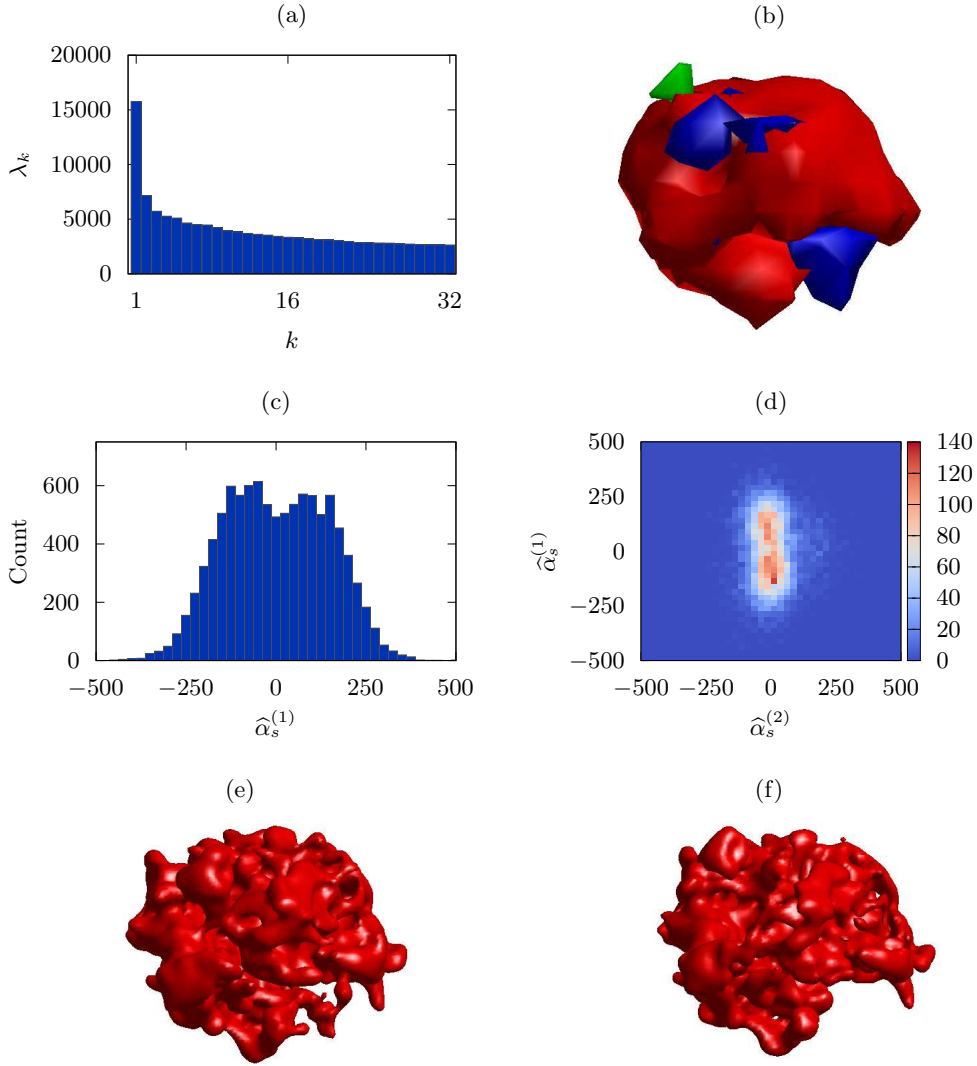
Figure 10. *Covariance estimation on the 70S ribosome dataset. (a) Largest eigenvalues of the estimated covariance matrix Σ_n. (b) The estimated mean volume (red), together with the positive (blue) and negative (green) components of the top eigenvector. (c) Histogram of coordinates $\{\widehat{\alpha}_1, \ldots, \widehat{\alpha}_n\}$ from the Wiener filter estimator. (d) Two-dimensional histogram of coordinates $\{(\widehat{\alpha}_1^{(1)}, \widehat{\alpha}_1^{(2)}), \ldots, (\widehat{\alpha}_n^{(1)}, \widehat{\alpha}_n^{(2)})\}$ from the Wiener filter estimator. (e,f) Full-resolution reconstructions obtained using RELION applied to the clusters identified in (c).*

viewing angles and translations associated with each projection image which are then given as input to our algorithms. To speed up computations, we downsample the images to 16-by-16. Since our goal is to cluster the images, it is not necessary to do this at full resolution if the discriminant features are already present at low resolution. The images are then whitened so that the noise is approximately white with variance $\sigma^2 = 1$. Given the images and the estimates of viewing angles and translations, we apply our mean estimation algorithm (see Algorithm 1). Using the mean estimate μ_n, we then apply our covariance estimation method

(see Algorithm 2).

The top eigenvalues of the covariance matrix estimate Σ_n is shown in Figure 10(a). There is a large eigenvalue well-separated from the rest, suggesting that the variability in molecular structure is at least one-dimensional, which is consistent with a two-class configuration. However, there is also a second eigenvalue of significant amplitude, possibly indicating the presence of a small third class. Figure 10(b) shows the estimated mean volume (red) with the leading eigenvector superimposed (positive part in blue, negative part in green). In the top part of the molecule, we observe the rotation of a small subunit indicated by the negative values on one side and positive values on the other side of that subunit. In the bottom, there is a subunit that attaches and detaches depending on the class.

To investigate this further, we plot the one-dimensional histogram obtained from the first coordinates $\widehat{\alpha}_s^{(1)}$ for $s = 1, \ldots, n$, and a two-dimensional heatmap obtained from $(\widehat{\alpha}_s^{(1)}, \widehat{\alpha}_s^{(2)})$ for $s = 1, \ldots, n$ in Figures 10(c) and 10(d), respectively. The heatmap is obtained by dividing the plane into square boxes and counting the number of points in each box. There does seem to be at least two structures while a third structure is hard to discern. It is therefore likely that this second dimension is due to some continuous variability between the two states.

Clustering the coordinates $\widehat{\alpha}_1^{(1)}, \ldots, \widehat{\alpha}_n^{(1)}$ into two classes and reconstructing from these subsets at full resolution, we obtain the two molecular structures shown in Figures 10(e,f). The two are very similar, with two differences consisting of the rotating subunit at the top and the appearance of the bottom subunit, which agrees the changes observed in the leading eigenvector (see Figure 10(b)). This is in agreement with the presumed structures of the dataset, which consists of a ribosome with and without EF-G. If we compare our clustering with the given annotation of the dataset, we obtain 88.7% accuracy. To compare, the accuracy achieved by RELION on the same dataset with number of classes set to two is 84.6%.

The total running time of Algorithms 1 and 2 was 9 minutes on a quad-core 3.4 GHz Intel Core i7 CPU. The initial estimation of viewing angles and translations using RELION took 356 minutes. This compares with running RELION configured with two classes, which had a running time of 520 minutes. Enabling support for GPU with a GeForce GTX 980 Ti, these running times dropped to 22 and 28 minutes for one and two classes, respectively.

9. Future Work. The proposed algorithm scales as $N^6 \log N$ in the image size N. As a result, doubling the size of the image results in more than a 64-fold increase in running time, not to mention memory usage. This makes estimation for N larger than 20 prohibitive in most cases. Since the estimated covariance matrix contains N^6 entries, this is close to optimal for that task. However, we often expect the population covariance to be approximately low rank, so it should be possible to reduce the computational complexity by incorporating a low-rank constraint during estimation. In addition, we will explore related approaches for low-rank matrix recovery such as low-rank matrix sensing via alternating minimization [33] and direct shrinkage of singular values [17].

Another drawback of the above approach is that it requires the viewing directions and translations to be known in advance. While this may be possible for small, localized heterogeneity, where homogeneous reconstruction methods can provide reasonable estimates, this is not always feasible. In that case, methods which combine heterogeneous reconstruction and parameter estimation enjoy a significant advantage [42, 63]. Extending the proposed method

for this more general setting provides another avenue for future research.

Finally, more work is needed to process the coordinate estimates $\widehat{\alpha}_1, \ldots, \widehat{\alpha}_n$. While standard clustering and manifold learning approaches outlined in this paper provide reasonable results, they do not perform well at high noise levels. A subject of further investigation is therefore how to incorporate informative priors on the space of volumes, such as variability caused by deformation, which should prove useful in this regime.

10. Conclusion. In this work, we have introduced a computationally efficient method for least-squares estimation of the 3D covariance matrix of the molecular structure from noisy 2D projection images. Given n images of size N-by-N, it has computational complexity $\mathcal{O}(\sqrt{\kappa} N^6 \log N + nN^4)$, where κ is in the range 10–200 for typical problems. This is achieved by reformulating the normal equations as a deconvolution problem, that is then appropriately preconditioned and solved using the conjugate gradient method. We also introduce a shrinkage variant which improves accuracy at low signal-to-noise ratios, decreasing by a factor of two the necessary signal power for accurate estimation. The estimated covariance matrices are then used to reconstruct the three-dimensional structures using Wiener filter techniques, which are then clustered into a number of discrete states or fitted to a continuous manifold structure. The accurate performance of both methods is confirmed through experiments on simulated and experimental datasets.

REFERENCES

[1] G. S. AMMAR AND W. B. GRAGG, *The generalized Schur algorithm for the superfast solution of Toeplitz systems*, in Rational approximation and its applications in mathematics and physics, Springer, 1987, pp. 315–330.

[2] A. AMUNTS, A. BROWN, X.-C. BAI, J. L. LLÁCER, T. HUSSAIN, P. EMSLEY, F. LONG, G. MURSHUDOV, S. H. SCHERES, AND V. RAMAKRISHNAN, *Structure of the yeast mitochondrial large ribosomal subunit*, Science, 343 (2014), pp. 1485–1489.

[3] J. ANDÉN, E. KATSEVICH, AND A. SINGER, *Covariance estimation using conjugate gradient for 3D classification in cryo-EM*, in 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI), IEEE, 2015, pp. 200–204.

[4] J. ANDÉN AND A. SINGER, *Factor analysis for spectral estimation*, in 2017 International Conference on Sampling Theory and Applications (SampTA), July 2017, pp. 169–173, https://doi.org/10.1109/SAMPTA.2017.8024447.

[5] O. AXELSSON, *Iterative solution methods*, Cambridge university press, 1996.

[6] A. BARNETT, L. GREENGARD, A. PATAKI, AND M. SPIVAK, *Rapid solution of the cryo-EM reconstruction problem by frequency marching*, SIAM Journal on Imaging Sciences, 10 (2017), pp. 1170–1195, https://doi.org/10.1137/16M1097171.

[7] W. T. BAXTER, R. A. GRASSUCCI, H. GAO, AND J. FRANK, *Determination of signal-to-noise ratios and spectral SNRs in cryo-EM low-dose imaging of molecules*, J. Struct. Biol., 166 (2009), pp. 126–132.

[8]   T. Bhamre, T. Zhang, and A. Singer, *Anisotropic twicing for single particle reconstruction using autocorrelation analysis*, (2017), https://arxiv.org/abs/1704.07969.

[9]   R. H. Chan and M. K. Ng, *Conjugate gradient methods for Toeplitz systems*, SIAM review, 38 (1996), pp. 427–482.

[10] T. F. Chan, *An optimal circulant preconditioner for Toeplitz systems*, SIAM Journal on Scientific and Statistical Computing, 9 (1988), pp. 766–771, https://doi.org/10.1137/0909051.

[11] X. Cheng, *Random Matrices in High-dimensional Data Analysis*, PhD thesis, Princeton University, 2013.

[12] Y. Cheng, N. Grigorieff, P. A. Penczek, and T. Walz, *A primer to single-particle cryo-electron microscopy*, Cell, 161 (2015), pp. 438–449.

[13] R. R. Coifman and S. Lafon, *Diffusion maps*, Applied and computational harmonic analysis, 21 (2006), pp. 5–30.

[14] J. W. Cooley and J. W. Tukey, *An algorithm for the machine calculation of complex Fourier series*, Mathematics of computation, 19 (1965), pp. 297–301.

[15] A. Dashti, P. Schwander, R. Langlois, R. Fung, W. Li, A. Hosseinizadeh, H. Y. Liao, J. Pallesen, G. Sharma, V. A. Stupina, et al., *Trajectories of the ribosome as a brownian nanomachine*, Proceedings of the National Academy of Sciences, 111 (2014), pp. 17492–17497.

[16] A. Dempster, N. Laird, and D. Rubin, *Maximum likelihood from incomplete data via the EM algorithm*, Journal of the Royal Statistical Society. Series B (Methodological), (1977), pp. 1–38.

[17] E. Dobriban, W. Leeb, and A. Singer, *Optimal prediction in the linearly transformed spiked model*, (2017), https://arxiv.org/abs/1709.03393.

[18] D. L. Donoho, M. Gavish, and I. M. Johnstone, *Optimal shrinkage of eigenvalues in the spiked covariance model*, (2013), https://arxiv.org/abs/1311.0851.

[19] V. L. Druskin and L. A. Knizhnerman, *Two polynomial methods of calculating functions of symmetric matrices*, USSR Computational Mathematics and Mathematical Physics, 29 (1989), pp. 112–121.

[20] A. Dutt and V. Rokhlin, *Fast Fourier transforms for nonequispaced data*, SIAM Journal on Scientific computing, 14 (1993), pp. 1368–1393.

[21] Editorial Board, *Method of the year 2015*, Nature Methods, 13 (2016), p. 1.

[22] H. Erickson and A. Klug, *Measurement and compensation of defocusing and aberrations by Fourier processing of electron micrographs*, Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences, (1971), pp. 105–118.

[23] J. A. Fessler, S. Lee, V. T. Olafsson, H. R. Shi, and D. C. Noll, *Toeplitz-based iterative image reconstruction for MRI with correction for magnetic field inhomogeneity*, Transactions on Signal Processing, 53 (2005), pp. 3393–3402.

[24] J. Frank, *Three-dimensional electron microscopy of macromolecular assemblies*, Academic Press, 2006.

[25] M. Gavish and D. L. Donoho, *Optimal shrinkage of singular values*, IEEE Transactions on Information Theory, 63 (2017), pp. 2137–2152.

[26] G. H. Golub and C. F. Van Loan, *Matrix computations*, vol. 3, JHU Press, 2012.

[27] L. Greengard and J.-Y. Lee, *Accelerating the nonuniform fast Fourier transform*, SIAM review, 46 (2004), pp. 443–454.

[28] N. Grigorieff, *FREALIGN: High-resolution refinement of single particle structures*, Journal of structural biology, 157 (2007), pp. 117–125.

[29] M. Guerquin-Kern, D. V. D. Ville, C. Vonesch, J.-C. Baritaux, K. P. Pruessmann, and M. Unser, *Wavelet-regularized reconstruction for rapid MRI*, in Proceedings of the International Symposium on Biomedical Imaging (ISBI), IEEE, 2009, pp. 193–196.

[30] G. T. Herman, *Fundamentals of computerized tomography: Image reconstruction from projections*, Springer Science & Business Media, 2009.

[31] M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, Journal of Research of the National Bureau of Standards, 49 (1952), http://dx.doi.org/10.6028/jres.049.044.

[32] N. J. Higham, *Functions of matrices: theory and computation*, SIAM, 2008.

[33] P. Jain, P. Netrapalli, and S. Sanghavi, *Low-rank matrix completion using alternating minimization*, in Proceedings of the forty-fifth annual ACM symposium on Theory of computing, ACM, 2013, pp. 665–674.

[34] I. M. Johnstone, *On the distribution of the largest eigenvalue in principal components analysis*, Annals of statistics, (2001), pp. 295–327.

[35] E. KATSEVICH, A. KATSEVICH, AND A. SINGER, *Covariance matrix estimation for the cryo-EM heterogeneity problem*, SIAM Journal on Imaging Sciences, 8 (2015), pp. 126–185.

[36] S. M. KAY, *Fundamentals of statistical signal processing, volume I: Estimation theory*, Prentice Hall, 1993.

[37] M. KHOSHOUEI, M. RADJAINIA, W. BAUMEISTER, AND R. DANEV, *Cryo-EM structure of haemoglobin at 3.2 Å determined with the volta phase plate*, Nature communications, 8 (2017).

[38] D. KIMANIUS, B. O. FORSBERG, S. H. SCHERES, AND E. LINDAHL, *Accelerated cryo-EM structure determination with parallelisation using GPUs in RELION-2*, eLife, 5 (2016), p. e18722.

[39] A. KLUG AND R. A. CROWTHER, *Three-dimensional image reconstruction from the viewpoint of information theory*, Nature, 238 (1972), pp. 435–440.

[40] W. KÜHLBRANDT, *The resolution revolution*, Science, 343 (2014), pp. 1443–1444.

[41] S. KUNIS AND D. POTTS, *Fast spherical Fourier algorithms*, Journal of Computational and Applied Mathematics, 161 (2003), pp. 75–98.

[42] R. R. LEDERMAN AND A. SINGER, *A representation theory perspective on simultaneous alignment and classification*, (2016), https://arxiv.org/abs/1607.03464.

[43] H. LIAO AND J. FRANK, *Classification by bootstrapping in single particle methods*, in 2010 IEEE International Symposium on Biomedical Imaging (ISBI), IEEE, 2010, pp. 169–172.

[44] H. Y. LIAO, Y. HASHEM, AND J. FRANK, *Efficient estimation of three-dimensional covariance and its application in the analysis of heterogeneous samples in cryo-electron microscopy*, Structure, 23 (2015), pp. 1129–1137.

[45] M. LIAO, E. CAO, D. JULIUS, AND Y. CHENG, *Structure of the TRPV1 ion channel determined by electron cryo-microscopy*, Nature, 504 (2013), p. 107.

[46] S. LLOYD, *Least squares quantization in PCM*, IEEE Transactions on Information Theory, 28 (1982), pp. 129–137.

[47] S. MALLAT, *A wavelet tour of signal processing*, Academic Press, 1999.

[48] V. A. MARČENKO AND L. A. PASTUR, *Distribution of eigenvalues for some sets of random matrices*, Mathematics of the USSR-Sbornik, 1 (1967), p. 457.

[49] E. MICHIELSSEN AND A. BOAG, *A multilevel matrix decomposition algorithm for analyzing scattering from large structures*, IEEE Transactions on Antennas and Propagation, 44 (1996), pp. 1086–1093.

[50] J. L. MILNE, M. J. BORGNIA, A. BARTESAGHI, E. E. TRAN, L. A. EARL, D. M. SCHAUDER, J. LENGYEL, J. PIERSON, A. PATWARDHAN, AND S. SUBRAMANIAM, *Cryo-electron microscopy–a primer for the non-microscopist*, FEBS Journal, 280 (2013), pp. 28–45.

[51] J. A. MINDELL AND N. GRIGORIEFF, *Accurate determination of local defocus and specimen tilt in electron microscopy*, Journal of structural biology, 142 (2003), pp. 334–347.

[52] B. MUSICUS, *Levinson and fast Cholesky algorithms for Toeplitz and almost Toeplitz matrices*, 1984.

[53] F. NATTERER, *The mathematics of computerized tomography*, Springer, 1986.

[54] D. PAUL, *Asymptotics of sample eigenstructure for a large dimensional spiked covariance model*, Statistica Sinica, (2007), pp. 1617–1642.

[55] P. PENCZEK, M. KIMMEL, AND C. SPAHN, *Identifying conformational states of macromolecules by eigen-analysis of resampled cryo-EM images*, Structure, 19 (2011), pp. 1582–1590.

[56] P. A. PENCZEK, *Chapter two: Image restoration in cryo-electron microscopy*, Methods Enzymol., 482 (2010), pp. 35–72.

[57] A. PUNJANI, J. L. RUBINSTEIN, D. J. FLEET, AND M. A. BRUBAKER, *cryoSPARC: algorithms for rapid unsupervised cryo-EM structure determination*, Nature Methods, 14 (2017), pp. 290–296.

[58] J. RADON, *Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten*, Berichte Sächsishen Akad. Wissenschaft., Math. Phys. Klass, 69 (1917), pp. 262–277.

[59] V. ROKHLIN AND M. TYGERT, *Fast algorithms for spherical harmonic expansions*, SIAM Journal on Scientific Computing, 27 (2006), pp. 1903–1928.

[60] Y. SAAD, *Analysis of some Krylov subspace approximations to the matrix exponential operator*, SIAM Journal on Numerical Analysis, 29 (1992), pp. 209–228.

[61] Y. SAAD, *Iterative methods for sparse linear systems*, SIAM, 2003.

[62] S. SCHERES, *RELION: implementation of a Bayesian approach to cryo-EM structure determination*, Journal of structural biology, 180 (2012), pp. 519–530.

[63] S. H. SCHERES, *A Bayesian view on cryo-EM structure determination*, Journal of Molecular Biology, 415

(2012), pp. 406–418.

[64] J. SCHUR, *Über potenzreihen, die im innern des einheitskreises beschränkt sind.*, Journal für die reine und angewandte Mathematik, 147 (1917), pp. 205–232.

[65] M. SHATSKY, R. HALL, E. NOGALES, J. MALIK, AND S. BRENNER, *Automated multi-model reconstruction from single-particle electron microscopy data*, Journal of Structural Biology, 170 (2010), pp. 98–108.

[66] J. SHI AND J. MALIK, *Normalized cuts and image segmentation*, IEEE Transactions on pattern analysis and machine intelligence, 22 (2000), pp. 888–905.

[67] Y. SHKOLNISKY AND A. SINGER, *Viewing direction estimation in cryo-EM using synchronization*, SIAM journal on imaging sciences, 5 (2012), pp. 1088–1110.

[68] F. J. SIGWORTH, *A maximum-likelihood approach to single-particle image refinement*, Journal of Structural Biology, 122 (1998), pp. 328–339.

[69] A. SINGER, R. R. COIFMAN, F. J. SIGWORTH, D. W. CHESTER, AND Y. SHKOLNISKY, *Detecting consistent common lines in cryo-EM by voting*, Journal of structural biology, 169 (2010), pp. 312–322.

[70] D. SLEPIAN, *Prolate spheroidal wave functions, Fourier analysis and uncertainty–iv: extensions to many dimensions; generalized prolate spheroidal functions*, Bell System Technical Journal, 43 (1964), pp. 3009–3057.

[71] G. STRANG, *A proposal for Toeplitz matrix calculations*, Studies in Applied Mathematics, 74 (1986), pp. 171–176.

[72] H. D. TAGARE, A. KUCUKELBIR, F. J. SIGWORTH, H. WANG, AND M. RAO, *Directly reconstructing principal components of heterogeneous particles from cryo-EM images*, Journal of structural biology, 191 (2015), pp. 245–262.

[73] L. N. TREFETHEN AND D. BAU, *Numerical linear algebra*, vol. 50, SIAM, 1997.

[74] M. TYGERT, *Fast algorithms for spherical harmonic expansions, II*, Journal of Computational Physics, 227 (2008), pp. 4260–4279.

[75] M. TYGERT, *Fast algorithms for spherical harmonic expansions, III*, Journal of Computational Physics, 229 (2010), pp. 6181–6192.

[76] E. E. TYRTYSHNIKOV, *Optimal and superoptimal circulant preconditioners*, SIAM Journal on Matrix Analysis and Applications, 13 (1992), pp. 459–473.

[77] M. VAN HEEL, B. GOWEN, R. MATADEEN, E. V. ORLOVA, R. FINN, T. PAPE, D. COHEN, H. STARK, R. SCHMIDT, M. SCHATZ, ET AL., *Single-particle electron cryo-microscopy: towards atomic resolution*, Quarterly reviews of biophysics, 33 (2000), pp. 307–369.

[78] K. R. VINOTHKUMAR AND R. HENDERSON, *Single particle electron cryomicroscopy: trends, issues and future perspective*, Quarterly reviews of biophysics, 49 (2016).

[79] C. VONESCH, L. WANG, Y. SHKOLNISKY, AND A. SINGER, *Fast wavelet-based single-particle reconstruction in cryo-EM*, in Proceedings of the International Symposium on Biomedical Imaging (ISBI), IEEE, 2011, pp. 1950–1953.

[80] M. VULOVIĆ, R. B. RAVELLI, ET AL., *Image formation modeling in cryo-electron microscopy*, J. Struct. Biol., (183(1): 19–32 (2013)).

[81] R. WADE, *A brief look at imaging and contrast transfer*, Ultramicroscopy, 46 (1992), pp. 145–156.

[82] F. T. A. W. WAJER AND K. P. PRUESSMANN, *Major speedup of reconstruction for sensitivity encoding with arbitrary trajectories*, in Proceedings of the 8th Annual Meeting of the ISMRM, 2001, p. 767.

[83] L. WANG, Y. SHKOLNISKY, AND A. SINGER, *A Fourier-based approach for iterative 3D reconstruction from cryo-EM images*, (2013), https://arxiv.org/abs/1307.5824.

[84] L. WANG, A. SINGER, AND Z. WEN, *Orientation determination of cryo-EM images using least unsquared deviations*, SIAM journal on imaging sciences, 6 (2013), pp. 2450–2483.

[85] Z. ZHAO AND A. SINGER, *Fourier–Bessel rotational invariant eigenimages*, J. Opt. Soc. Am. A, 30 (2013), pp. 871–877.

[86] Z. ZHAO AND A. SINGER, *Rotationally invariant image representation for viewing direction classification in cryo-EM*, Journal of structural biology, 186 (2014), pp. 153–166.