2014 IEEE 28-th Convention of Electrical and Electronics Engineers in Israel

Collaborative Detection Of Common Lines In Cryo EM Images Using Maximum Likelihood

Mor Cohen
School of Electrical Engineering
Tel-Aviv University

Yoel Shkolnisky
Department of Applied Mathematics
Tel-Aviv University

Arie Yeredor
School of Electrical Engineering
Tel-Aviv University

***Abstract*—This paper presents a maximum likelihood (ML) algorithm for detecting (shared) common lines between pairs of cryo-EM projection images. The algorithm is based on a global iterative detector in which we jointly estimate and classify common lines using the data from all projection images. We demonstrate by simulations that the algorithm improves the detection rate of common lines compared to state of the art methods, and operates well even with non white imaging noise.**

I. Introduction

Cryo-electron microscopy (cryo-EM) is a methodology for determining the three dimensional structure of a molecule from its two dimensional projection images. The imaging directions of these projection images are unknown, and they are extremely noisy. Algorithms for estimating the imaging directions are often based on the "Angular Reconstitution" method of Van Heel [1], in which a coordinate system is established from three projections, and the orientation of the molecule corresponding to each image is deduced from radial common lines among the images. Van Heel showed that the common lines between three projections uniquely determine their relative orientations up to handedness (chirality). This observation is the basis of the angular reconstitution method, which was also developed independently by Vainshtein and Goncharov [2]. The main problem with this method is its sensitivity to false detection of common lines, which leads to errors in the reconstruction of the 3D structure. To mitigate this problem, common lines are usually detected by first denoising the images followed by computing correlations between radial lines in the (Fourier transformed) images. The denoising process takes advantage of the large number of available images, as this implies that there must be images that are identical up to in-plane rotation. Such identical images, which were imaged from the same direction, can be averaged to improve their SNR, assuming they can be correctly detected and aligned. In this work we present a maximum likelihood (ML) algorithm for detecting common lines that takes advantage of the highly redundant data in a different way. Our algorithm classifies the noisy polar Fourier lines from all the images into K clusters, and then uses all lines in a given cluster to estimate the underlying common line. This process of cluster assignment and estimation iterates until convergence.

A. *Fourier slice theorem and common lines*

In cryo-EM, we are given a large set of N noisy two-dimensional projection images of a given molecule, denoted $\{P_n\}_{n=1}^{N}$ which were imaged from unknown random directions (for example see Fig 3). The intensity of pixels in each projection corresponds to line integrals of the electric potential $\phi(x, y, z)$ induced by the molecule along the path of the imaging electrons. Excluding the contribution of noise, the pixel intensities in image P_n correspond to the Radon transform of the underlying molecule, and are given explicitly by

$$P_n\left(x, y\right) = \int \phi_n(x, y, z)dz, \tag{1}$$

where $\phi_n(\boldsymbol{r}) = \phi\left(\boldsymbol{R}_n^{-1}\boldsymbol{r}\right)$, $\boldsymbol{r} = (x, y, z)$, $\boldsymbol{R}_n$ is the unknown rotation matrix describing the orientation of the molecule in the n-th image, and ϕ is the electric potential of the molecule in a fixed "laboratory" coordinate system. The cryo-EM reconstruction problem is thus stated as follows: find $\phi(x, y, z)$ given a collection of N projections $\{P_n\}_{n=1}^{N}$. The imaging process is illustrated in Fig 1. If the rotations $\{\boldsymbol{R}_n\}_{n=1}^{N}$ were known, then the reconstruction could have been performed by classical tomography methods. Unfortunately, in cryo-EM the highly intense electron beam destroys the molecule, and it is therefore impractical to take several projection images of the same molecule as in the case of classical computerized tomography. In other words, a single molecule can be imaged only once, and by using many copies of the same molecule, we obtain many projection images of the same underlying structure. Since each molecule is free to move in the liquid medium until its orientation is fixed by freezing, every image is a projection of the same molecule but at an unknown viewing angle. In this formulation, all molecules are assumed to have the exact same structure; they differ only by their spatial orientation. The Fourier slice theorem states that the 2D Fourier transform of a projection image P_n, denoted $\hat{P}_n$, is the restriction of the 3D Fourier transform of the projected object $\hat{\phi}$ to the central plane (i.e., going through the origin) perpendicular to the imaging direction (see Fig 2). Thus, the Fourier transforms of any two projections must share a common line. In the derivation of our algorithm we assume that the underlying molecule has no symmetry, and thus each pair of projections shares a single common line. To find these common lines in practice, one first samples L radial lines $\{\boldsymbol{s}_{n,\ell}\}_{\ell=1}^{L}$ from each Fourier transform $\hat{P}_n$ of P_n. Explicitly, $\boldsymbol{s}_{n\ell} \in \mathbb{C}^p$ is given by

$$\begin{aligned} \boldsymbol{s}_{n\ell} &= \hat{P}_n\left(\boldsymbol{\rho}, \alpha_\ell\right) \\ &= \frac{1}{(2\pi)^2} \iint P_n\left(x, y\right) e^{-\imath(x\boldsymbol{\rho} cos\alpha_\ell + y\boldsymbol{\rho} sin\alpha_\ell)} dxdy, \end{aligned} \tag{2}$$

where $\alpha_\ell = \frac{2\pi}{L}\ell$ is the angular direction and $\boldsymbol{\rho}$ is a vector of p equispaced points along the radial line (the integral of a vector is interpreted as applying the integral to each coordinate of the vector). Then for a pair of images, say

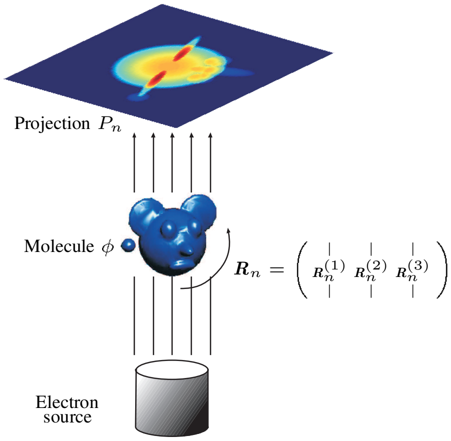

Projection P_n
Molecule ϕ
$R_n = \begin{pmatrix} | & | & | \\ R_n^{(1)} & R_n^{(2)} & R_n^{(3)} \\ | & | & | \end{pmatrix}$
Electron source

Fig. 1: Illustration of the cryo-EM imaging process: each projection image corresponds to the line integrals of an unknown molecule rotated by an unknown three-dimensional rotation.

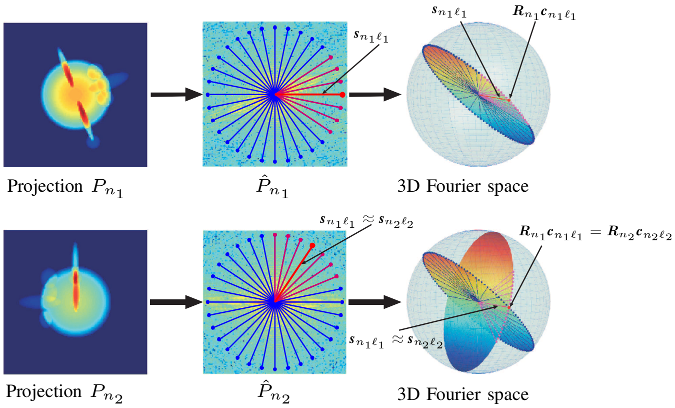


Fig. 2: Fourier slice theorem and its induced geometry.

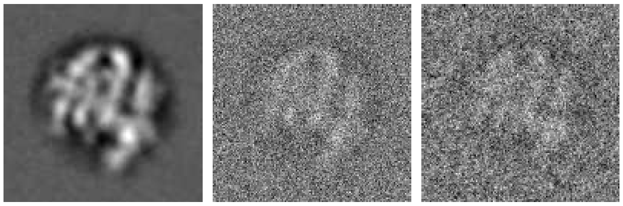
P_{n_1}, P_{n_2}, there exists a line ℓ_1 in image n_1 and a line ℓ_2 in image n_2, such that

$$\|s_{n_1\ell_1} - s_{n_2\ell_2}\|^2 \leq \epsilon, \tag{3}$$

where ϵ depends on L. Equivalently, if $\hat{P}_{n_1}$ and $\hat{P}_{n_2}$ are the two-dimensional Fourier transforms of projections P_{n_1} and P_{n_2} , then in the noiseless case, there must be a central line in $\hat{P}_{n_1}$ and a central line in $\hat{P}_{n_2}$ on which the two transforms agree, up to a discretization error which depends on L (see Fig 2). This pair of vectors is known as the common line. The geometry induced by the Fourier slice theorem is illustrated in Fig 2. Any two slices share a common line, i.e. the intersection line of the two planes. Every radial line in the two-dimensional Fourier transform of a projection image is also a radial line in the three-dimensional Fourier transform of the molecule. Moreover, it can be shown [4] that the common line for each pair of images satisfies

$$\boldsymbol{R}_{n_1}\boldsymbol{c}_{n_1\ell_1} = \boldsymbol{R}_{n_2}\boldsymbol{c}_{n_2\ell_2}, \tag{4}$$

where $\boldsymbol{c}_{n_1\ell_1}$ and $\boldsymbol{c}_{n_2\ell_2}$ are the direction unit vectors of the common line in the images n_1 and n_2, respectively. Once enough common lines are correctly detected, there are several algorithms for estimating $\{\boldsymbol{R}_n\}_{n=1}^N$ without any additional geometric information (see e.g. [6] and references therein).



(a) clean (b) SNR=$\frac{1}{10}$ (AWGN) (c) SNR=$\frac{1}{10}$ (AWCN)

Fig. 3: One of the simulated projections of the density map of the E. coli 50S ribosomal subunit with additive Gaussian noise.

B. Problem formulation

Following (3), let P_{n_1}, P_{n_2} be two 2D tomographic projection images of the same molecule, taken from different orientations, and let $\{\boldsymbol{s}_{n_1,\ell}\}_{\ell=1}^L$ and $\{\boldsymbol{s}_{n_2,\ell}\}_{\ell=1}^L$ be the collections of their polar Fourier lines given by equation (2). By the Fourier slice theorem, there is a common line shared between these two images, namely indices ℓ_1 and ℓ_2 such that

$$\boldsymbol{s}_{n_1\ell_1} = \boldsymbol{s}_{n_2\ell_2} = \boldsymbol{s}_k \tag{5}$$

where $\boldsymbol{s}_k \in \mathbb{C}^p$ is the clean (unknown) vector shared by $\hat{P}_{n_1}$ and $\hat{P}_{n_2}$. Our goal is to find these indices for each pair of images, when the signal to noise ratio is extremely low, that is when $\frac{\sigma^2_{projection}}{\sigma^2_{noise}} < 1$.

II. Maximum Likelihood Detection

A major component of the noise in the images is "shot noise", associated with the process of electron counting in the detector, and originates from the discrete nature of the electric charge [3]. For a large number of electrons, the shot noise approaches a normal distribution. Therefore, in this work we assume a fairly standard model in which each polar Fourier line $\boldsymbol{s}_{n\ell}$ is contaminated by an additive zero mean complex-valued, circular Gaussian noise (for example see Fig 3). Namely,

$$\boldsymbol{x}_{n\ell} = h_n \boldsymbol{s}_{n\ell} + \boldsymbol{\varepsilon}_{n\ell}, \tag{6}$$

where $n = 1, ..., N$ is the image index, $\ell = 1, ..., L$ is the radial line index, $\boldsymbol{s}_{n\ell} \in \mathbb{C}^p$ are unknown vectors of length p modeling the clean underlying signal, $\boldsymbol{h} = \{h_n\}_{n=1}^N$ are unknown real numbers modeling the variation in brightness of each image, and $\boldsymbol{\varepsilon}_{n\ell}$ are correlated complex-valued, circular Gaussian noise with a known covariance matrix $\boldsymbol{C}_{\varepsilon\varepsilon}$

$$\boldsymbol{\varepsilon}_{n\ell} \sim \mathcal{CN}(0, \boldsymbol{C}_{\varepsilon\varepsilon}). \tag{7}$$

As will be explained below, the main assumption behind our algorithm is that $\boldsymbol{s}_{n\ell}$ are samples from a finite set of K signals, denoted $\boldsymbol{S} = \{\boldsymbol{s}_k\}_{k=1}^K$. In other words, no matter how many images we collect, their polar Fourier lines will be noisy snapshots of these K signals. Therefore, we propose to detect common lines between images by clustering the noisy polar Fourier lines into K clusters. Specifically, our algorithm first finds ML estimate of $\boldsymbol{S}$ by classifying the noisy Fourier lines into K clusters, such that $\boldsymbol{S}$ are the clusters' centers. Then, it finds the common line index between each pair of images by searching for the two lines that share the same cluster. To derive the ML, we start by determining the number K (as explained below). Then, for each noisy observation $\boldsymbol{x}_{n\ell}$ and its clean underlying signal $\boldsymbol{s}_{n\ell}$ (see equation (6)) we introduce a corresponding set of “indicators”, denoted $\boldsymbol{z}_{n\ell} = \{z_{n\ell,k}\}_{k=1}^{K}$, describing the label of the cluster to which $\boldsymbol{s}_{n\ell}$ belongs. More precisely, $\boldsymbol{z}_{n\ell}$ is a vector of length K, where its k-th element, denoted by $z_{n\ell,k}$ equals 1 if $\boldsymbol{s}_{n\ell} = \boldsymbol{s}_k$. The elements of the indicator are mutually exclusive and exhaustive (i.e., only one of the $z_{n\ell,k}$ is equal to 1, and all others are 0). In addition, we assume that the elements are (a-priori) uniformly distributed over the labels, that is,

$$Pr(z_{n\ell,k} = 1) = \frac{1}{K} \quad \forall n, \ell, k. \tag{8}$$

Next, a common line match is defined when there are two lines $\boldsymbol{x}_{n_1\ell_1}$,$\boldsymbol{x}_{n_2\ell_2}$ that share the same cluster $\boldsymbol{s}_k$. Namely, that there exists a label k such that

$$z_{n_1\ell_1,k} = z_{n_2\ell_2,k} = 1. \tag{9}$$

Thus, our goal is to compute the ML estimate of the unknown parameters of the mixture, i.e. the signals $\boldsymbol{S}$ and the indicator vectors $\boldsymbol{z}_{n\ell}$. A classical well-known solution for estimating such parameters is to use the Expectation Maximization (EM) algorithm [5]. For applying the EM algorithm we consider $\boldsymbol{X} = [\boldsymbol{x}_{11}, ..., \boldsymbol{x}_{NL}]$ as the observed data, and add $\boldsymbol{Z} = [\boldsymbol{z}_{11}, ..., \boldsymbol{z}_{NL}]$ so that $(\boldsymbol{X}, \boldsymbol{Z})$ together form the complete data. The unknown parameters are $\boldsymbol{\Theta} = (\boldsymbol{S}, \boldsymbol{h})$. The probabilistic model is $Pr(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{\Theta})$, given by

$$Pr\left(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{\Theta}\right) = \prod_{n=1}^{N}\prod_{\ell=1}^{L} Pr\left(\boldsymbol{x}_{n\ell}, \boldsymbol{z}_{n\ell}; \boldsymbol{\Theta}\right), \tag{10}$$

where the probability of a single element is given by

$$Pr\left(\boldsymbol{x}_{n\ell}, \boldsymbol{z}_{n\ell}; \boldsymbol{\Theta}\right) = \prod_{k=1}^{K}\left[\left(\frac{1}{K}\right) Pr(\boldsymbol{x}_{n\ell}|\boldsymbol{s}_{n\ell} = \boldsymbol{s}_k)\right]^{z_{n\ell,k}}. \tag{11}$$

Given an estimate $\boldsymbol{\Theta}^{(t)}$ of the parameters from a previous iteration, the EM algorithm proceeds iteratively by alternating between the E-Step and the M-Step as follows:

- In the E-Step we calculate $R(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)}) = \mathbf{E}_{Z|X;\Theta^{(t)}}\left[\log Pr(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{\Theta})\right]$.
- In the M-Step we maximize $R(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)})$ with respect to $\boldsymbol{\Theta}$, thereby obtaining $\boldsymbol{\Theta}^{(t+1)}$, the new estimate of $\boldsymbol{\Theta}$.

We denote all the quantities estimated by our algorithm with superscript $(\cdot)^{(t)}$, where t is the iteration index.

A. E-Step

From (11) we get

$$\log Pr(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{\Theta}) = \sum_{n,\ell,k} z_{n\ell,k}\big(-\log(K) + \log Pr(\boldsymbol{x}_{n\ell}|\boldsymbol{s}_{n\ell} = \boldsymbol{s}_k; \boldsymbol{S}, \boldsymbol{h})\big). \tag{12}$$

From the complex-valued, circular Gaussian measurement model we have

$$Pr(\boldsymbol{x}_{n\ell}|\boldsymbol{s}_{n\ell} = \boldsymbol{s}_k; \boldsymbol{S}, \boldsymbol{h}) = \frac{1}{\pi|\boldsymbol{C}_{\varepsilon\varepsilon}|}\exp\left\{-(\boldsymbol{x}_{n\ell} - h_n\boldsymbol{s}_k)^H \boldsymbol{C}_{\varepsilon\varepsilon}^{-1}(\boldsymbol{x}_{n\ell} - h_n\boldsymbol{s}_k)\right\},$$

obtaining

$$\log Pr(\boldsymbol{X}, \boldsymbol{Z}; \boldsymbol{\Theta}) = \sum_{n,\ell,k} z_{n\ell,k}(\beta_k - (\boldsymbol{x}_{n\ell} - h_n\boldsymbol{s}_k)^H \boldsymbol{C}_{\varepsilon\varepsilon}^{-1}(\boldsymbol{x}_{n\ell} - h_n\boldsymbol{s}_k)), \tag{13}$$

where $\beta_k = -\log(K) - \log(\pi|\boldsymbol{C}_{\varepsilon\varepsilon}|)$. For the E-step we need to take the mean of this expression with respect to the missing data $\boldsymbol{Z}$, given $\boldsymbol{X}$ and assuming that the parameters are $\boldsymbol{\Theta}^{(t)}$. It is evident that given $\boldsymbol{X}$, the only random elements in (13) are the elements $z_{n\ell,k}$ which multiply the other (deterministic) terms, and therefore we only need to calculate

$$z_{n\ell,k}^{(t)} = \mathbf{E}\left(z_{n\ell,k}|\boldsymbol{x}_{n\ell}; \boldsymbol{S}^{(t)}, h_n^{(t)}\right). \tag{14}$$

Now since $z_{n\ell,k}$ equals either 1 or 0, the expectation step (E-step) is given by

$$\begin{aligned} z_{n\ell,k}^{(t)} &= Pr\left(z_{n\ell,k} = 1|\boldsymbol{x}_{n\ell}; \boldsymbol{S}^{(t)}, h_n^{(t)}\right) \qquad (15)\\ &= \frac{Pr\left(z_{n\ell,k} = 1\right) Pr\left(\boldsymbol{x}_{n\ell}|z_{n\ell,k} = 1; \boldsymbol{s}_k^{(t)}, h_n^{(t)}\right)}{Pr\left(\boldsymbol{x}_{n\ell}; \boldsymbol{S}^{(t)}, h_n^{(t)}\right)}\\ &= \frac{\frac{1}{K}\exp\left\{-(\boldsymbol{x}_{n\ell} - h_n^{(t)}\boldsymbol{s}_k^{(t)})^H \boldsymbol{C}_{\varepsilon\varepsilon}^{-1}(\boldsymbol{x}_{n\ell} - h_n^{(t)}\boldsymbol{s}_k^{(t)})\right\}}{\frac{1}{K}\sum_{k=1}^{K}\exp\left\{-(\boldsymbol{x}_{n\ell} - h_n^{(t)}\boldsymbol{s}_k^{(t)})^H \boldsymbol{C}_{\varepsilon\varepsilon}^{-1}(\boldsymbol{x}_{n\ell} - h_n^{(t)}\boldsymbol{s}_k^{(t)})\right\}} \end{aligned}$$

where we used Bayes theorem, and equation (8). Next, we assume for simplicity that the denominator is equal for all indices i, ℓ and k. Thus, we denote it with a constant α_1 and plug it in the equation above to get

$$\begin{aligned} z_{n\ell,k}^{(t)} &= \alpha_1 Pr\left(\boldsymbol{x}_{n\ell}|z_{n\ell,k} = 1; \boldsymbol{s}_k^{(t)}, h_n^{(t)}\right) \qquad (16)\\ &= \alpha_2 exp\left(-(\boldsymbol{x}_{n\ell} - h_n^{(t)}\boldsymbol{s}_k^{(t)})^H \boldsymbol{C}_{\varepsilon\varepsilon}^{-1}(\boldsymbol{x}_{n\ell} - h_n^{(t)}\boldsymbol{s}_k^{(t)})\right), \end{aligned}$$

where α_1, α_2 are variables independent of i, ℓ, k which ensure that $\sum_k Pr(z_{n\ell,k} = 1|x_{n\ell}; \boldsymbol{S}^{(t)}, h_n^{(t)}) = 1$. In words, $z_{n\ell,k}^{(t)}$ is a soft estimate of the probability that $\boldsymbol{x}_{n\ell}$ belongs to the k-th cluster (i.e. $\boldsymbol{s}_{n\ell} = \boldsymbol{s}_k^{(t)}$). We therefore have

$$R(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)}) = \sum_{n,\ell,k} z_{n\ell,k}^{(t)}(\beta_k - (\boldsymbol{x}_{n\ell} - h_n\boldsymbol{s}_k)^H \boldsymbol{C}_{\varepsilon\varepsilon}^{-1}(\boldsymbol{x}_{n\ell} - h_n\boldsymbol{s}_k)), \tag{17}$$

where the expression for $z_{n\ell,k}^{(t)}$ is given in (16) above.

B. M-Step

We now need to maximize $R(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)})$ with respect to $\boldsymbol{\Theta}$. To this end, since this is a non-linear, not-convex optimization problem, we take a two-step approach, first maximizing with respect to the elements $\{s_k\}_{k=1}^{K}$ of $\boldsymbol{S}$ while substituting each h_n with $h_n^{(t)}$ and regarding these variables as fixed, and then maximizing with respect to the elements of $\boldsymbol{h}$, regarding the elements of $\boldsymbol{S}$ as fixed. We note that this does not guarantee maximization of $R(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)})$ with respect to $\boldsymbol{\Theta}$, but it does guarantee an increase of $R(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)})$. Namely, if the resulting parameters are denoted $\boldsymbol{\Theta}^{(t+1)}$, the relation $R(\boldsymbol{\Theta}^{(t+1)}, \boldsymbol{\Theta}^{(t)}) \geq R(\boldsymbol{\Theta}^{(t)}, \boldsymbol{\Theta}^{(t)})$ is guaranteed, which in turn still guarantees convergence of the EM algorithm to a (possibly local) maximum.

1) ***Maximizing with respect to S:*** Obviously, when maximizing $R(\boldsymbol{\Theta}, \boldsymbol{\Theta}^{(t)})$ with respect to $\boldsymbol{\Theta}$, we only need to minimize $\sum_{n,\ell,k} z^{(t)}_{n\ell,k}(\boldsymbol{x}_{n\ell} - h_n\boldsymbol{s}_k)^H \boldsymbol{C}^{-1}_{\varepsilon\varepsilon}(\boldsymbol{x}_{n\ell} - h_n\boldsymbol{s}_k))$. Substituting h_n with $h_n^{(t)}$, differentiating with respect to $\boldsymbol{s}_k$ and equating to zero yields

$$\boldsymbol{s}_k^{(t+1)} = \frac{\sum_{n\ell} z^{(t)}_{n\ell,k} h_n^{(t)} \boldsymbol{x}_{n\ell}}{\sum_{n\ell} z^{(t)}_{n\ell,k} (h_n^{(t)})^2}. \tag{18}$$

*2) **Maximizing with respect to h**:* Differentiating (17) with respect to h_n and equating to zero yields

$$\sum_{\ell,k} z^{(t)}_{n\ell,k} (\boldsymbol{s}_k^{(t+1)})^H \boldsymbol{C}^{-1}_{\varepsilon\varepsilon} \left(\boldsymbol{x}_{\ell,k} - h_n \boldsymbol{s}_k^{(t+1)}\right) = 0 \tag{19}$$

$$h_n \sum_{\ell,k} z^{(t)}_{n\ell,k} (\boldsymbol{s}_k^{(t+1)})^H \boldsymbol{C}^{-1}_{\varepsilon\varepsilon} \boldsymbol{s}_k^{(t+1)} = \sum_{\ell,k} z^{(t)}_{n\ell,k} (\boldsymbol{s}_k^{(t+1)})^H \boldsymbol{C}^{-1}_{\varepsilon\varepsilon} \boldsymbol{x}_{n\ell}$$

$$h_n^{(t+1)} = \frac{\sum_{\ell,k} z^{(t)}_{n\ell,k} (\boldsymbol{s}_k^{(t+1)})^H \boldsymbol{C}^{-1}_{\varepsilon\varepsilon} \boldsymbol{x}_{n\ell}}{\sum_{\ell,k} z^{(t)}_{n\ell,k} (\boldsymbol{s}_k^{(t+1)})^H \boldsymbol{C}^{-1}_{\varepsilon\varepsilon} \boldsymbol{s}_k^{(t+1)}}.$$

*3) **Maximizing with respect to common line indices**:* Motivated by equation (5) and the definition in (9), the potential common lines are pairs of Fourier lines, from different images, that belong to the same cluster k. The likelihood of a pair, say $\boldsymbol{x}_{n_1\ell_1}, \boldsymbol{x}_{n_2\ell_2}$ is given by

$$\begin{aligned} &Pr(\boldsymbol{x}_{n_1,\ell_1}, \boldsymbol{x}_{n_2,\ell_2} | z_{n_1\ell_1,k} = z_{n_2\ell_2,k} = 1) \\ &\quad = Pr\left(\boldsymbol{x}_{n_1,\ell_1} | z_{n_1\ell_1,k} = 1; \boldsymbol{S}^{(t)}, h_{n_1}^{(t)}\right) \\ &\quad Pr\left(\boldsymbol{x}_{n_2,\ell_2} | z_{n_2\ell_2,k} = 1; \boldsymbol{S}^{(t)}, h_{n_2}^{(t)}\right) = \alpha_1^2 z^{(t)}_{n_1\ell_1,k} z^{(t)}_{n_2\ell_2,k}. \end{aligned} \tag{20}$$

Thus, in each iteration t, we may collect all potential common lines and then for each pair of images (n_1, n_2) search for the triplet ℓ_1, ℓ_2 and k that attains the maximum likelihood, namely

$$(n_1\ell_1, n_2\ell_2) = \underset{\substack{1 \le k \le K \\ 1 \le \ell_1, \ell_2 \le L}}{\arg\max}\; z^{(t)}_{n_1\ell_1,k} z^{(t)}_{n_2\ell_2,k}. \tag{21}$$

In summary, the algorithm for detecting common lines is given in Algorithm 1 below.

III. Practical Considerations

There are several practical considerations that we must address. We will detail two of them in this paper. First, we will elaborate on how we measure the performance of our algorithm. Second, we will explain in detail how we choose the number of clusters K. Due to space restrictions, we do not elaborate on the initialization steps in lines 4-5.

A. Performance metric

Although the Gaussian mixture model is used in a wide range of problems in image processing, not many results are presented on the performance of the classification vs. SNR. Most researches that use the model are more interested in stability of the algorithms (i.e. that they converge for a large variety of images), and in computational efficiency. We, on the other hand, are more interested in performance in the sense of the detection rate (probability of correct common line) vs. the image SNR, given (in dB) by

$$SNR(dB) = 10 \log \left(\frac{\sigma^2_{projection}}{\sigma^2_{noise}} \right). \tag{22}$$

Algorithm 1

```
 1: procedure Cryo-EM Common lines Detection
 2: Initialization:
 3:     $t \leftarrow 0$
 4:     $\left\{\boldsymbol{s}_k^{(t)}\right\}_{k=1}^{K} \leftarrow$ estimate $\{\boldsymbol{s}_k\}_{k=1}^{K}$
 5:     $\left\{h_n^{(t)}\right\}_{n=1}^{N} \leftarrow$ estimate $\{h_n\}_{n=1}^{N}$
 6: E-Step:
 7:     for $1 \le n \le N$ do
 8:         for $1 \le \ell \le L$ do
 9:             Estimate the qualifiers $z^{(t)}_{n\ell} = \left\{z^{(t)}_{n\ell,k}\right\}_{k=1}^{K}$
    using equation (16).
10: M-Step:
11:     Compute $\left\{\boldsymbol{s}_k^{(t+1)}\right\}_{k=1}^{K}$ using equation (18).
12:     Compute $\left\{h_n^{(t+1)}\right\}_{n=1}^{N}$ using equation (19).
13: $t \leftarrow t + 1$.
14: if $\sum_{k=1}^{K} \|\boldsymbol{s}_k^{(t+1)} - \boldsymbol{s}_k^{(t)}\|^2 < \epsilon$ goto Detection.
15: else goto E-Step
16: Detection:
17:     for $n_1 \neq n_2$ do
18:         Detect the common line indices using (21).
```

A common line $(n_1\ell_1), (n_2\ell_2)$ between images n_1 and n_2 is considered correctly detected if the estimated indices ℓ_1 and ℓ_2 deviate from their ground truth (that can be computed using $\boldsymbol{R}_{n_1}$ and $\boldsymbol{R}_{n_2}$ as in [6]) by up to a given tolerance (we have used ±5° in our simulation).

B. Choosing K

Our algorithm depends strongly on the number of signals K that is used for classifying the noisy Fourier lines. Choosing a wrong value for K results in poor detection rates. We infer K from the geometry induced by the Fourier slice theorem, which implies that each clean signal $\boldsymbol{s}_{n\ell}$ from Fourier image $\hat{P}_n$ corresponds to a direction vector $\boldsymbol{\beta}_{n,\ell} = \boldsymbol{R}_n \boldsymbol{c}_{n,\ell}$ on the unit sphere (i.e. a unit vector). Under appropriate smoothness assumptions on ϕ, we have that signals with close direction vectors will be close in the sense of equation (3). Therefore, we define a small spherical cap of angle α on the unit sphere (the set of direction vectors in $\mathbb{R}^3$) and consider all direction vectors falling in the same cap to be the same. This implies that their corresponding clean signals are also the same. Thus, the number of different clean signals K is the number of spherical caps of angle α required to cover the unit sphere in $\mathbb{R}^3$. This idea is illustrated in Fig 4. If we further assume that all spherical caps cluster the same number of vectors, then the number of vectors in each spherical cap is exactly $(N \times L)\sin^2\left(\frac{\alpha}{4}\right)$, where $N \times L$ is the total number of vectors. Now since antipodal caps correspond to the same clean (complex conjugate) signal, K can be estimated by

$$K = \frac{1}{2} \frac{N \times L}{(N \times L)\sin^2\left(\frac{\alpha}{4}\right)} = \frac{1}{2\sin^2\left(\frac{\alpha}{4}\right)}, \tag{23}$$

where N is the number of images, L is the number of radial lines in each image, and α is angle of the cap. Finally,

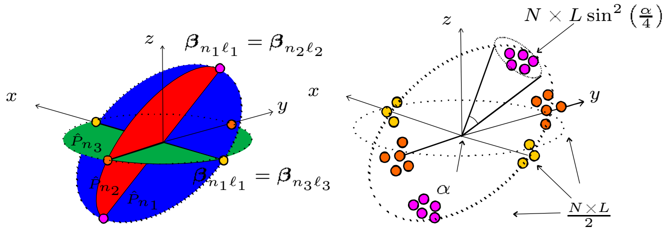

z
$\boldsymbol{\beta}_{n_1\ell_1} = \boldsymbol{\beta}_{n_2\ell_2}$
x
y
P_{n_3}
P_{n_2}
P_{n_1}
$\boldsymbol{\beta}_{n_1\ell_1} = \boldsymbol{\beta}_{n_3\ell_3}$
z
$N \times L \sin^2\left(\frac{\alpha}{4}\right)$
x
y
α
$\frac{N \times L}{2}$

Fig. 4: On the left: three images intersect at three common lines. On the right: we approximte the number of possible clean underlying Fourier lines by using the area of a spherical with an angle of α. All lines that appear on same spherical cap define the same signal.

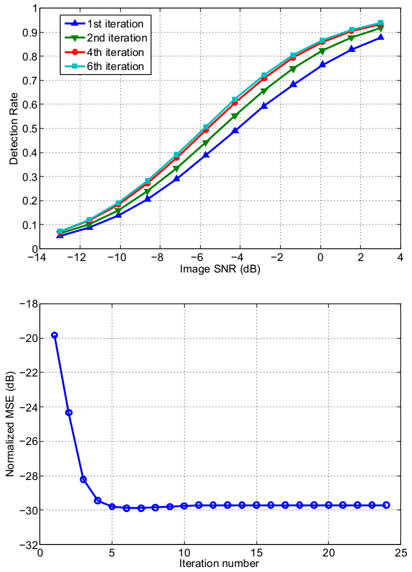


Fig. 5: Convergence: the number of correctly detected common lines increases with each iteration, and the estimated mean square error (MSE) of $\boldsymbol{S}^{(t)}$ converges pretty quickly (six iterations are nearly enough to reach best performance.)

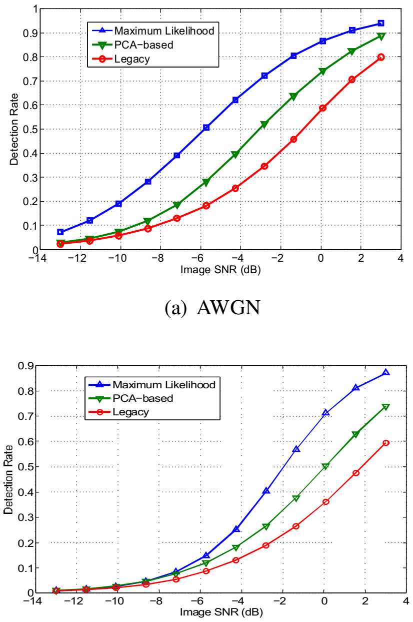
plugging $\alpha = \frac{2\pi}{L}$ yields

$$K = \left\lceil \frac{1}{2\sin^2\left(\frac{2\pi}{4L}\right)} \right\rceil . \qquad (24)$$

IV. Results

In the simulation we generated $N = 1000$ projection images and fixed the angular resolution to be $L = 360$ ($K = 26000$ according to equation (24)). We contaminated each image with additive white Gaussian noise (AWGN) according to a predefined SNR, ran six iterations of the detection algorithm and measured the percentage of correctly detected common lines (detection-rate). The convergence of the algorithm is illustrated in Fig 5. The results of our algorithm are compared to the legacy algorithm (maximum correlations) and to a PCA-based algorithm. We applied our algorithm also to colored Gaussian noise (ACGN), with results in Fig 6.

V. Conclusion



(a) AWGN

(b) ACGN

Fig. 6: Detection-rate vs. SNR with 1000 images.

In this work, we developed a ML framework for detecting common lines between pairs of cryo-EM images. The algorithm we suggest is composed of two steps: in the first step, we iteratively classify the polar Fourier lines into K signals (clusters) via an EM algorithm. That is, for each noisy line we compute soft weights indicating the likelihood of each cluster. In the second step, we detect a "hard" common line index between each pair of images, by searching for two lines, one from each of the two images, that attain maximum likelihood. The number of correctly identified common lines increases with each iteration of our algorithm, which converges within a small number of iterations.

Acknowledgment

This research was supported by THE ISRAEL SCIENCE FOUNDATION grant No. 485/10.

References

[1] Van Heel, M., *Angular reconstitution: a posteriori assignment of projection directions for 3D reconstruction*, Elsevier, Ultramicroscopy 21(2), pp. 111-123, 1987

[2] Vainshtein, B. and Goncharov, A., *Determination of the spatial orientation of arbitrarily arranged identical particles of an unknown structure from their projections*, Proc. llth Intern. Congr. on Elec. Mirco. pp. 459-460, 1986

[3] Frank, J., *Three-Dimensional Electron Microscopy of Macromolecular Assemblies: Visualization of Biological Molecules in Their Native State*, Oxford, 1996.

[4] Penczek, P. A., Zhu, J. and Frank, J., *A common-lines based method for determining orientations for $N > 3$ particle projections simultaneously*, Elsevier, Ultramicroscopy 63, pp. 205-218, 1996.

[5] Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer Science+Business Media, 2006.

[6] Y. Shkolnisky and A. Singer, *Viewing Directions Estimation in Cryo-EM using Synchronization*, SIAM Journal on Imaging Sciences, 5(3), pp. 1088-1110, 2012.