Ab-initio Contrast Estimation and Denoising of Cryo-EM Images

Yunpeng Shi[*1] and Amit Singer[1,2]

[1]Program in Applied and Computational Mathematics, Princeton University
[2]Department of Mathematics, Princeton University

Abstract

Background and Objective:
The contrast of cryo-EM images varies from one to another, primarily due to the uneven thickness of the ice layer. This contrast variation can affect the quality of 2-D class averaging, 3-D ab-initio modeling, and 3-D heterogeneity analysis. Contrast estimation is currently performed during 3-D iterative refinement. As a result, the estimates are not available at the earlier computational stages of class averaging and ab-initio modeling. This paper aims to solve the contrast estimation problem directly from the picked particle images in the ab-initio stage, without estimating the 3-D volume, image rotations, or class averages.

Methods:
The key observation underlying our analysis is that the 2-D covariance matrix of the raw images is related to the covariance of the underlying clean images, the noise variance, and the contrast variability between images. We show that the contrast variability can be derived from the 2-D covariance matrix and we apply the existing Covariance Wiener Filtering (CWF) framework to estimate it. We also demonstrate a modification of CWF to estimate the contrast of individual images.

Results:
Our method improves the contrast estimation by a large margin, compared to the previous CWF method. Its estimation accuracy is often comparable to that of an oracle that knows the ground truth covariance of the clean images. The more accurate contrast estimation also improves the quality of image restoration as demonstrated in both synthetic and experimental datasets.

Conclusions:
This paper proposes an effective method for contrast estimation directly from noisy images without using any 3-D volume information. It enables contrast correction in the earlier stage of single particle analysis, and may improve the accuracy of downstream processing.

1 Introduction

In the past decade, single particle reconstruction (SPR) by cryo-electron microscopy (cryo-EM) has emerged as a critical technique for high resolution 3-D structure determination of

---

*Corresponding author, yunpengs@princeton.edu

macromolecules [1, 6, 10, 26, 29, 30]. In SPR, the 3-D structure needs to be determined from many noisy tomographic projection images with unknown viewing directions. Cryo-EM images are typically very noisy due to the limited electron dosage required to avoid significant radiation damage.

Mathematically, the formation of cryo-EM images can be summarized as follows. Let $\varphi(\boldsymbol{r})$ be the electrostatic potential of a molecule where $\boldsymbol{r} = (x, y, z)^T \in \mathbb{R}^3$. The i-th observed raw image $\boldsymbol{I}_i$ is modeled as

$$\boldsymbol{I}_i(x, y) = c_i \boldsymbol{h}_i * \int \varphi(\boldsymbol{R}_i^{-1} \boldsymbol{r}) dz + \boldsymbol{N}_i. \tag{1}$$

Namely, the molecule φ is first rotated by the rotation matrix $\boldsymbol{R}_i$, and followed by projection in the z-direction to form the 2-D clean projection image. Next, the clean image is convolved with the 2-D filter $\boldsymbol{h}_i$, often known as the point spread function, or the inverse Fourier transform of the contrast transfer function (CTF). The convolved i-th clean image is further rescaled by its amplitude contrast c_i. At last, additive noise $\boldsymbol{N}_i$ is applied to the resulting image (translations are omitted from (1) just for the sake of simplicity of exposition). The goal of SPR is to estimate φ from the set of observed noisy images $\{\boldsymbol{I}_i\}_{i \in [n]}$, where $[n] := \{1, 2, \ldots, n\}$.

The challenges of SPR lie in several different aspects. First, the noise term $\boldsymbol{N}_i$ typically has much larger magnitude than that of the clean signal, making the clean signal hard to distinguish from the noise even by the naked eye. As a result, a large number of particle images (10^4-10^6) is often required for reconstruction [6]. Second, the rotations $\{\boldsymbol{R}_i\}_{i \in [n]}$ are unknown. These additional unknown variables make the estimation of the 3-D volume difficult in the low signal-to-noise-ratio (SNR) regime. The third challenge arises from the CTF. Although the CTF can be estimated from the power spectrum of the micrograph [12, 21, 23, 38], CTF correction is a challenging deconvolution problem. The main reason is that the CTFs are highly oscillatory and have zeros at many frequencies. Those zero-crossings completely remove the information of the images at those frequencies. As a result, accurate CTF correction requires the usage of several images from different defocus groups (namely different CTFs), assuming that those CTFs have non-overlapping zero-crossings [4]. Last but not least, the underlying clean signals may have different scaling c_i. This amplitude variation is mainly due to the unevenness of the ice layers where the molecule samples reside [34]. Thicker ice layers increase inelastic scattering of electrons by ice, hence decreasing elastic scattering by the molecule and effectively weakening the signal, i.e., a smaller scaling coefficient c_i. The large variation of c_i may cause inaccurate image denoising and CTF correction. Moreover, the scale variations may severely affect the similarity measures used to detect images from similar viewing directions for 2-D class averaging [40, 5], 3-D heterogeneity analysis, and the identification of common lines for 3-D ab-initio modeling [2]. In particular, as pointed in Table 2 of [2], the uneven image contrast is the most important factor that negatively affects the accuracy of rotation estimation by some common line based approaches. At last, scaling variability must be accounted for 3-D heterogeneity analysis to prevent artificial classes that correspond to contrast variations [17]. In this work, we aim to address the last challenge, contrast estimation, in the *ab-initio* stage. In other words, we are interested in the direct estimation of c_i without estimating φ and $\boldsymbol{R}_i$. Furthermore, we use the improved contrast estimation to obtain better denoising and CTF correction of the images.

2 Related Work

There exist several works that estimate the amplitude contrast using estimated φ and $\boldsymbol{R}_i$'s [24, 27]. Specifically, assuming accurately estimated CTFs and given the estimates $\widehat{\varphi}$, $\widehat{\boldsymbol{R}}_i$, one can compute the estimated i-th CTF-effected clean image $\widehat{\boldsymbol{I}}_i := \widehat{\boldsymbol{h}}_i * \int \widehat{\varphi}(\widehat{\boldsymbol{R}}_i^{-1}\boldsymbol{r})dz$ and then c_i can be estimated as $\langle \boldsymbol{I}_i, \widehat{\boldsymbol{I}}_i \rangle / \|\widehat{\boldsymbol{I}}_i\|^2$. The estimates of c_i, $\boldsymbol{R}_i$ and φ are often iteratively refined using the EM algorithm [25]. Estimating c_i without any knowledge of rotations and 3-D structure is a challenging task. We refer to this task of contrast estimation as *ab-initio* contrast estimation (ACE). To the best of our knowledge, ACE has not been extensively studied in previous works. The mean pixel value of the CTF-corrected and denoised images can be used to approximate the contrast. However, [4] only uses the estimated contrasts to filter out junk particles (outliers), while the accuracy of contrast estimation itself was not tested. There are other contrast-related techniques. Image normalization [25, 31] rescales the images so that the background noise level is approximately the same across the images. However, its normalization factor depends on the noise level, not the amplitude contrast. There are also works on contrast enhancement [37, 20, 32]. These aim to enhance the brightness of the underlying signal so that it is more distinguishable from the noise. However, they do not directly estimate the amplitude contrast of the clean signal, and in the process they alter the image contrast.

There are also several commonly used ab-initio methods for simultaneous image denoising and CTF correction, such as traditional Wiener filtering (TWF), and covariance Wiener filtering (CWF) [4]. TWF denoises each image using its own information, which suffers from low SNR and zero-crossings in CTF. CWF overcomes these issues by estimating the population covariance of a set of images. However, its denoising performance degrades when the covariance is not accurately estimated. Image restoration can also be done by 2-D class averaging [40, 9, 15, 24]. These methods require pairwise comparison and alignment of images, unlike the preprocessing methods such as [4] and [7]. It is also shown in [7] that an appropriate image preprocessing can significantly improve the results of 2-D class averaging. Deep learning based methods were recently introduced for image denoising and enhancement [20, 3, 11, 33]. Noise2noise [3] requires multiple video frames of the same micrograph, which are not always available. Other CNN and GAN based methods [20, 11, 33] require clean projections to form clean-noisy pairs of images to train the model, but the clean projections are not available in the ab-initio reconstruction stage, and training with clean projections of other molecules may introduce model bias.

3 Methodology

In this work, we directly estimate the amplitude contrast from the CTF-corrected and denoised images in the ab-initio stage of SPR. Our method is based on CWF with additional constraints on the covariance matrix which we find realistic and useful for contrast estimation.

In order to address the ACE problem, we first derive from (1) a simplified image formation model that is independent of $\boldsymbol{R}_i$ and φ. We then propose our method to solve the ACE problem under this model.

3.1   A Simplified Image Formation Model

To demonstrate the simplified model, we first reshape the images in (1) as vectors and obtain

$$\boldsymbol{y}_i = c_i \boldsymbol{A}_i \boldsymbol{x}_i + \boldsymbol{\epsilon}_i \tag{2}$$

where $\boldsymbol{y}_i$ and $\boldsymbol{x}_i$ are respectively the vectors of i-th noisy and clean images, $\boldsymbol{A}_i$ is the square matrix operator corresponding to the convolution with $\boldsymbol{h}_i$, $\boldsymbol{\epsilon}_i$ is the Gaussian noise vector and c_i is the contrast to be estimated. In this model, c_i, $\boldsymbol{x}_i$ and $\boldsymbol{\epsilon}_i$ are unknown. However, the power spectral density (PSD) of $\boldsymbol{\epsilon}_i$ is assumed known as it can be estimated from the corners of the observed images y_i. We assume that $\boldsymbol{A}_i$ and its Fourier transform, the CTF, are known, since they can often be accurately estimated in advance from the noisy micrographs. Throughout this work, we assume that the CTFs are radially symmetric by ignoring astigmatism. Without loss of generality (WLOG) we assume that the noise distribution is white Gaussian whose covariance is $\sigma^2 \boldsymbol{I}$. For colored Gaussian noise, one can whiten the noise by applying $\boldsymbol{W}$ (noise covariance to the power $-1/2$) to $\boldsymbol{y}_i$, so that $\boldsymbol{W}\boldsymbol{y}_i = c_i \boldsymbol{W}\boldsymbol{A}_i \boldsymbol{x}_i + \boldsymbol{W}\boldsymbol{\epsilon}_i$ and the covariance of the whitened noise $W\epsilon_i$ is the identity matrix. The goal of ACE is to estimate c_i from the observed $\boldsymbol{y}_i$.

Without additional assumptions on $\boldsymbol{x}_i$ and c_i, the ACE problem is ill-posed due to the scale ambiguity of $\boldsymbol{x}_i$ and c_i. To make it a well-posed problem, WLOG we assume that $\boldsymbol{x}_i$ and c_i are random variables such that for all $i \in [n]$,

1. c_i and $\boldsymbol{x}_i$ are independent of each other.

2. $\mathbb{E}(c_i) = 1$.

3. $\boldsymbol{x}_i^\top \boldsymbol{1} = s$ for some constant $s > 0$, where $\boldsymbol{1}$ is the all-ones vector of the same size as $\boldsymbol{x}_i$.

The first assumption is reasonable since the contrast c_i primarily depends on the thickness of the ice layer, which is indeed independent of the rotation $\boldsymbol{R}_i$ and consequently independent of $\boldsymbol{x}_i$. The second assumption is needed to overcome the global scale ambiguity of c_i. The last assumption states that the sum of pixel values of the clean projection image is the same for all clean images. This is a reasonable assumption, because for each $i \in [n]$, the sum of the elements in $\boldsymbol{x}_i$ is approximately $\int \int (\int \varphi(\boldsymbol{R}_i^{-1}\boldsymbol{r})dz)dxdy = \int \varphi(\boldsymbol{r})d\boldsymbol{r}$ which is a constant independent of i. In other words, the sum of pixel values of any 2-D projection image equals the sum of 3-D voxel values. In fact, it is also invariant to translations (i.e., non-perfect centering of the images).

We note that our model assumes that c_i's are identically distributed, but it does not require the contrasts to be independent of each other. Namely, we allow correlations among c_i's, which is often observed in experimental data. For example, two particle images that are closely located in the same micrograph often share similar contrasts. In principle, this information can be used to improve the contrast estimation, but this is left to future work. We refer the reader to Figure 23 in Section 5.1 for further discussion.

We remark that the estimation of c_i remains challenging due to the CTF that affects the sum of pixel values and due to the high noise level. Thus, in order to obtain a good estimate of c_i, it is useful to denoise the image and to correct the CTF effect. A well-known method for such image restoration is CWF [4], which is elaborated in the next subsection.

3.2 Preliminaries: Covariance Wiener Filtering

CWF estimates $c_i \boldsymbol{x}_i$ from $\boldsymbol{y}_i$ by minimizing the expected mean squared error given an estimated covariance matrix of $c_i \boldsymbol{x}_i$. Assume that the true covariance matrix of $c_i x_i$, denoted by $\boldsymbol{\Sigma}_{cx}$, is given by an oracle. Then, under the model $\boldsymbol{y}_i = \boldsymbol{A}_i (c_i \boldsymbol{x}_i) + \boldsymbol{\epsilon}_i$, the linear minimum mean squared error (LMMSE) estimator is given by

$$\widehat{c_i \boldsymbol{x}_i}^{\mathrm{CWF}} = \mathrm{CWF}(\boldsymbol{y}_i, \boldsymbol{A}_i, \boldsymbol{\Sigma}_{cx}) = \arg\min_{\widehat{c_i \boldsymbol{x}_i}} \mathbb{E}(\|\widehat{c_i \boldsymbol{x}_i} - c_i \boldsymbol{x}_i\|^2 | \boldsymbol{y}_i) \tag{3}$$

$$= \boldsymbol{\mu} + \boldsymbol{\Sigma}_{cx} \boldsymbol{A}_i^\top (\boldsymbol{A}_i \boldsymbol{\Sigma}_{cx} \boldsymbol{A}_i^\top + \sigma^2 \boldsymbol{I})^{-1} (\boldsymbol{y}_i - \boldsymbol{A}_i \boldsymbol{\mu}), \tag{4}$$

where $\boldsymbol{\mu}$ is the true mean of $c_i \boldsymbol{x}_i$.

We note that CWF naturally induces an optimal linear estimator of contrast given the true covariance $\boldsymbol{\Sigma}_{cx}$. Indeed, it can be easily shown that :

$$\arg\min_{\boldsymbol{1}^\top \widehat{c_i \boldsymbol{x}_i}} \mathbb{E}(\|\boldsymbol{1}^\top \widehat{c_i \boldsymbol{x}_i} - \boldsymbol{1}^\top c_i \boldsymbol{x}_i\|^2 | \boldsymbol{y}_i)$$

$$= \boldsymbol{1}^\top \boldsymbol{\mu} + \boldsymbol{1}^\top \boldsymbol{\Sigma}_{cx} \boldsymbol{A}_i^\top (\boldsymbol{A}_i \boldsymbol{\Sigma}_{cx} \boldsymbol{A}_i^\top + \sigma^2 \boldsymbol{I})^{-1} (\boldsymbol{y}_i - \boldsymbol{A}_i \boldsymbol{\mu}) = \boldsymbol{1}^\top \widehat{c_i \boldsymbol{x}_i}^{\mathrm{CWF}}. \tag{5}$$

Namely, $\boldsymbol{1}^\top \widehat{c_i \boldsymbol{x}_i}^{\mathrm{CWF}}$, the sum of pixel values of the CWF estimate of $c_i \boldsymbol{x}_i$, is the best linear estimator of $\boldsymbol{1}^\top c_i \boldsymbol{x}_i$ given $\boldsymbol{\Sigma}_{cx}$. Note that $\boldsymbol{1}^\top c_i \boldsymbol{x}_i = s \cdot c_i$ by the third assumption of our model. Therefore, we have obtained the optimal linear estimator of contrast c_i up to a global constant s. This scale ambiguity can be solved by using the second assumption $\mathbb{E}(c_i) = 1$ in our model. That is, after obtaining the estimates of $s \cdot c_i$, we normalize the estimates by a global constant so that the average of the set $\{\boldsymbol{1}^\top \widehat{c_i \boldsymbol{x}_i}^{\mathrm{CWF}}\}_{i \in [n]}$ is 1.

However, the optimal properties of the aforementioned estimates only hold when $\boldsymbol{\Sigma}_{cx}$ is given, which is not true in practice. In [4], the mean of $c_i \boldsymbol{x}_i$ is estimated by minimizing the least squares error between the noisy images and the CTF transformed mean. Specifically,

$$\widehat{\boldsymbol{\mu}} = \arg\min_{\boldsymbol{\mu}} \sum_{i=1}^n \|\boldsymbol{y}_i - \boldsymbol{A}_i \boldsymbol{\mu}\|^2. \tag{6}$$

Similarly, the covariance matrix $\boldsymbol{\Sigma}_{cx}$ is estimated by minimizing the squared deviations between the sample covariance of $\boldsymbol{y}_i$ and the population covariance of $\boldsymbol{A}_i (c_i \boldsymbol{x}_i) + \boldsymbol{\epsilon}_i$. That is,

$$\widehat{\boldsymbol{\Sigma}}_{cx} = \arg\min_{\boldsymbol{\Sigma}} \sum_{i=1}^n \left\| (\boldsymbol{y}_i - \boldsymbol{A}_i \widehat{\boldsymbol{\mu}})(\boldsymbol{y}_i - \boldsymbol{A}_i \widehat{\boldsymbol{\mu}})^\top - (\boldsymbol{A}_i \boldsymbol{\Sigma} \boldsymbol{A}_i^\top + \sigma^2 \boldsymbol{I}) \right\|_F^2. \tag{7}$$

By setting the first order derivative of (7) to zero, we end up with the following linear system of equations:

$$\sum_{i=1}^n \boldsymbol{A}_i^\top \boldsymbol{A}_i \widehat{\boldsymbol{\Sigma}}_{cx} \boldsymbol{A}_i^\top \boldsymbol{A}_i = \sum_{i=1}^n \boldsymbol{A}_i^\top (\boldsymbol{y}_i - \boldsymbol{A}_i \widehat{\boldsymbol{\mu}})(\boldsymbol{y}_i - \boldsymbol{A}_i \widehat{\boldsymbol{\mu}})^\top \boldsymbol{A}_i + \sigma^2 \sum_{i=1}^n \boldsymbol{A}_i^\top \boldsymbol{A}_i. \tag{8}$$

We note that the first term on the right hand side (RHS) of (8) corresponds to the sample covariance of $\boldsymbol{A}_i^\top \boldsymbol{y}_i$. However, $\boldsymbol{y}_i$ often has dimension $> 10^4$ which is comparable to the number of images. In this high dimensional setting, the sample covariance is not a consistent estimator of the population covariance. As a result, an eigenvalue shrinkage method is

applied to the RHS of (8) to improve the covariance estimation. At last, (8) is solved by applying the conjugate gradient method. We refer the readers to [4] for more details.

We remark that under low SNR or insufficient number of samples, $\mathbf{\Sigma}_{cx}$ could be poorly estimated. In such a case, the CWF method and its induced contrast estimator (5) are far from being optimal. Therefore, there is still room for improvement on the CWF-based contrast estimation. Indeed, the CWF-estimator does not fully exploit our model assumptions. As we show in the next subsection, the three assumptions of our model imply novel constraints on $\mathbf{\Sigma}_{cx}$ which turn out to significantly improve contrast estimation.

3.3   Novel Constraints on the Covariance Matrices

The new constraints on the covariance matrix are stated in the following proposition. Let $\mathbf{\Sigma}_x$ be the true covariance of $\boldsymbol{x}_i$ and $\mathrm{Var}(c)$ be the variance of each c_i.

Proposition 1. *If the three assumptions for the model (2) are satisfied, then the following two constraints hold:*

$$1. \quad \mathbf{\Sigma}_{cx} = (\mathit{Var}(c) + 1)\mathbf{\Sigma}_x + \mathit{Var}(c)\boldsymbol{\mu}\boldsymbol{\mu}^\top \qquad (9)$$

$$2. \quad \mathbf{\Sigma}_x \mathbf{1} = \mathbf{0}. \qquad (10)$$

This proposition suggests that the true covariance of $c_i\boldsymbol{x}_i$ is the combination of two components, one corresponds to the covariance without contrast variability whose eigenvectors are perpendicular to the all-ones vector, and the other corresponds to a rank-one matrix whose eigenvector is the mean. The derivation of the two constraints is simple. To prove the first constraint, we use the identity that for any independent scalar random variable c and random vector $\boldsymbol{x}$, $\mathrm{Cov}(c\boldsymbol{x}) = \mathbb{E}(c^2)\mathrm{Cov}(\boldsymbol{x}) + \mathrm{Var}(c)\mathbb{E}(\boldsymbol{x})\mathbb{E}(\boldsymbol{x})^\top$. By letting $c = c_i$ and $\boldsymbol{x} = \boldsymbol{x}_i$, we obtain that for any $i \in [n]$

$$\begin{aligned} \mathbf{\Sigma}_{cx} = \mathrm{Cov}(c\boldsymbol{x}) &= \mathbb{E}(c^2)\mathrm{Cov}(\boldsymbol{x}) + \mathrm{Var}(c)\mathbb{E}(\boldsymbol{x})\mathbb{E}(\boldsymbol{x})^\top \\ &= (\mathrm{Var}(c) + \mathbb{E}^2(c))\mathbf{\Sigma}_x + \mathrm{Var}(c)\boldsymbol{\mu}\boldsymbol{\mu}^\top. \end{aligned}$$

By using the assumption $\mathbb{E}(c_i) = 1$, we conclude the first constraint. The second constraint states that the variation of the sum of elements in $\boldsymbol{x}_i$ is 0, namely the contrast variability of clean signals is 0. It can be verified easily by using the third assumption of our model. Specifically,

$$\mathbf{\Sigma}_x \mathbf{1} = \mathbb{E}\left((\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^\top\right)\mathbf{1} = \mathbb{E}\left((\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i^\top \mathbf{1} - \boldsymbol{\mu}^\top \mathbf{1})\right) = \mathbf{0},$$

where the last equality follows from the assumption that : $\boldsymbol{x}_i^\top \mathbf{1} = \boldsymbol{\mu}^\top \mathbf{1} = s$. In the next subsection, we propose two methods that use the two covariance constraints to refine the estimated covariance $\mathbf{\Sigma}_{cx}$.

3.4   Refinement of Covariance Matrices

We first use the two covariance constraints (9) and (10) to estimate the contrast variance $\mathrm{Var}(c)$. By combining the two constraints,

$$\mathbf{\Sigma}_{cx}\mathbf{1} = (\mathrm{Var}(c) + 1)\mathbf{\Sigma}_x\mathbf{1} + \mathrm{Var}(c)\boldsymbol{\mu}\boldsymbol{\mu}^\top\mathbf{1} = \mathrm{Var}(c)\boldsymbol{\mu}\boldsymbol{\mu}^\top\mathbf{1}, \qquad (11)$$

where the second equality follows from the second covariance constraint. We note that (11) relates $\mathrm{Var}(c)$ to $\mathbf{\Sigma}_{cx}$ and $\boldsymbol{\mu}$, where the latter two can be estimated from the noisy data.

Given the estimated $\widehat{\boldsymbol{\Sigma}}_{cx}$ and $\widehat{\boldsymbol{\mu}}$, the variance of the image contrast can be estimated by least squares as follows.

$$\widehat{\mathrm{Var}(c)} = \underset{\mathrm{Var}(c)}{\arg\min} \| \widehat{\boldsymbol{\Sigma}}_{cx} \mathbf{1} - \mathrm{Var}(c)(\widehat{\boldsymbol{\mu}}^{\top} \mathbf{1}) \widehat{\boldsymbol{\mu}} \|_2^2 = \frac{\widehat{\boldsymbol{\mu}}^{\top} \widehat{\boldsymbol{\Sigma}}_{cx} \mathbf{1}}{\| \widehat{\boldsymbol{\mu}} \|^2 \widehat{\boldsymbol{\mu}}^{\top} \mathbf{1}}. \tag{12}$$

We remark that the initially estimated $\widehat{\boldsymbol{\Sigma}}_{cx}$ often does not satisfy the constraints in Proposition 1. Therefore, we introduce two methods to enforce the covariance constraints using the estimated $\widehat{\mathrm{Var}(c)}$. We refer to the first method as semi-definite programming (SDP).

SDP: We seek to find the closest positive semidefinite matrix to the initially estimated $\widehat{\boldsymbol{\Sigma}}_{cx}$ such that the two covariance constraints are satisfied. Namely, we seek a solution of the following SDP problem.

$$\widehat{\boldsymbol{\Sigma}}_{cx}^{\mathrm{SDP}} = \underset{\boldsymbol{\Sigma}_{cx}^{\mathrm{SDP}}}{\arg\min} \left\| \boldsymbol{\Sigma}_{cx}^{\mathrm{SDP}} - \widehat{\boldsymbol{\Sigma}}_{cx} \right\|_F^2 \tag{13}$$

$$\text{subject to} \quad \boldsymbol{\Sigma}_{cx}^{\mathrm{SDP}} = (\widehat{\mathrm{Var}(c)} + 1)\boldsymbol{\Sigma}_x^{\mathrm{SDP}} + \widehat{\mathrm{Var}(c)}\widehat{\boldsymbol{\mu}}\widehat{\boldsymbol{\mu}}^{\top}$$

$$\boldsymbol{\Sigma}_{cx}^{\mathrm{SDP}} \succeq \mathbf{0}$$

$$\boldsymbol{\Sigma}_x^{\mathrm{SDP}} \succeq \mathbf{0}$$

$$\boldsymbol{\Sigma}_x^{\mathrm{SDP}} \mathbf{1} = \mathbf{0}.$$

Since $\boldsymbol{\Sigma}_{cx}^{\mathrm{SDP}} = (\widehat{\mathrm{Var}(c)} + 1)\boldsymbol{\Sigma}_x^{\mathrm{SDP}} + \widehat{\mathrm{Var}(c)}\widehat{\boldsymbol{\mu}}\widehat{\boldsymbol{\mu}}^{\top}$, $\boldsymbol{\Sigma}_{cx}^{\mathrm{SDP}}$ is positive semidefinite when $\boldsymbol{\Sigma}_x^{\mathrm{SDP}}$ is so. Thus, one can drop the constraint $\boldsymbol{\Sigma}_{cx}^{\mathrm{SDP}} \succeq \mathbf{0}$, and plug in the first constraint of (13) to $\boldsymbol{\Sigma}_{cx}^{\mathrm{SDP}}$ in its objective function. This yields the following SDP fomulation that optimizes for $\boldsymbol{\Sigma}_x^{\mathrm{SDP}}$.

$$\widehat{\boldsymbol{\Sigma}}_x^{\mathrm{SDP}} = \underset{\boldsymbol{\Sigma}_x^{\mathrm{SDP}}}{\arg\min} \left\| \boldsymbol{\Sigma}_x^{\mathrm{SDP}} - \frac{\widehat{\boldsymbol{\Sigma}}_{cx} - \widehat{\mathrm{Var}(c)}\widehat{\boldsymbol{\mu}}\widehat{\boldsymbol{\mu}}^{\top}}{\widehat{\mathrm{Var}(c)} + 1} \right\|_F^2 \tag{14}$$

$$\text{subject to} \quad \boldsymbol{\Sigma}_x^{\mathrm{SDP}} \mathbf{1} = \mathbf{0}$$

$$\boldsymbol{\Sigma}_x^{\mathrm{SDP}} \succeq \mathbf{0}.$$

After solving $\widehat{\boldsymbol{\Sigma}}_x^{\mathrm{SDP}}$, we immediately obtain $\widehat{\boldsymbol{\Sigma}}_{cx}^{\mathrm{SDP}} = (\widehat{\mathrm{Var}(c)} + 1)\widehat{\boldsymbol{\Sigma}}_x^{\mathrm{SDP}} + \widehat{\mathrm{Var}(c)}\widehat{\boldsymbol{\mu}}\widehat{\boldsymbol{\mu}}^{\top}$. Next, we introduce a faster but heuristic alternative that uses the Gram-Schmidt (GS) process to approximately solve (14).

Gram-Schmidt (GS) Process: We first note that the constraint $\boldsymbol{\Sigma}_x^{\mathrm{SDP}} \mathbf{1} = \mathbf{0}$ in (14) is equivalent to that all the eigenvectors of $\boldsymbol{\Sigma}_x^{\mathrm{SDP}}$ are orthogonal to $\mathbf{1}$. Similar to (14), we seek a positive semidefinite matrix $\boldsymbol{\Sigma}_x^{\mathrm{GS}}$ that is close to $\widehat{\boldsymbol{\Sigma}}_x := (\widehat{\boldsymbol{\Sigma}}_{cx} - \widehat{\mathrm{Var}(c)}\widehat{\boldsymbol{\mu}}\widehat{\boldsymbol{\mu}}^{\top})/(\widehat{\mathrm{Var}(c)} + 1)$ whose eigenvectors are orthogonal to $\mathbf{1}$.

Let $\widehat{\boldsymbol{\Sigma}}_x = \widehat{\boldsymbol{V}}\widehat{\boldsymbol{D}}\widehat{\boldsymbol{V}}^{\top}$ be the eigenvalue decomposition of $\widehat{\boldsymbol{\Sigma}}_x$, where $\widehat{\boldsymbol{V}}$ is the eigenmatrix whose columns are eigenvectors of $\widehat{\boldsymbol{\Sigma}}_x$, and $\widehat{\boldsymbol{D}}$ is the diagonal matrix of its eigenvalues. We seek a refined covariance $\widehat{\boldsymbol{\Sigma}}_x^{\mathrm{GS}} = \widehat{\boldsymbol{U}}\widehat{\boldsymbol{\Lambda}}\widehat{\boldsymbol{U}}^{\top}$ with refined eigenvalues and eigenvectors such that $\widehat{\boldsymbol{\Lambda}}$ is nonnegative (so that $\widehat{\boldsymbol{\Sigma}}_x^{\mathrm{GS}} \succeq \mathbf{0}$) and $\widehat{\boldsymbol{U}}^{\top}\mathbf{1} = \mathbf{0}$ (so that $(\widehat{\boldsymbol{\Sigma}}_x^{\mathrm{GS}})^{\top}\mathbf{1} = \mathbf{0}$), and $\widehat{\boldsymbol{\Lambda}}$ and $\widehat{\boldsymbol{U}}$ are respectively close to $\widehat{\boldsymbol{D}}$ and $\widehat{\boldsymbol{V}}$.

The solution of $\widehat{\boldsymbol{\Lambda}}$ is obtained by simply thresholding the negative values in $\widehat{\boldsymbol{D}}$. That is, $\widehat{\boldsymbol{\Lambda}} = \max(\widehat{\boldsymbol{D}}, 0)$. The solution of the eigenmatrix $\widehat{\boldsymbol{U}}$ is trickier, due to the nonconvex constraint $\widehat{\boldsymbol{U}}^\top \widehat{\boldsymbol{U}} = \boldsymbol{I}$, namely the columns of $\widehat{\boldsymbol{U}}$ (the eigenvectors of $\widehat{\boldsymbol{\Sigma}}_x^{\mathrm{GS}}$) form an orthonormal basis. It asks to solve the following nonconvex optimization problem.

$$\widehat{\boldsymbol{U}} = \arg\max_{\boldsymbol{U}} \ \mathrm{Tr}(\widehat{\boldsymbol{V}}^\top \boldsymbol{U}) \tag{15}$$
$$\text{subject to} \quad \boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}$$
$$\boldsymbol{U}^\top \boldsymbol{1} = \boldsymbol{0}.$$

Instead of directly solving (15), we argue that a simple Gram-Schmidt process on $\widehat{\boldsymbol{V}}$ is often sufficient to obtain a satisfying solution. Let $\widehat{\boldsymbol{V}} = [\widehat{\boldsymbol{v}}_1, \widehat{\boldsymbol{v}}_2, \ldots, \widehat{\boldsymbol{v}}_p]$ where p is the dimension of each $\boldsymbol{x}_i$, and the eigenvectors are placed in descending order of eigenvalues.

Let $\widehat{\boldsymbol{U}} := [\widehat{\boldsymbol{u}}_1, \widehat{\boldsymbol{u}}_2, \ldots, \widehat{\boldsymbol{u}}_p]$ and $[\boldsymbol{1}, \widehat{\boldsymbol{V}}_{-p}] := [\boldsymbol{1}, \widehat{\boldsymbol{v}}_1, \widehat{\boldsymbol{v}}_2, \ldots, \widehat{\boldsymbol{v}}_{p-1}]$ be p-by-p square matrices. Application of Gram-Schmidt orthogonalization to $[\boldsymbol{1}, \widehat{\boldsymbol{V}}_{-p}]$ yields a new orthogonal matrix $[\boldsymbol{1}, \widehat{\boldsymbol{U}}_{-p}] := [\boldsymbol{1}, \widehat{\boldsymbol{u}}_1, \widehat{\boldsymbol{u}}_2, \ldots, \widehat{\boldsymbol{u}}_{p-1}]$. That is, $\widehat{\boldsymbol{u}}_1$ is computed by projecting $\widehat{\boldsymbol{v}}_1$ onto the orthogonal complement of $\boldsymbol{1}$ and then normalize to unit vector. Once $\widehat{\boldsymbol{u}}_{i-1}$ for $1 \leq i \leq p-1$ are computed, $\widehat{\boldsymbol{u}}_i$ is computed by projecting $\widehat{\boldsymbol{v}}_i$ onto the orthogonal complement of linear subspace spanned by $\boldsymbol{1}, \widehat{\boldsymbol{u}}_1, \ldots, \widehat{\boldsymbol{u}}_{i-1}$ and then normalize. At last, the solution $\widehat{\boldsymbol{U}}$ is obtained by finding the orthogonal complement of $\widehat{\boldsymbol{U}}_{-p}$ to complete its missing column $\widehat{\boldsymbol{u}}_p$. In this way, the columns of $\widehat{\boldsymbol{U}}$ form an orthonormal basis, and its first $p-1$ columns are orthogonal to $\boldsymbol{1}$. Although $\widehat{\boldsymbol{u}}_p$ may not necessarily be orthogonal to $\boldsymbol{1}$, its eigenvalue is 0 in most of the cases and thus won't affect the solution of $\widehat{\boldsymbol{\Sigma}}_x^{\mathrm{GS}}$. In practice, the GS process is done by the QR decomposition for its better numerical stability.

Iterating from the top eigenvectors has two benefits. First, the top eigenvectors of $\widehat{\boldsymbol{\Sigma}}_x$ are more robust to the noise. That is, the top eigenvectors $\widehat{\boldsymbol{v}}_1, \widehat{\boldsymbol{v}}_2, \ldots$ of $\widehat{\boldsymbol{\Sigma}}_x$ are often closer to those of the true $\boldsymbol{\Sigma}_x$. Due to the constraint $\boldsymbol{\Sigma}_x \boldsymbol{1} = \boldsymbol{0}$, the top eigenvectors of $\widehat{\boldsymbol{\Sigma}}_x$ often have smaller correlation with $\boldsymbol{1}$. In other words, the top eigenvectors are cleaner and thus their refinement is easier and more accurate, and therefore they should be put at the earlier stage of the sequential projection procedure to reduce error accumulation. Second, iterating from the top eigenvectors makes them more accurately projected to the orthogonal complement of $\boldsymbol{1}$ with minimal changes to their original values. This is beneficial for contrast estimation since these top eigenvectors are more important for explaining the contrast variations.

3.5 Ab-initio Contrast Estimation and Denoising

After applying the aforementioned SDP or GS method to the initial covariance matrix $\widehat{\boldsymbol{\Sigma}}_{cx}$, we obtain the refined covariance $\widehat{\boldsymbol{\Sigma}}_{cx}^{\mathrm{RF}} = \widehat{\boldsymbol{\Sigma}}_{cx}^{\mathrm{SDP}}$ or $\widehat{\boldsymbol{\Sigma}}_{cx}^{\mathrm{GS}}$.

Recall that the CWF estimator of $c_i \boldsymbol{x}_i$ is defined in (3). Our refined estimate of $c_i \boldsymbol{x}_i$ for each $i \in [n]$ is

$$\widehat{c_i \boldsymbol{x}_i}^{\mathrm{RF}} = \mathrm{CWF}\left(\boldsymbol{y}_i, \boldsymbol{A}_i, \widehat{\boldsymbol{\Sigma}}_{cx}^{\mathrm{RF}}\right). \tag{16}$$

Then, by applying our model assumptions that $\boldsymbol{1}^\top \boldsymbol{x}_i = s$ for all $i \in [n]$ and $\mathbb{E}(c_i) = 1$, we obtain our refined estimate of c_i as

$$\widehat{c}_i^{\mathrm{RF}} = \frac{\boldsymbol{1}^\top \widehat{c_i \boldsymbol{x}_i}^{\mathrm{RF}}}{\frac{1}{n} \sum_{i=1}^{n} \boldsymbol{1}^\top \widehat{c_i \boldsymbol{x}_i}^{\mathrm{RF}}}, \tag{17}$$

8

where the numerator is the contrast estimator up to s. The denominator is the normalization factor to remove the scale ambiguity and enforce the average of $\widehat{c}_i^{\,\mathrm{RF}}$ to be 1.

After estimating the contrasts, we present two methods for estimating the clean image $\boldsymbol{x}_i$: the image normalization and 2-stage CWF.

Image Normalization
In the first approach, we estimate $\boldsymbol{x}_i$ as

$$\widehat{\boldsymbol{x}}_i^{\mathrm{RF}} = \frac{\widehat{c_i \boldsymbol{x}_i}^{\mathrm{RF}}}{\widehat{c}_i^{\,\mathrm{RF}}}. \tag{18}$$

That is, we simply normalize the estimated $c_i \boldsymbol{x}_i$ by the estimated contrast c_i, so that the resulting images all share the same sum of pixel values.

2-Stage CWF
In the second method, we apply an additional CWF estimator to directly estimate $\boldsymbol{x}_i$. Recall that the original version of CWF aims to estimate $c_i \boldsymbol{x}_i$. It treats $c_i \boldsymbol{x}_i$ as a single variable and considers the model $\boldsymbol{y}_i = \boldsymbol{A}_i(c_i \boldsymbol{x}_i) + \boldsymbol{\epsilon}_i$. In order to directly estimate $\boldsymbol{x}_i$, we treat c_i as known and absorb it into the CTF term, and consider the model $\boldsymbol{y}_i = (\widehat{c}_i^{\,\mathrm{RF}} \boldsymbol{A}_i)\boldsymbol{x}_i + \boldsymbol{\epsilon}_i$. Given this model, a natural estimate of $\boldsymbol{x}_i$ is

$$\widehat{\boldsymbol{x}}_i^{\mathrm{RF}} = \mathrm{CWF}\left(\boldsymbol{y}_i, \, \widehat{c}_i^{\,\mathrm{RF}} \boldsymbol{A}_i, \, \widehat{\boldsymbol{\Sigma}}_x^{\mathrm{RF}}\right). \tag{19}$$

Since the refined $\widehat{\boldsymbol{\Sigma}}_x^{\mathrm{RF}}$ satisfies the constraint $\widehat{\boldsymbol{\Sigma}}_x^{\mathrm{RF}} \mathbf{1} = \mathbf{0}$, the resulting recovered image $\widehat{\boldsymbol{x}}_i^{\mathrm{RF}}$ automatically has the same sum of pixel values. Ideally, if $\widehat{c}_i^{\,\mathrm{RF}} = c_i$ and $\widehat{\boldsymbol{\Sigma}}_x^{\mathrm{RF}} = \boldsymbol{\Sigma}_x$, then (19) is the optimal linear estimator of $\boldsymbol{x}_i$.

3.6 Computational Issues and Steerable Basis

Although our model and methodology were presented in real image space for simplicity, in practice, implementing the CWF-based methods in real image domain is computationally intractable and memory demanding. For images of size $L \times L$, the dimension of $\boldsymbol{x}_i$ is of order $O(L^2)$, and the covariance of $\boldsymbol{x}_i$ in real space has $O(L^4)$ entries. This leads to high time and space complexities that make the computation impractical.

Therefore, we follow [4] and expand the Fourier transformed images $\mathcal{F}(\boldsymbol{I}_i)$ in the Fourier-Bessel basis

$$\psi_r^{k,q}(\theta, \xi) = \begin{cases} N_{k,q} J_k(R_{k,q}\xi/r)e^{\imath k\theta} & \xi \leq r \\ 0 & \text{otherwise,} \end{cases} \tag{20}$$

where $0 < r \leq 1/2$ is the band-limit radius of images (default $= 1/2$), k, q are respectively angular and radial frequencies, (ξ, θ) are the polar coordinates in Fourier domain, J_k is the Bessel function of the first kind of order k, $R_{k,q}$ is the q-th root of J_k, and $N_{k,q} = (r\sqrt{\pi}|J_{k+1}(R_{k,q})|)^{-1}$ is the normalization factor. We refer the readers to [39] for details of the expansion.

Denote the image formation model in the Fourier-Bessel basis as

$$\boldsymbol{y}_i^{\mathrm{FB}} = c_i \boldsymbol{A}_i^{\mathrm{FB}} \boldsymbol{x}_i^{\mathrm{FB}} + \boldsymbol{\epsilon}_i$$

where $\boldsymbol{A}_i^{\mathrm{FB}}$, $\boldsymbol{x}_i^{\mathrm{FB}}$ $\boldsymbol{y}_i^{\mathrm{FB}}$ are respectively the CTF, and the clean and noisy Fourier transformed images in the Fourier-Bessel basis. Expanding images in Fourier-Bessel basis (or other steerable basis) enjoys some nice properties. For example, image rotation in the Fourier-Bessel domain is easy. Indeed, rotation of images corresponds to phase modulation of their corresponding Fourier-Bessel coefficients. This relationship between rotation and phase modulation enables easy and fast computation of the covariance matrix of any set of images that are augmented by all their possible in-plane rotations and reflections. More importantly, as shown in [39], the resulting covariance matrix of the augmented images is block diagonal, where blocks are indexed by the angular frequency k. That is, the $((k_1, q_1), (k_2, q_2))$-th entry of the covariance matrix is nonzero only when the angular frequencies are equal, namely $k_1 = k_2$. This reduces the number of variables in the covariance matrix from $O(L^4)$ to $O(L^3)$ which is a significant saving of computation time and memory usage. Similarly, a radially symmetric CTF in the Fourier-Bessel basis is also block diagonal and has the same block structure as that of the covariance. As a result, the estimation of each diagonal block of $\boldsymbol{x}_i^{\mathrm{FB}}$ is completely independent and decoupled from the rest of the blocks. Thus, the task of estimating $\boldsymbol{x}_i^{\mathrm{FB}}$ is divided into $O(L)$ independent tasks of much smaller sizes, which enables faster and parallelized computation.

Although the Fourier-Bessel expansion facilitates fast computation of CWF, our model and covariance refinement method require more careful adaptation to the Fourier Bessel domain. The main issue is that the Fourier-Bessel transform preserves the ℓ_2 norm by Parseval's identity, but not the sum of pixel values. As a result, $\mathbf{1}^\top \boldsymbol{x}_i^{\mathrm{FB}} = s$ and $\boldsymbol{\Sigma}_x^{\mathrm{FB}} \mathbf{1} = \mathbf{0}$ are not necessarily satisfied.

To address this issue, we observe that

$$\mathbf{1}^\top \boldsymbol{x}_i = \mathbf{1}^\top \boldsymbol{F}^* \boldsymbol{F}_B^* \boldsymbol{F}_B \boldsymbol{F} \boldsymbol{x}_i = \mathbf{1}_{\mathrm{FB}}^\top \boldsymbol{x}_i^{\mathrm{FB}}, \tag{21}$$

where $\boldsymbol{F}, \boldsymbol{F}_B$ are the matrix operators of the Fourier and Fourier-Bessel transforms, $\boldsymbol{F}^*, \boldsymbol{F}_B^*$ denote their corresponding adjoint operators, $\mathbf{1}_{\mathrm{FB}} = \boldsymbol{F}_B \boldsymbol{F} \mathbf{1}$ is the Fourier-Bessel transform of the Fourier-transformed all-ones image. Therefore, the new constraints in the Fourier-Bessel domain are $\mathbf{1}_{\mathrm{FB}}^\top \boldsymbol{x}_i^{\mathrm{FB}} = s$ and $\boldsymbol{\Sigma}_x^{\mathrm{FB}} \mathbf{1}_{\mathrm{FB}} = \mathbf{0}$. By replacing every $\mathbf{1}$ in the previous formulation by $\mathbf{1}_{\mathrm{FB}}$, exactly the same argument automatically follows in the Fourier-Bessel domain.

We finally remark that $\mathbf{1}_{\mathrm{FB}}$ is only nonzero in the zero-th angular frequency. Indeed, $\boldsymbol{F}\mathbf{1}$ is the dirac delta image $\boldsymbol{I}_\delta$ whose only nonzero pixel is located at the origin. Let $\mathbf{1}_{k,q}$ be the (k, q)-th coefficient of $\mathbf{1}_{\mathrm{FB}}$.

$$\mathbf{1}_{k,q} = \int \int \boldsymbol{I}_\delta(\theta, \xi) \overline{\psi_r^{k,q}(\theta, \xi)} \xi d\xi d\theta = \overline{\psi_r^{k,q}(0,0)} = N_{k,q} J_k(0) \tag{22}$$

$$= \begin{cases} N_{0,q} = \frac{1}{r\sqrt{\pi}|J_1(R_{0,q})|} & \text{for } k = 0, \\ 0 & \text{otherwise,} \end{cases} \tag{23}$$

where the last equality follows from that: $J_k(0) = 1$ when $k = 0$ and $J_k(0) = 0$ for $k \neq 0$. In view of (21) and (23), the contrasts of the real images are only determined by the zero-th angular blocks of their Fourier-Bessel expansion. This is a favorable property from the computational aspect. For example, when computing the numerator of (17), one can simply take the dot product between the zero-th angular frequency blocks of the Fourier-Bessel coefficients of $\mathbf{1}$ and $\widehat{c_i \boldsymbol{x}_i}^{\mathrm{RF}}$, without the need to access the entire vectors.

3.7 Summary of the Algorithm and Computational Complexity

Our ACE and image restoration methods are respectively summarized in Algorithms 1 and 2.

Algorithm 1 Ab-initio Contrast Estimation

Input: $\{A_i\}_{i\in[n]}$, $\{\boldsymbol{y}_i\}_{i\in[n]}$, option=SDP or GS

$\quad \boldsymbol{A}_{i,0} \leftarrow$ extract the zero-th angular frequency diagonal block of $\boldsymbol{A}_i$ for $i \in [n]$

$\quad \boldsymbol{y}_{i,0} \leftarrow$ extract the entries of $\boldsymbol{y}_i$ corresponding to the zero-th angular frequency for $i \in [n]$

$\quad \widehat{\boldsymbol{\mu}}_0 \leftarrow \arg\min_{\boldsymbol{\mu}} \sum_{i=1}^n \|\boldsymbol{y}_{i,0} - \boldsymbol{A}_{i,0}\boldsymbol{\mu}\|^2$

$\quad \widehat{\boldsymbol{\Sigma}}_{cx,0} \leftarrow \arg\min_{\boldsymbol{\Sigma}} \sum_{i=1}^n \left\| (\boldsymbol{y}_{i,0} - \boldsymbol{A}_{i,0}\widehat{\boldsymbol{\mu}}_0)(\boldsymbol{y}_{i,0} - \boldsymbol{A}_{i,0}\widehat{\boldsymbol{\mu}}_0)^\top - (\boldsymbol{A}_{i,0}\boldsymbol{\Sigma}\boldsymbol{A}_{i,0}^\top + \sigma^2\boldsymbol{I}) \right\|_F^2$

$\quad \widehat{\mathrm{Var}(c)} \leftarrow \widehat{\boldsymbol{\mu}}_0^\top \widehat{\boldsymbol{\Sigma}}_{cx,0}\boldsymbol{1}_0 / (\|\widehat{\boldsymbol{\mu}}_0\|^2 \widehat{\boldsymbol{\mu}}_0^\top \boldsymbol{1}_0)$

$\quad \widehat{\boldsymbol{\Sigma}}_{x,0} \leftarrow (\widehat{\boldsymbol{\Sigma}}_{cx,0} - \widehat{\mathrm{Var}(c)}\widehat{\boldsymbol{\mu}}_0\widehat{\boldsymbol{\mu}}_0^\top)/(\widehat{\mathrm{Var}(c)} + 1)$

$\quad$ **if** option=SDP **then**

$\quad\quad \widehat{\boldsymbol{\Sigma}}_{x,0}^{\mathrm{RF}} \leftarrow \arg\min_{\boldsymbol{\Sigma}_{x,0}^{\mathrm{SDP}}} \left\| \boldsymbol{\Sigma}_{x,0}^{\mathrm{SDP}} - \widehat{\boldsymbol{\Sigma}}_{x,0} \right\|_F^2 \quad$ subject to $\quad \boldsymbol{\Sigma}_{x,0}^{\mathrm{SDP}}\boldsymbol{1}_0 = \boldsymbol{0} \quad \boldsymbol{\Sigma}_{x,0}^{\mathrm{SDP}} \succeq \boldsymbol{0}$

$\quad$ **else**

$\quad\quad (\widehat{\boldsymbol{d}}_i, \widehat{\boldsymbol{v}}_i)_{i=1}^p \leftarrow$ pairs of eigenvalues/eigenvectors of $\widehat{\boldsymbol{\Sigma}}_{x,0}$ sorted in descending order

$\quad\quad \widehat{\boldsymbol{D}}_0 \leftarrow \mathrm{diag}(\max(\widehat{\boldsymbol{d}}_i, 0))$

$\quad\quad \widehat{\boldsymbol{V}}_0 \leftarrow [\boldsymbol{1}_0, \widehat{\boldsymbol{v}}_1, \widehat{\boldsymbol{v}}_2, \ldots, \widehat{\boldsymbol{v}}_{p-1}]$

$\quad\quad [\boldsymbol{1}_0, \widehat{\boldsymbol{U}}_0] \leftarrow \mathrm{Gram\text{-}Schmidt}(\widehat{\boldsymbol{V}}_0)$

$\quad\quad \widehat{\boldsymbol{U}}_0 \leftarrow [\widehat{\boldsymbol{U}}_0, \boldsymbol{v}_p]$

$\quad\quad \widehat{\boldsymbol{\Sigma}}_{x,0}^{\mathrm{RF}} \leftarrow \widehat{\boldsymbol{U}}_0 \widehat{\boldsymbol{D}}_0 \widehat{\boldsymbol{U}}_0^\top$

$\quad$ **end if**

$\quad \widehat{\boldsymbol{\Sigma}}_{cx,0}^{\mathrm{RF}} \leftarrow (\widehat{\mathrm{Var}(c)} + 1)\boldsymbol{\Sigma}_{x,0}^{\mathrm{RF}} + \widehat{\mathrm{Var}(c)}\widehat{\boldsymbol{\mu}}_0\widehat{\boldsymbol{\mu}}_0^\top$

$\quad \widehat{c_i\boldsymbol{x}_{i,0}}^{\mathrm{RF}} \leftarrow \mathrm{CWF}\left(\boldsymbol{y}_{i,0}, \boldsymbol{A}_{i,0}, \widehat{\boldsymbol{\Sigma}}_{cx,0}^{\mathrm{RF}}\right)$

$\quad \widehat{c}_i^{\mathrm{RF}} \leftarrow n\boldsymbol{1}_0^\top \widehat{c_i\boldsymbol{x}_{i,0}}^{\mathrm{RF}} / \sum_{i=1}^n \boldsymbol{1}_0^\top \widehat{c_i\boldsymbol{x}_{i,0}}^{\mathrm{RF}}$

Output: $\{\widehat{c}_i^{\mathrm{RF}}\}_{i\in[n]}$

We comment on the computational complexity of our methods. From [4], the overall complexity for the original CWF is $O(TDL^4 + nL^3)$. The first term corresponds to covariance estimation, where T is the number of conjugate gradient iterations for estimating $\boldsymbol{\Sigma}_{cx}$ and D is the number of defocus groups. The second term corresponds to denoising by Wiener filtering. Our contrast estimator takes two additional steps that cost extra computation. The covariance refinement by GS process takes $O(L^3)$ operations due to the eigenvalue decomposition of the diagonal block of $\boldsymbol{\Sigma}_x$ corresponding to the zero-th angular frequency. This step is negligible compared to the computational complexity of CWF. In contrast, the SDP method suffers from much higher complexity. For both splitting conic solver (SCS) [19] and interior point method [13], the per-iteration complexity is $O(m^3)$ where $m = L^2$ is the number of variables in the diagonal block of $\boldsymbol{\Sigma}_x$ corresponding to the zero-th angular frequency. Therefore, the total complexity of our SDP method (using the aforementioned solvers) is $O(L^6)$. Nevertheless, in our synthetic and experimental data $L < 400$ so empirically we observe that SDP can still be implemented in less than one hour. The step of estimating the image contrasts using the Fourier-Bessel basis (eq. (21)) requires $O(nL)$ operations, which is negligible compared to the cost of CWF. In summary, our method with GS-refinement of covariance has similar complexity to that of the original CWF. Our

---
Algorithm 2 Ab-initio Image Restoration

Input: $\{A_i\}_{i\in[n]}$, $\{\boldsymbol{y}_i\}_{i\in[n]}$, option=normalization or 2-stage

   $\widehat{\mu}$, $\widehat{\boldsymbol{\Sigma}}_{cx} \leftarrow$ by solving (6), (7)

   $\{\widehat{c}_i^{\,\mathrm{RF}}\}_{i\in[n]}, \widehat{\boldsymbol{\Sigma}}_{cx,0}^{\mathrm{RF}}, \widehat{\boldsymbol{\Sigma}}_{x,0}^{\mathrm{RF}} \leftarrow$ by implementing Algorithm 1

   $\widehat{\boldsymbol{\Sigma}}_{cx}^{\mathrm{RF}} \leftarrow$ replace the zero-th angular frequency diagonal block of $\widehat{\boldsymbol{\Sigma}}_{cx}$ by $\widehat{\boldsymbol{\Sigma}}_{cx,0}^{\mathrm{RF}}$

   $\widehat{\boldsymbol{\Sigma}}_x^{\mathrm{RF}} \leftarrow$ replace the zero-th angular frequency diagonal block of $\widehat{\boldsymbol{\Sigma}}_{cx}$ by $\widehat{\boldsymbol{\Sigma}}_{x,0}^{\mathrm{RF}}$

   if option = normalization **then**

      $\widehat{c_i\boldsymbol{x}_i}^{\mathrm{RF}} \leftarrow$ by (16)

      $\widehat{\boldsymbol{x}}_i^{\mathrm{RF}} \leftarrow$ by (18)

   else

      $\widehat{\boldsymbol{x}}_i^{\,\mathrm{RF}} \leftarrow$ by (19)

   end if

Output: $\{\widehat{\boldsymbol{x}}_i^{\mathrm{RF}}\}_{i\in[n]}$
---

method with SDP-refinement of covariance has higher complexity but it is still practical.

4   Results for Synthetic Data

In this section, we compare our method with the original CWF method for contrast estimation and image denoising using synthetic data. To generate the synthetic data, we create the 2-D clean images by projecting a 3-D volume from uniformly distributed viewing directions. The images are downsampled to size 256×256. We use the 3-D volume of the P. falciparum 80S ribosome bound to E-tRNA, which can be freely obtained from the Electron Microscopy Data Bank (EMDB) with ID number EMD-2660 [35]. We apply 10 different CTFs to the projected clean images, whose defocus values range from 1 μm to 4 μm. For all CTFs, we choose the voltage as 300 kV, the amplitude contrast as 7%, and the spherical aberration as 2 mm. We then rescale the clean CTF-transformed images by image amplitude contrasts that are i.i.d. uniformly distributed in $[0.5, 1.5]$. At last, we add additive white or colored Gaussian noise. For the colored noise, we choose the noise power spectrum as $1/\sqrt{k^2 + 1}$ up to a constant, where k is the radial frequency (in $1/(128$ pixel size)) in the Fourier domain. The pixel size is set as $1.34 \times 360/256$ Å, where 360 is the original dimension of the volume before downsampling.

We implement all algorithms on a cluster with 750GB shared memory and 72 cores running at 2.3 GHz, where 20 cores were used. We implement CWF using the ASPIRE package [36] with its default setting. As for our methods, the SDP covariance refinement formulation is solved in CVXPY [8] by its default solver SCS [19]. Our Python code is available at https://github.com/yunpeng-shi/contrast-cryo, and is planned to be integrated into ASPIRE.

We next comment on the runtime of the algorithms. The Fourier-Bessel expansion for a batch of 1000 images takes 110 seconds. With 10 defocus groups, the covariance estimation by CWF takes 1780 seconds for white noise and 2040 seconds for colored noise. The covariance refinement by SDP takes 1.5 seconds, whereas for GS it is less than 1 second. Image denoising by Wiener filtering of 1000 images takes 84 seconds. For the same images, the runtime for computing contrasts from the Fourier-Bessel coefficients is less than one second which is negligible.

4.1   Synthetic Data with White Noise

Figure 1 shows an example of a clean image and its noisy counterparts at different SNRs.

| clean | SNR = 1 | SNR = 0.1 | SNR = 0.02 | SNR = 0.01 |
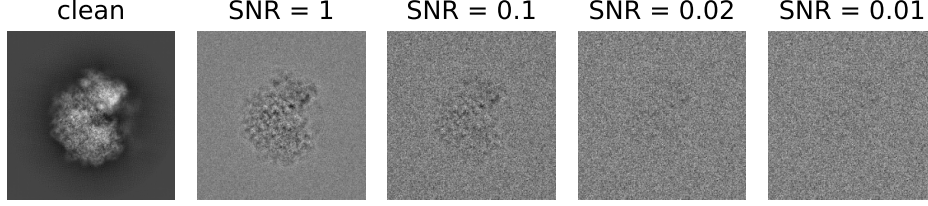
Figure 1: An example of clean and noisy images with white noise. The defocus value for the CTF of the noisy images in this example is 2.67 μm.

We next examine the performance of our estimator of contrast variance (12) under different SNRs and number of images. Since our simulated contrasts are uniformly distributed on $[0.5, 1.5]$, the ground truth variance is $1/12$, and thus ideally the line plots in Figure 2 should align with the horizontal line $y = 1$. For small number of images $n = 1000$, our method often underestimates the variability of contrast, especially under low SNR. In this regime, our method mainly captures the magnitude of image noise, which is indeed assumed as approximately a constant (so variance is small) across images. For medium size of n, namely $n = 10000$, our method gives good estimate of contrast variance at SNR= $1, 0.1$, but tends to overestimate it when SNR goes lower. In this regime, the overestimation is mainly due to the inaccurate estimation of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}_{cx}$. Ideally, in the absence of noise, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}_{cx}\mathbf{1}$ should be parallel to each other due to (11). When $\widehat{\boldsymbol{\mu}}$ and $\widehat{\boldsymbol{\Sigma}}_{cx}\mathbf{1}$ are far from being parallel, then one would expect a larger $\mathrm{Var}(c)$ to minimize the energy in (12). We finally remark that when $n = 10^5$, we are able to accurately estimate $\mathrm{Var}(c)$ for SNR as low as $1/100$.
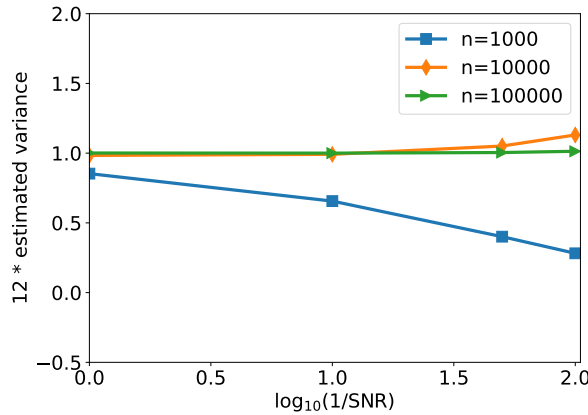
Figure 2: The estimated variance of contrasts with varying SNR and number of images n. The image noise is white Gaussian. The ground truth value of the y-axis is 1, because the image contrasts are sampled from the uniform distribution on $[0.5, 1.5]$.

We next check the estimation of the covariance matrix $\boldsymbol{\Sigma}_{cx}$. In Figure 3, we present the line plot of the normalized estimation error $e_{\boldsymbol{\Sigma}} := \|\widehat{\boldsymbol{\Sigma}}_{cx} - \boldsymbol{\Sigma}_{cx}\|_F^2 / \|\boldsymbol{\Sigma}_{cx}\|_F^2$. We observe from Figure 3 that the estimation error of the refined covariance matrix strongly depends on the estimation error of $\mathrm{Var}(c)$. Indeed, when $n = 10000$ and SNR= 1/50 and 1/100, $e_{\boldsymbol{\Sigma}}$ of CWF-GS and CWF-SDP are both significantly larger than that of CWF. This large error is mainly due to the inaccurate estimation of the contrast variance at those SNRs. Our refined covariance matrices are more accurate under low SNR and large n, such as SNR= 1/100 and $n = 10^5$ where $\mathrm{Var}(c)$ is accurately estimated. We show in Figures 3-6 that although the refinement of covariance matrices does not necessarily reduce the estimation error, it plays a critical role for accurate contrast estimation.
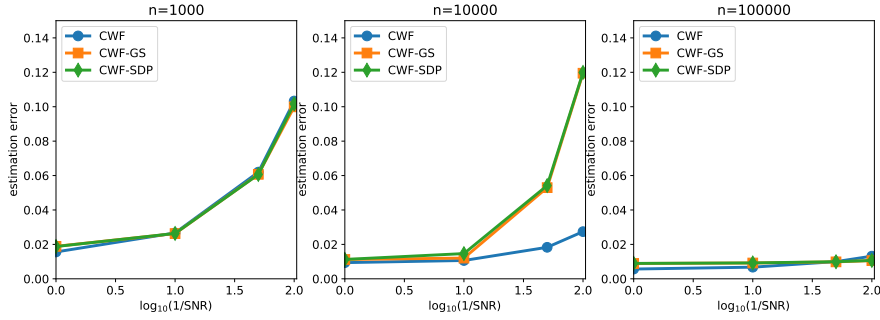
Figure 3: Normalized error of covariance estimates by different methods. The image noise is white Gaussian.

To visualize the quality of contrast estimation by different methods, we present scatter plots of estimated contrasts v.s. ground truth ones. Ideally, the points in scatter plots should align well with the line $y = x$. We first show in Figure 4 the scatter plots of different methods when $n = 10000$ and SNR $= 1$. "CWF-Oracle" refers to the CWF method with ground truth mean and covariance. We note that the oracle is the best linear estimator of the contrast. Our CWF-GS and CWF-SDP perform similarly and both of them achieve near-oracle accuracy for contrast estimation, and they are significantly better than the plain CWF.
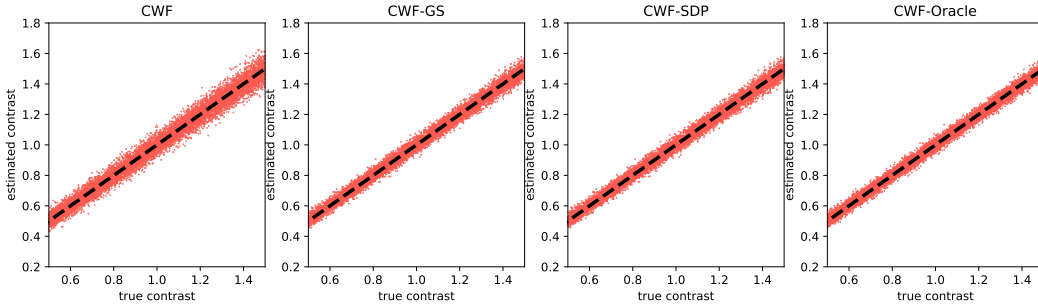
Figure 4: Scatter plots of estimated contrasts v.s. true contrasts. $n = 10000$, SNR $= 1$. The image noise is white Gaussian. Ideally each scatter plot should align well with the line $y = x$.

14

Next, in Figure 5 we keep the number of images fixed and lower the SNR to 0.1. All algorithms perform significantly worse than the results of SNR = 1. However, CWF-GS and CWF-SDP produce more accurate contrast estimates than those of plain CWF and are comparable to the oracle, which is consistent with Figure 4.
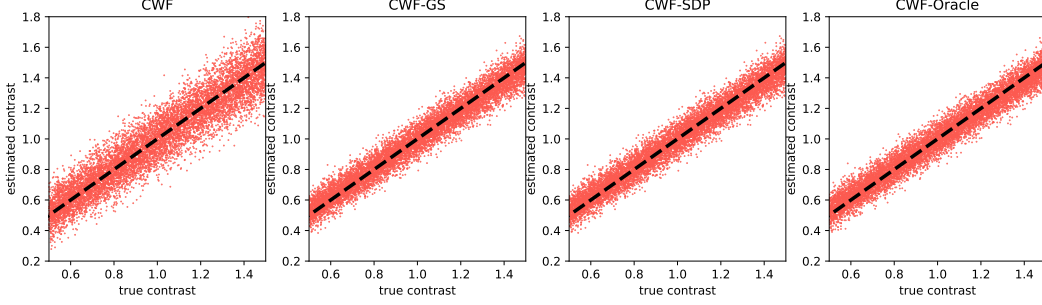
Figure 5: Scatter plots of estimated contrasts v.s. true contrasts. $n = 10000$, SNR = 0.1. The image noise is white Gaussian. Ideally each scatter plot should align well with the line $y = x$.

Next, we show that the number of images often does not significantly affect the performance of our methods. That is, unlike CWF, our method does not require a large sample size for estimating the contrasts. In Figure 6, we reduce the number of images to 1000 while keeping SNR = 0.1. The contrast estimation by the plain CWF is much less accurate after reducing the number of images. In contrast, our methods better maintain the quality of contrast estimates after reducing n. This suggests that our method is more robust to inaccuracies of the estimated covariance matrix.
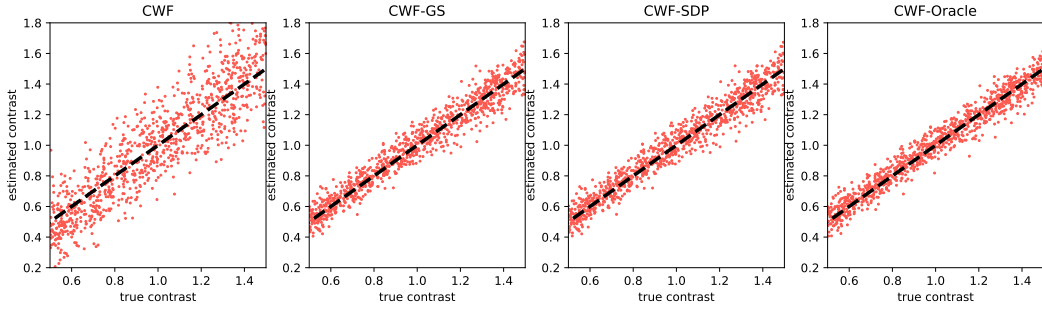
Figure 6: Scatter plots of estimated contrasts v.s. true contrasts. $n = 1000$, SNR = 0.1. The image noise is white Gaussian. Ideally each scatter plot should align well with the line $y = x$.

In Figure 7 we compare the contrast estimation error of different methods under different SNRs and number of images. We use the averaged relative error

$$e_c = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{\widehat{c}_i - c_i}{c_i} \right| \tag{24}$$

15

to measure the performance of the contrast estimation. We limit the y-axis of the line plot on the interval $[0, 0.28]$ since any contrast estimation error above 0.28 is regarded as non-informative. Indeed, a trivial contrast estimator that estimates every c_i as 1 would give the error close to 0.28 in expectation. We observe that when $n = 10000$, although the covariance matrices are not very accurately estimated, CWF-GS and CWF-SDP both achieve performance that is comparable to the oracle. However, CWF needed 100000 samples to reduce the gap to the oracle. Even with $n = 100000$, CWF is still slightly worse than the oracle and our methods. Therefore, the key factor that determines the quality of contrast estimation is not how covariance is close to the true one, but is whether the covariance is enforced to satisfy the constraints stated in Proposition 1.
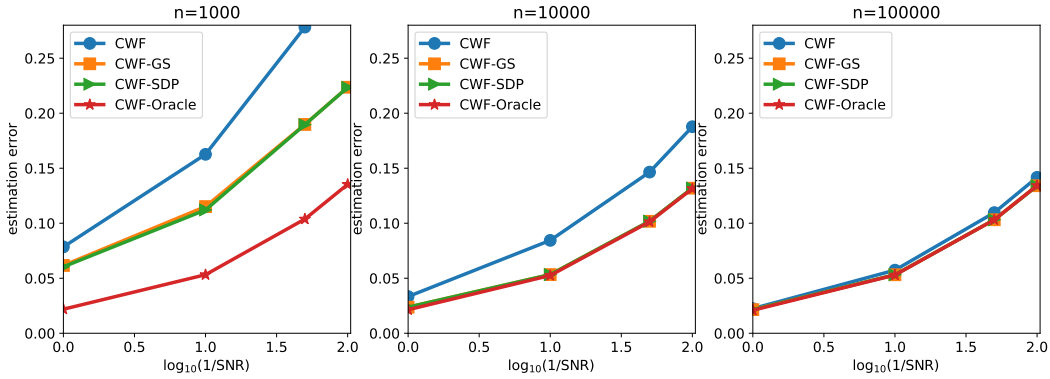
Figure 7: Contrast estimation error under different SNRs and number of images. The image noise is white Gaussian.

Next, we test the performance of the algorithms on image denoising. We compare the plain CWF and the ones with our refined covariances. We also compare the denoised images with image normalization and our 2-stage CWF procedure, introduced in Section 3.5. The two previous methods we compare are CWF and CWF-norm [4]. The latter one is the CWF with an image normalization step. The labels "-GS" and "-SDP" refer to usage of the refined covariance matrix (estimated by our GS procedure and SDP method) for CWF.

Before presenting the estimation errors, we show an example of clean and noisy images and denoised ones by different methods. In this example, SNR = 0.1 and $n = 10000$. From the result of Figure 8, the denoised image by the original CWF with normalization looks similar to the ones by our normalization methods, although they have slightly different contrasts. The denoised images by our 2-stage methods have clearer fine details than those that are denoised by other methods.
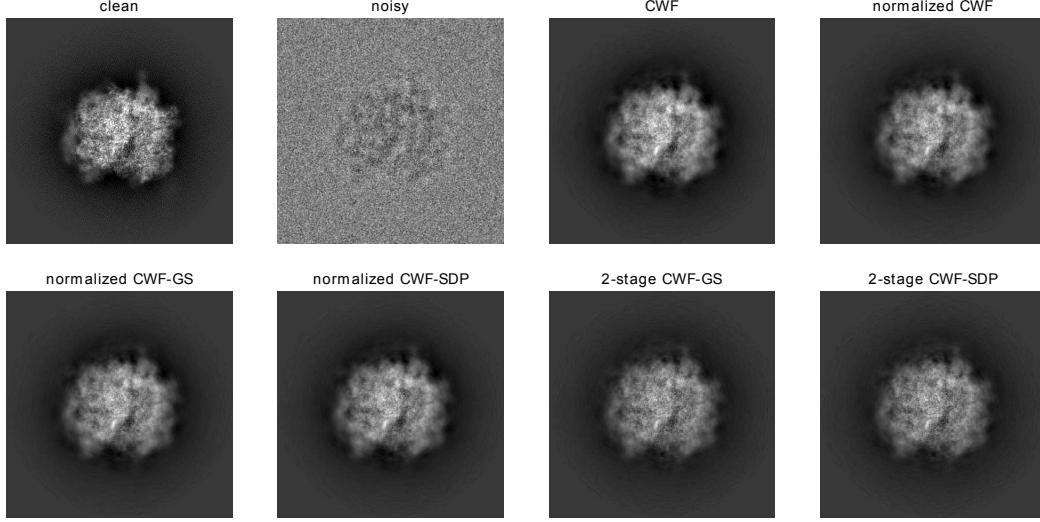
Figure 8: Clean, noisy and denoised images with SNR = 0.1 and $n = 10000$. The image noise is white Gaussian.

We evaluate the denoising performance by the normalized root mean squared error (NRMSE) within a circular mask whose radius is half the image size. From Figure 9, CWF with image normalization often gives large errors under low SNR and small to medium n. Our image normalization and 2-staged methods consistently perform better than CWF and CWF-normalization, where 2-staged methods are slightly better. We also observe that our GS and SDP refinement yield similar estimation errors, where GS is slightly better under low SNRs.
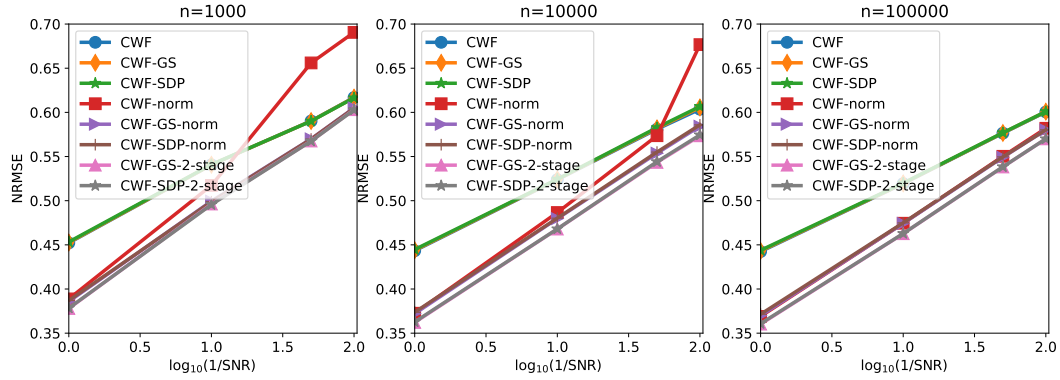
Figure 9: NRMSE of the denoised images under different SNRs and the number of images. The image noise is white Gaussian.

We further examine the contrast estimation error in each defocus group. In Figure 10 we compare the contrast estimation errors of different methods in each of the 10 defocus groups for $n = 10000$ and SNR = 0.1. The defocus groups are sorted by defocus values

in ascending order. In addition to the previously tested methods, we include a stronger oracle that knows the true clean images (not just true covariance). It estimates the contrast by $\widehat{c}_i = \langle \boldsymbol{y}_i, A_i \boldsymbol{x}_i \rangle / \| A_i \boldsymbol{x}_i \|_2^2$. We refer to this method as "Oracle". On the right panel of Figure 10 we test the contrast estimation when the observed noisy images are randomly shifted by 1-5 pixels in x and y directions. From Figure 10, the contrast estimation errors of both CWF and our methods tend to decrease when the defocus value increases. This makes sense, since CTFs with larger defocus values have higher absolute values around the zero-th frequency, and thus enjoy higher SNRs at low frequencies. When all images are centered, the "oracle" indicates the best possible contrast estimation that a template-based method can achieve, which obviously outperforms all other methods including the CWF-oracle. We remark that "oracle" knows the true manifold of the clean images, whereas "CWF-oracle" assumes a linear approximation of it. However, the new oracle is not robust to shifts, unlike other methods. When the noisy images are shifted, the oracle, assuming it does not know the shifts in the observation and only computes the dot product between the shifted $\boldsymbol{y}_i$ and centered $A_i \boldsymbol{x}_i$, gives poor contrast estimation. To mitigate this issue, a low pass filter to $\boldsymbol{y}_i$ and $A_i \boldsymbol{x}_i$ is often needed.
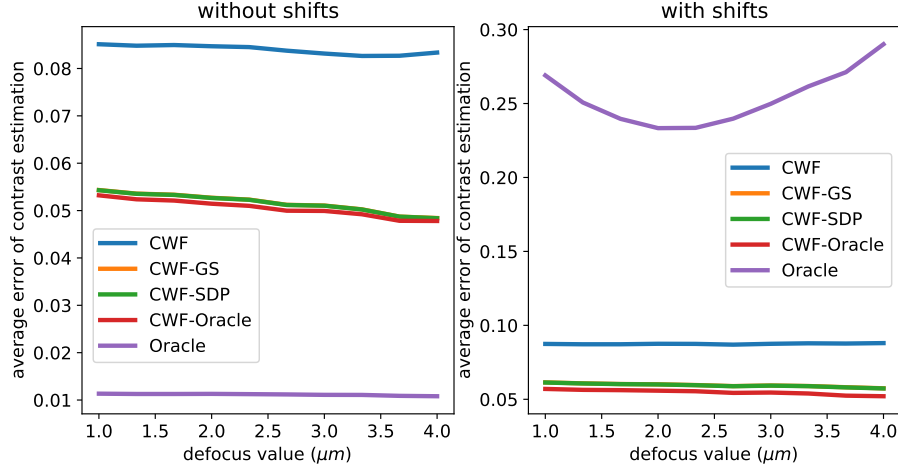
Figure 10: Average error per defocus group of contrast estimation by different methods. $n = 10000$, SNR $= 0.1$. The image noise is white Gaussian. The left figure panel uses centered noisy images. In the right panel, we randomly shifted the noisy images by 1-5 pixels in the x and y directions independently. In both panels, the two lines corresponding to CWF-GS and CWF-SDP overlap with each other.

At last, in the left panel of Figure 11, we compare the NRMSE of the denoised images by CWF with image normalization and our methods for each defocus group. On the right panel we show the relationship between the NRMSE of the denoised images and their contrast values. In particular, we divide the images into 10 groups by their true contrast values. Namely, the images with contrasts between 0.5 and 0.6 are classified as the first contrast group, and those with contrasts 0.6-0.7 are considered the second group and so on. We do not show CWF with image normalization since it has significantly higher NRMSE than other methods and will screw the scale of the y-axis. From the figure, for all methods,

18

the NSMSEs often decrease when defocus values and contrast values increase. This agrees with our argument that higher defocus and contrast correspond to higher SNRs at low frequencies. However, with higher defocus values, more energy of clean signals is spilled outside of the image disk [28], and CTFs have more zero-crossings, which may have negative effects on image denoising. Indeed, we notice that when defocus values approach 4 μm, the NRMSEs slightly increase. Overall, the 2-stage methods perform significantly better than other methods.
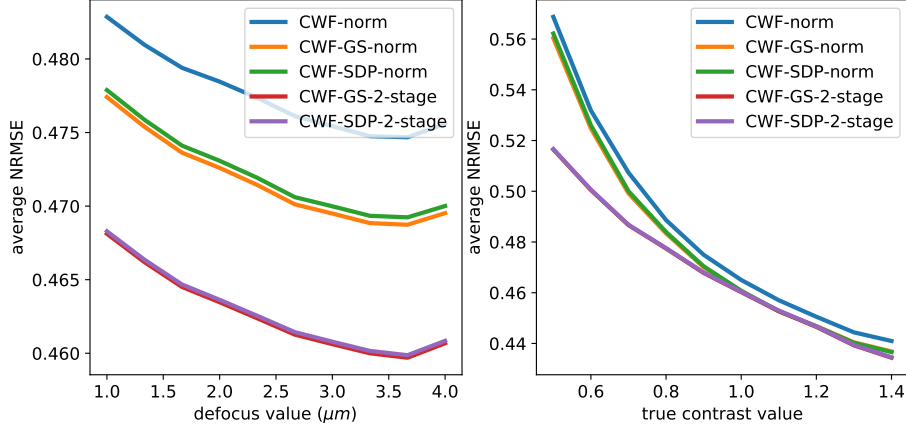
Figure 11: Average NRMSE of denoised images from different methods, per defocus group (left figure) and per contrast group (right figure). $n = 10000$, SNR $= 0.1$. The image noise is white Gaussian. In the right panel, the red and purple lines overlap with each other.

4.2 Synthetic Data with Colored Noise

We retest different methods on synthetic data with colored noise. The data generation procedure is exactly the same as before, except that now we use colored noise whose power spectrum decays with the radial frequency. Colored noise is more realistic in the sense that it better mimics the noise statistics observed in experimental images. Our choice of colored noise makes contrast estimation more challenging. Indeed, given the noise spectrum $1/\sqrt{k^2 + 1}$ (up to a constant) where k is measured in $1/(128$ pixel size), under the same SNR, the noise power spectrum in the zeroth frequency is expected to be 40 times larger than that of the white noise. Since contrast (mean of the pixels) is all about the zeroth frequency, the high noise at low frequencies poses a serious challenge.

Figure 12 shows an example of a clean image and noisy ones at different SNRs. Comparing with Figure 1, the particles are harder to identify by human eyes than in the case of white noise. Indeed, starting from SNR=0.1, it already becomes hard to visually distinguish the particle from the colored noise in the background.

19
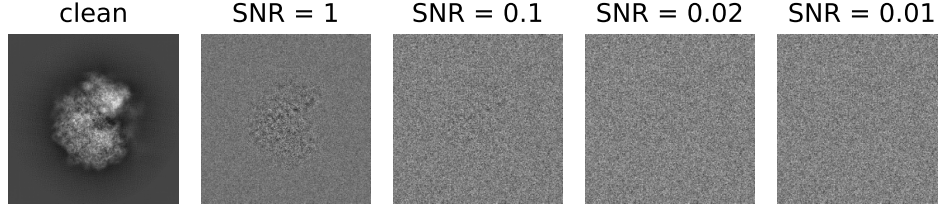
Figure 12: An example of clean and noisy images with colored noise. The defocus value for the CTF of the noisy images in this example is 2.67 μm.

We next test the variance estimation for the contrasts. As shown in Figure 13, the performance of the variance estimation is indeed worse than that for images with white noise. For $n = 1000$, our method consistently underestimates the variance. For $n = 10000$, there is an interesting transition from overestimation to underestimation between SNR= $1/50$ and $1/100$. This is likely due to that at SNR= $1/100$, our method starts to learn the variance of the average pixel values of noise which is close to 0. However, for $n = 100000$, we are able to reliably estimate Var(c) up to SNR $= 0.1$.
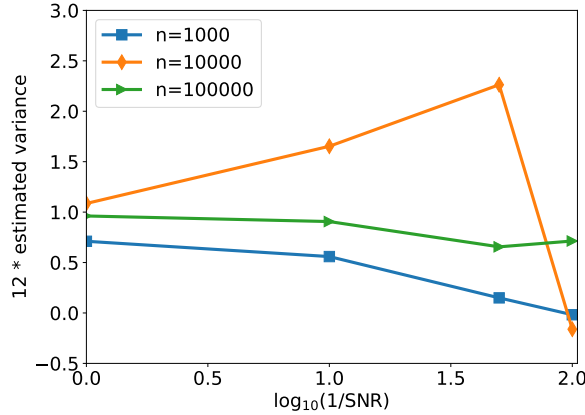
Figure 13: The estimated variance of contrasts with varying SNR and n. The image noise is colored Gaussian with decaying PSD. The ground truth of the y-axis is 1.

Figure 14 shows the covariance estimation error for the different methods. Similar to the white noise case, the large errors of our methods when $n = 10000$ are mainly due to overestimation of Var(c). We notice a slight drop of covariance estimation error at SNR= 0.01 when $n = 10000$. This is due to the reduced error of contrast variance estimation (see the orange line in Figure 13). However, as we show next, these refined covariance matrices are key for accurate contrast estimation.
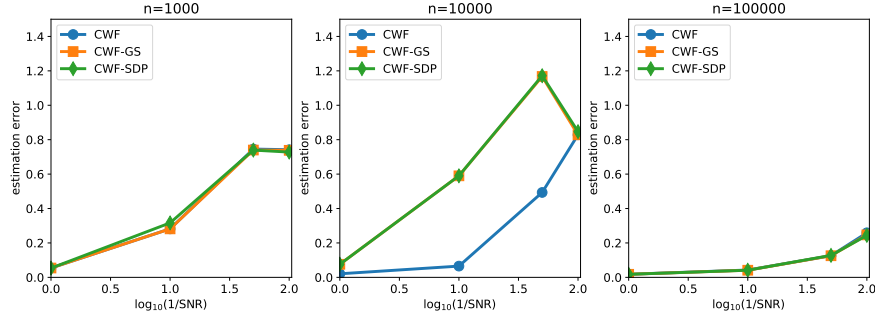
Figure 14: Normalized error of covariance estimates by different methods. The image noise is colored Gaussian with decaying PSD. The line of CWF does not appear in the left panel due to its high error. In the right panel, the line of CWF overlaps with the lines of other methods.

As before, we assess the quality of the contrast estimation through scatter plots. From the result of Figure 15, all methods perform significantly worse than in the white noise case. However our methods are still comparable to the oracle one and are considerably better than the original CWF.
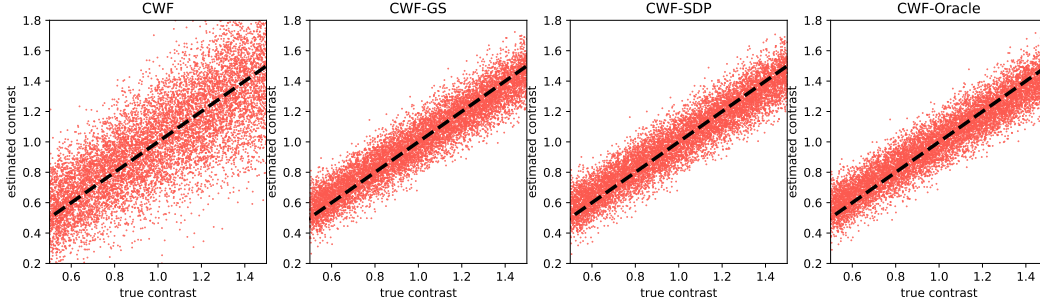
Figure 15: Scatter plots of estimated contrasts v.s. true contrasts. $n = 10000$, $\text{SNR} = 1$. The image noise is colored Gaussian with decaying PSD.

Next, we fix n and decrease SNR to 0.1. In Figure 16, the original CWF almost fails since there is no clear linear association between its estimated contrasts and the true ones. However, one can see a clear trend between the contrasts estimated by our methods and the ground truth ones. Again, our methods achieve comparable accuracy to the oracle one.
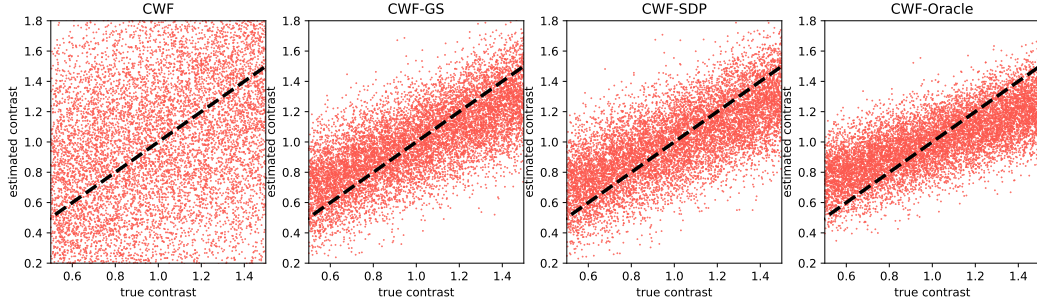
Figure 16: Scatter plots of estimated contrasts v.s. true contrasts. $n = 10000$, SNR $= 0.1$. The image noise is colored Gaussian with decaying PSD.

We next keep the SNR and reduce the number of images to 1000. In Figure 17, the performance gap between our methods and CWF is even larger, and our methods are still comparable to the oracle. In both Figure 16 and 17 the scatter plots of our methods tend to follow a straight line with a smaller slope, due to the high noise. Indeed, consider the extreme case of pure noise images, the estimated contrasts should follow a horizontal line.
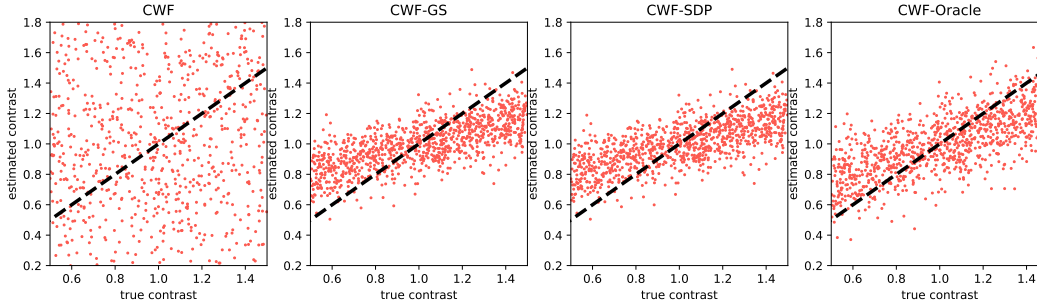
Figure 17: Scatter plots of estimated contrasts v.s. true contrasts. $n = 1000$, SNR $= 0.1$. The image noise is colored Gaussian with decaying PSD.

In Figure 18 we compare the contrast estimation error of different methods under different SNRs and number of images. We observe that the CWF-GS and CWF-SDP both perform comparably to the oracle for $n \geq 10000$. They also perform close to the oracle for the small sample size $n = 1000$, which indicates their robustness to the sample size unlike CWF. When $n = 1000$, the error of CWF is always above 0.28, thus it does not appear in the plot. There is still a large gap between CWF and our methods (and oracle) when $n = 100000$.

22
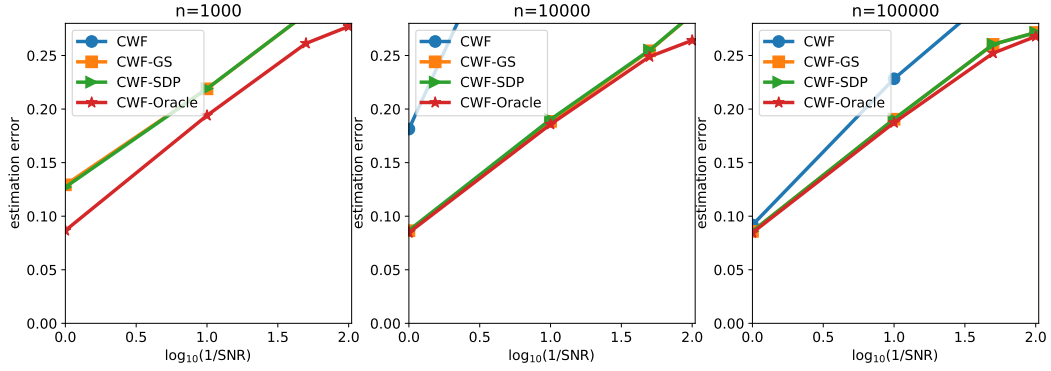
Figure 18: Contrast estimation error under different SNRs and number of images. The image noise is colored Gaussian with decaying PSD.

As for image denoising, we first show an example of clean and noisy images and denoised ones by different methods. In this example, SNR = 0.1 and $n = 10000$. From the result of Figure 19, the denoised image by the original CWF with normalization gives much lower contrast than the ones by our normalization methods. The denoised images by our 2-stage methods seem to have clearer fine details than those that are denoised by other methods.
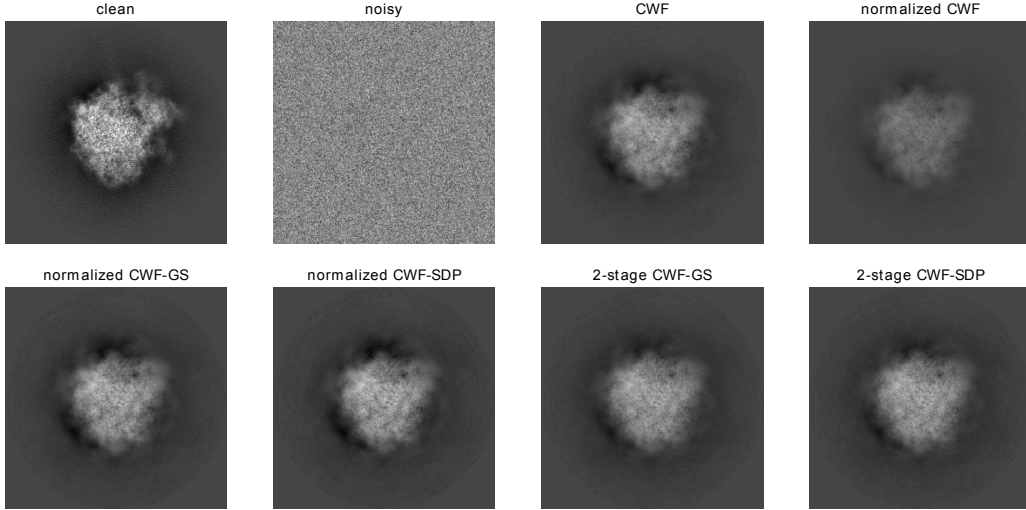
Figure 19: Clean, noisy and denoised images with SNR = 0.1 and $n = 10000$. The image noise is colored Gaussian with decaying PSD.

Next, we compare the NRMSE of the denoised images by the different algorithms. From Figure 20, CWF with image normalization is very unstable. It does not appear in the first subplot due to exceeding the y-axis limit. Similar to the white noise case, our image normalization and 2-staged methods often have smaller errors than other methods, where 2-staged methods are slightly better. Similar to the white noise case, our GS refinement

yields slightly smaller estimation errors than the SDP method under low SNRs.
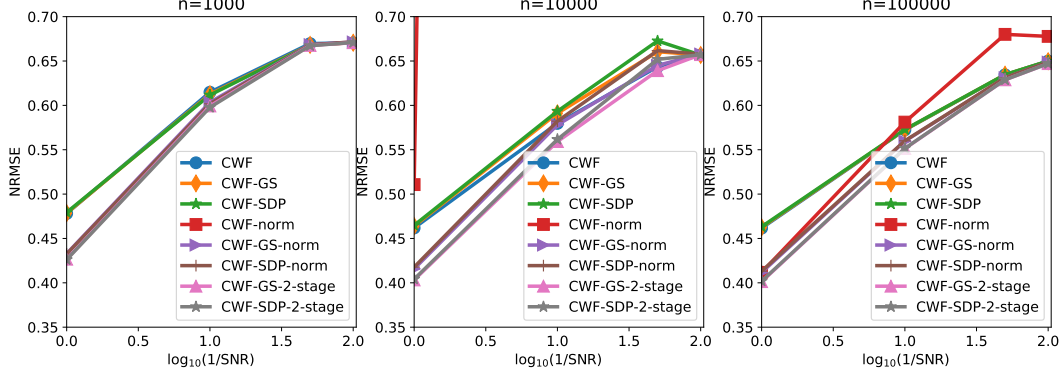
Figure 20: NRMSE of the denoised images under different SNRs and number of images. The image noise is colored Gaussian with decaying PSD.

Similar to the white noise case, we examine the relationship between contrast estimation errors and the defocus values of the corresponding CTFs. In Figure 21 we compare the average contrast estimation errors of different methods in each of the 10 defocus groups. The defocus groups are sorted by defocus values in ascending order. On the right of Figure 21 we test the contrast estimation when the observed noisy images are randomly shifted by 1-5 pixels in the x and y directions. From Figure 21, the contrast estimation errors of both CWF and our methods tend to decrease when defocus value increases. The instability of the "oracle" method to the shifts of images is also observed.
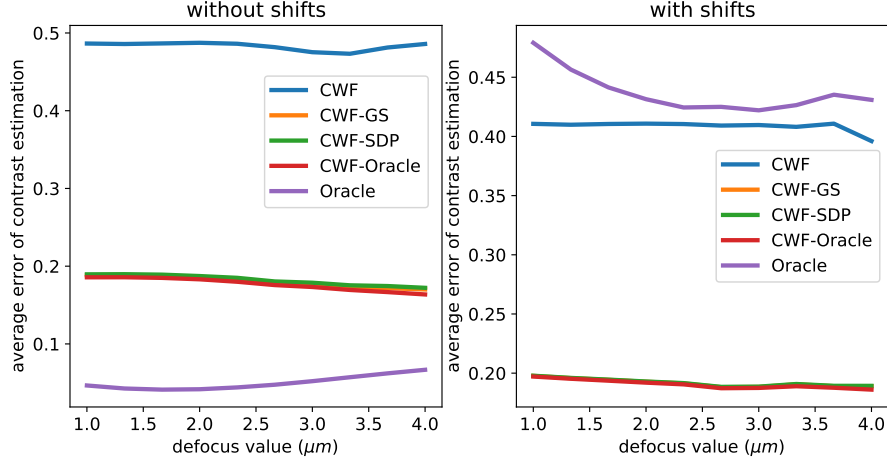
Figure 21: Average error per defocus group of contrast estimation by different methods. $n = 10000$, SNR $= 0.1$. The image noise is colored Gaussian with decaying PSD. The left panel uses centered noisy images. In the right panel, we randomly shift noisy images by 1-5 pixels in the x and y directions independently. In the right panel, the lines corresponding to CWF-GS and CWF-SDP overlap with each other.
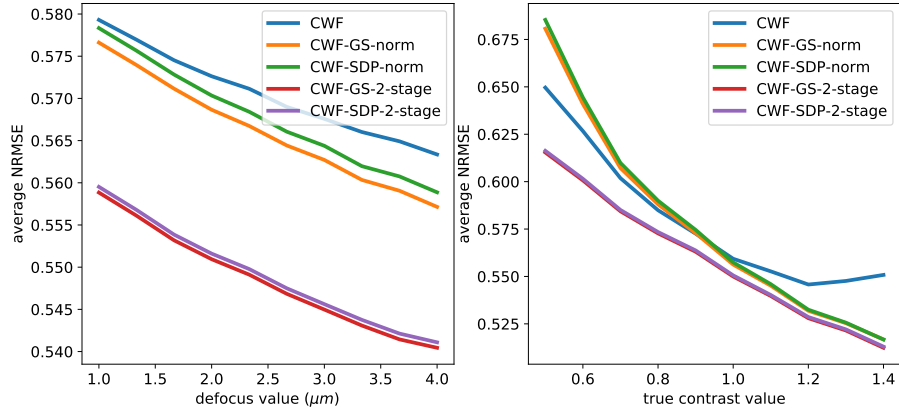
Figure 22: Average NRMSE of denoised images from different methods, per defocus group (left panel) and per contrast group (right panel). $n = 10000$, SNR $= 0.1$. The image noise is colored Gaussian with decaying PSD.

Same as the white noise case, in the left panel of Figure 22, we compare the NRMSE of the denoised images by CWF and our methods within each defocus group. The right panel shows the relationship between NRMSE of denoised images and their contrast values. From the figure, for all methods, the NSMSE often decreases when defocus value and contrast value increase. The results of this section suggest that the 2-stage methods outperform the

25

other CWF-based methods, and we therefore expect them to be the method of choice also for experimental data.

5 Results for Experimental Data

We compare our methods with CWF on three experimental datasets, which are freely downloadable from the Electron Microscope Pilot Image Archive (EMPIAR) database [14]. We chose these datasets for a purely technical reason, as each micrograph in these datasets has a single CTF, which reduces the total number of CTFs and runtime of our method. For the datasets where each image has its own CTF, it is possible accelerate our method by implementing in 2-D Fourier space the operations that involve the CTFs. However, to keep the idea of this work clean and focused, we leave this modification to future work. Due to the similar performance of our methods on the the three datasets, in this section we only present the result for EMPIAR-10028 [35], and refer the reader to the supplementary material for the results for EMPIAR-10005 [16] and EMPIAR-10073 [18].

For all datasets, we first normalize each individual image by the standard deviation (std) of the pixel values at the image corners that are located outside a circular mask with radius $0.45L$, where L is the dimension of the square image. Next, for each defocus group we estimate the PSD of the noise in the normalized images, using the pixel values outside of the same mask. We then perform background subtraction by subtracting the mean of pixel values outside the mask. For each defocus group, the images are whitened by applying the single whitening filter that equals to the -0.5 power of the estimated noise PSD of that defocus group. By doing this, we are assuming that the images in the same micrograph have similar noise PSDs, in order to reduce the estimation error of the noise PSD which could be quite large for a single image. Moreover, whitening by defocus group accelerates our algorithm. In particular, given the whitened image formation model $\boldsymbol{W}_i \boldsymbol{y}_i = c_i \boldsymbol{W}_i \boldsymbol{A}_i \boldsymbol{x}_i + \boldsymbol{W}_i \boldsymbol{\epsilon}_i$, to recover $\boldsymbol{x}_i$ we use the whitened CTFs $\boldsymbol{W}_i \boldsymbol{A}_i$. Using distinct whitening filter $\boldsymbol{W}_i$ for all images increases the number of distinct CTFs and the computational complexity of the existing implementation of CWF. We also note that we have to estimate the PSD before the background subtraction, otherwise the estimated PSD will vanish at the zeroth frequency and cause numerical issues when whitening the image. The possibly inaccurately estimated PSD, together with imperfect centering of particles and the ignored astigmatism in CTF, may cause imperfect CTF correction and additional blurring in the restored images. However, we demonstrate in our experimental results that our methods are more robust to these factors than the original CWF, especially for contrast estimation. The machine and the number of cores we used for the experimental datasets are the same as those of the synthetic simulations.

5.1 EMPIAR-10028

We test the algorithms on a dataset of the Plasmodium falciparum 80S ribosome bound to the anti-protozoan drug emetine. The picked particles are downloadable from EMPIAR-10028 [35]. Its 3-D reconstruction can be found on EMDB as EMD-2660 [35]. The dataset consists of 105247 motion corrected and picked particle images of size 360×360 with 1.34 Å pixel size, from 1081 defocus groups. We estimate the covariance using all images, and use 21 defocus groups to estimate the contrast of individual images and then denoise the selected images. The background subtraction, whitening, Fourier-Bessel expansion and covariance

estimation took 10 hours. It took 5 seconds for SDP covariance refinement and less than 1 second for the GS one. We apply Wiener filtering to 21 defocus groups, which take 11 minutes. Contrast estimation from the Fourier-Bessel coefficients took less than one second.

We first examine the relationship between the contrasts of particle images and their locations in a micrograph. Since the ground truth clean contrasts are not available, we use the approximate ground truth contrasts that are obtained from template matching with clean projections of the 3-D volume estimated by RELION (available in EMD-2660). In order to do this, we first generate 1000 clean templates that are projected from uniformly distributed viewing directions. Next, for each particle image, we find its viewing direction, 2-D in-plane rotation and shift by aligning its CWF-denoised image with each of the clean template by the method of [22]. We found that using the denoised images often provide more accurate alignment than using the raw images. To compute the oracle contrast of each noisy image Y_i, we apply its CTF to the aligned clean template and obtain Y_i^*. However, the contrast directly computed by

$$c_i^* = \langle Y_i , Y_i^* \rangle / \|Y_i^*\|_F^2$$

can be sensitive to even slight errors in alignment, as we demonstrated in Figures 10 and 21 in synthetic data simulations. To mitigate this issue, we apply Gaussian smoothing to both Y_i and Y_i^* before computing the "ground truth" contrast using the above formula. We choose an envelope function with a B-factor 1000 as our Gaussian filter. We remark that the clean projections are only used to generate approximate ground truth for evaluation, and are not used in CWF and our methods.

Each subplot of Figure 23 corresponds to one micrograph, where each dot represents a picked particle image in that micrograph. The location of the dots are the location of the particle images in that micrograph, whose color represents the oracle contrast by template matching. The defocus values of the three micrographs (from left to right) are respectively 0.8131 μm, 1.9676 μm, and 2.6643 μm. Figure 23 suggests the existence of local correlations of image contrasts. Indeed, many pairs of nearby particles have very similar contrasts. However, the correlation of contrasts is only present within very small sub-regions of the micrograph.
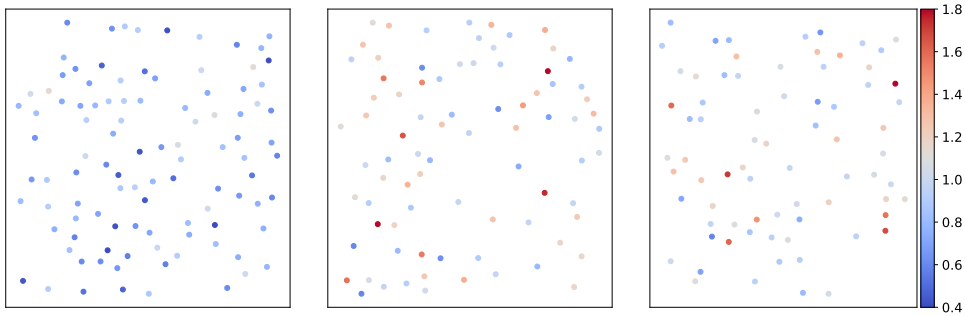

Figure 23: Demonstration of the relationship between the contrast of picked particle and their locations in the micrographs of EMPIAR-10028. Each dot corresponds to a particle image, whose color represents its estimated contrast.

We next present a box plot of both oracle contrasts (top subplot) and the contrasts estimated by CWF-GS (bottom subplot) for each of the 21 defocus groups in Figure 24. We

ignore the result from CWF-SDP as it is very similar to the one of CWF-GS. We also ignore the result from CWF, since its contrast estimation is not accurate (see later in Figure 25) and thus its box plot is not informative. From left to right in each subfigure, the defocus values are sorted in ascending order, ranging from 0.8131 μm to 2.6643 μm. In each box plot, the 5 horizontal lines, from top to bottom, respectively correspond to max value, 75% quantile, median, 25% quantile and min value. The two box plots are similar, even though their contrasts are estimated using completely different methods. One can also see a clearer trend from the second subfigure (our method) that micrographs with higher defocus values tend to have higher contrast. This makes sense, as CTFs with higher defocus values preserve more low frequency information, which yields higher SNR in low frequencies. Interestingly, both subfigures show that contrast variation within each micrograph is often larger than the variance of the median contrast of each micrograph (the variance of the y-values of the orange lines). This possibly indicates that using a single contrast value per micrograph, as assumed in the 3-D iterative refinement stage, is not appropriate.

Next, we present the scatter plot between the estimated contrasts and the oracle contrast. It is clear from Figure 25 that our estimates have much better correlation with the oracle. We remark that we do not expect a strong correlation in any case, since the oracle itself is noisy and suffers from imperfect alignment. However, this is strong evidence that our methods provide much better contrast estimates than CWF.
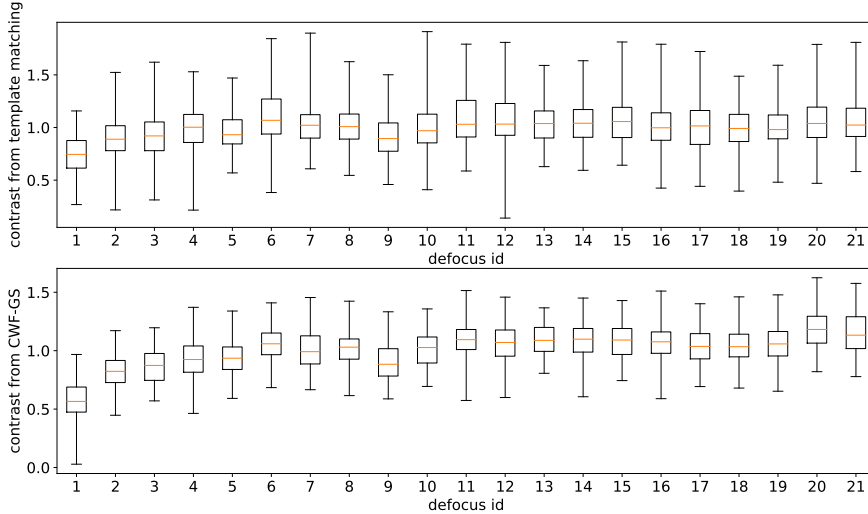

Figure 24: Box plot of the oracle contrasts (top) and our estimated contrasts (bottom) in 21 defocus groups of the dataset EMPIAR-10028. The defocus values are sorted in ascending order, ranging from 0.8131 μm to 2.6643 μm.

We next compare the image denoising performance by CWF and the ones with our refined covariance matrix. Since image normalization only affects the global scale of the image and normalized CWF performs poorly, we only show the denoised images without normalization. We also found that the 2-stage CWF often performs worse than the 1-stage version, possibly due to violation of assumptions in our synthetic model, such as
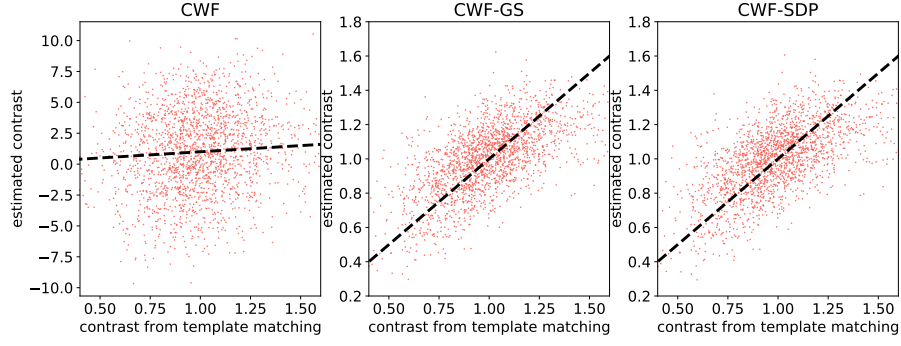
Figure 25: The scatter plots of the estimated contrast v.s. the oracle contrast for three defocus groups in the dataset EMPIAR-10028. The dashed line corresponds to the function $y = x$.

imperfect centering and astigmatism in CTF. Thus, we recommend applying the one-stage algorithm for experimental datasets, and we compare their denoised images as follows. From Figure 26, all methods produce dark areas around the boundary of the particle. These dark rings are likely due to the imperfect CTF correction by CWF. The denoised images by our methods have less dark areas, comparing to that of CWF. Since negative pixel values are often observed in CTF-affected clean images, these dark rings in CWF-denoised images are possibly due to inaccurate CTF-correction, which suggests better CTF correction by our methods. Furthermore, we observe better denoised images by our methods with closer contrast to the clean templates.
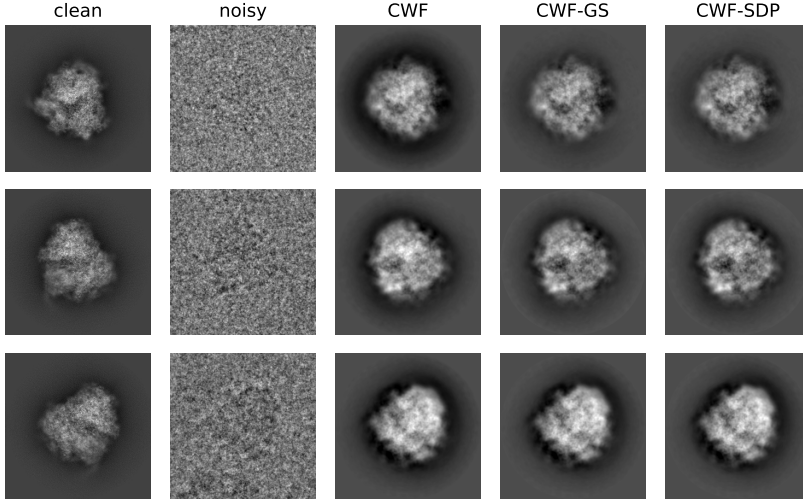

Figure 26: Denoising results of EMPIAR-10028.

At last, to quantitatively compare the denoising results of different methods, we compute

the Fourier ring correlations (FRC) between the denoised images and their aligned clean templates. That is, for each pair of two images I_1 and I_2 and their Fourier coefficient vectors $\boldsymbol{f}_{1,r}, \boldsymbol{f}_{2,r}$ at radial frequency r, we compute

$$\text{FRC}(r) = \frac{\Re(\boldsymbol{f}_{1,r}^* \boldsymbol{f}_{2,r})}{\|\boldsymbol{f}_{1,r}\|\|\boldsymbol{f}_{2,r}\|},$$

where $\Re$ denotes taking the real part of a complex number. For each method, we compute the average FRC between the denoised images and the clean templates over the 2015 images from 21 defocus groups. We notice that FRC is very sensitive to image rotations and shifts. With slight error in image alignment, the FRC of all methods decreases rapidly as r increases. As a result, when alignment errors are present, FRC may not reflect the true image quality. However, even from the first few frequencies, the FRCs of CWF-based methods are much higher than that of the naïve phase flipping method. We also notice that CWF-denoised images have large errors in the first two frequencies, mainly due to its limitations in handling contrast variations. Since the clean templates are only aligned and registered with CWF-denoised images, the comparison is a bit unfair to our methods, as our methods may suffer from larger alignment errors. However, even in this scenario, our methods achieve much better FRC at the first two radial frequencies due to the better contrast estimation.
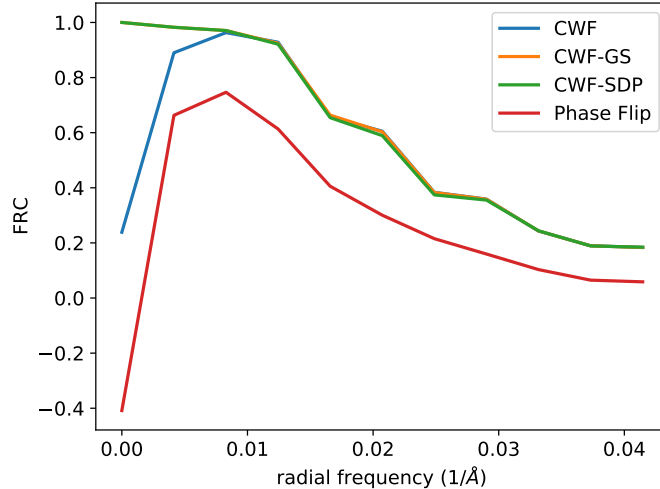

Figure 27: The average Fourier ring correlation between denoised images and the aligned clean templates over 2015 images from EMPIAR-10028.

6 Conclusion

We introduced an effective algorithm for estimating the amplitude contrast of individual images and the overall contrast variability in the ab-initio stage. Our method refines the initial estimated covariance so it satisfies additional constraints that follow from the image formation model by tomographic projection. Results for both synthetic and experimental

datasets indicate consistently better contrast estimation by our methods than CWF. On synthetic data, the contrast estimation errors of our methods are comparable to those of an oracle, even with small number of images. We also demonstrate that our method improves the image denoising result of CWF. Among the various contrast estimation and image denoising techniques that were considered in this paper, following the results for experimental datasets we recommend using CWF-GS (see Algorithm 1 with option=GS in Section 3.7) for contrast estimation and CWF-GS with image normalization for image denoising (see Algorithm 2 with option=normalization in Section 3.7). There are also some interesting future directions. For example, one can try techniques based on common-lines to directly estimate the rotations of molecules by using the denoised and normalized images from our methods with rudimentary 2-D class averaging [2]. Normalizing the images may also lead to improvement of 2-D class averaging procedures. Another interesting application of our method is to use our estimated contrasts to initialize their values in the iterative refinement procedure of RELION [24]. As for the computational aspect, one can modify the original CWF method so it can more efficiently handle per-image CTF, rather than a small number of defocus groups. Our Python code is available at `https://github.com/yunpeng-shi/contrast-cryo` which is planned to be integrated into ASPIRE [36].

Acknowledgement

References

[1] X.-C. Bai, G. McMullan, and S. H. Scheres. How cryo-EM is revolutionizing structural biology. *Trends in biochemical sciences*, 40(1):49–57, 2015.

[2] A. S. Bandeira, Y. Chen, R. R. Lederman, and A. Singer. Non-unique games over compact groups and orientation estimation in cryo-EM. *Inverse Problems*, 36(6):064002, 2020.

[3] T. Bepler, K. Kelley, A. J. Noble, and B. Berger. Topaz-denoise: general deep denoising models for cryoEM and cryoET. *Nature communications*, 11(1):1–12, 2020.

[4] T. Bhamre, T. Zhang, and A. Singer. Denoising and covariance estimation of single particle cryo-EM images. *Journal of structural biology*, 195(1):72–81, 2016.

[5] T. Bhamre, Z. Zhao, and A. Singer. Mahalanobis distance for class averaging of cryo-EM images. In *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, pages 654–658. IEEE, 2017.

[6] Y. Cheng. Single-particle cryo-EM—how did it get here and where will it go. *Science*, 361(6405):876–880, 2018.

[7] S.-C. Chung, H.-H. Lin, P.-Y. Niu, S.-H. Huang, I. Tu, W.-H. Chang, et al. Pre-pro is a fast pre-processor for single-particle cryo-EM by enhancing 2-d classification. *Communications biology*, 3(1):1–12, 2020.

[8] S. Diamond and S. Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

[9] Y. Fan and Z. Zhao. Cryo-electron microscopy image denoising using multi-frequency vector diffusion maps. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 3463–3467, 2021.

[10] J. Frank. Advances in the field of single-particle cryo-electron microscopy over the last decade. *Nature protocols*, 12(2):209–212, 2017.

[11] H. Gu, I. C. Unarta, X. Huang, and Y. Yao. Generative adversarial networks for robust cryo-EM image denoising. *arXiv preprint arXiv:2008.07307*, 2022.

[12] A. Heimowitz, J. Andén, and A. Singer. Reducing bias and variance for CTF estimation in single particle cryo-EM. *Ultramicroscopy*, 212:112950, 2020.

[13] C. Helmberg, F. Rendl, R. J. Vanderbei, and H. Wolkowicz. An interior-point method for semidefinite programming. *SIAM Journal on optimization*, 6(2):342–361, 1996.

[14] A. Iudin, P. K. Korir, J. Salavert-Torres, G. J. Kleywegt, and A. Patwardhan. EMPIAR: a public archive for raw electron microscopy image data. *Nature methods*, 13(5):387–388, 2016.

[15] B. Landa and Y. Shkolnisky. The steerable graph Laplacian and its application to filtering image datasets. *SIAM Journal on Imaging Sciences*, 11(4):2254–2304, 2018.

[16] M. Liao, E. Cao, D. Julius, and Y. Cheng. Structure of the TRPV1 ion channel determined by electron cryo-microscopy. *Nature*, 504(7478):107–112, 2013.

[17] A. Moscovich, A. Halevi, J. Andén, and A. Singer. Cryo-EM reconstruction of continuous heterogeneity by Laplacian spectral volumes. *Inverse Problems*, 36(2):024003, 2020.

[18] T. H. D. Nguyen, W. P. Galej, X.-c. Bai, C. Oubridge, A. J. Newman, S. H. Scheres, and K. Nagai. Cryo-EM structure of the yeast U4/U6.U5 tri-snRNP at 3.7 Å resolution. *Nature*, 530(7590):298–302, 2016.

[19] B. O'donoghue, E. Chu, N. Parikh, and S. Boyd. Conic optimization via operator splitting and homogeneous self-dual embedding. *Journal of Optimization Theory and Applications*, 169(3):1042–1068, 2016.

[20] E. Palovcak, D. Asarnow, M. G. Campbell, Z. Yu, and Y. Cheng. Enhancing the signal-to-noise ratio and generating contrast for cryo-EM images with convolutional neural networks. *IUCrJ*, 7(6), 2020.

[21] P. A. Penczek, J. Fang, X. Li, Y. Cheng, J. Loerke, and C. M. Spahn. CTER—rapid estimation of CTF parameters with error assessment. *Ultramicroscopy*, 140:9–19, 2014.

[22] B. S. Reddy and B. N. Chatterji. An FFT-based technique for translation, rotation, and scale-invariant image registration. *IEEE transactions on image processing*, 5(8):1266–1271, 1996.

[23] A. Rohou and N. Grigorieff. CTFFIND4: Fast and accurate defocus estimation from electron micrographs. *Journal of structural biology*, 192(2):216–221, 2015.

[24] S. H. Scheres. RELION: implementation of a Bayesian approach to cryo-EM structure determination. *Journal of structural biology*, 180(3):519–530, 2012.

[25] S. H. Scheres, M. Valle, P. Grob, E. Nogales, and J.-M. Carazo. Maximum likelihood refinement of electron microscopy data with normalization errors. *Journal of structural biology*, 166(2):234–240, 2009.

[26] F. J. Sigworth. Principles of cryo-EM single-particle image processing. *Microscopy*, 65(1):57–67, 2016.

[27] F. J. Sigworth, P. C. Doerschuk, J.-M. Carazo, and S. H. Scheres. An introduction to maximum-likelihood methods in cryo-EM. *Methods in enzymology*, 482:263–294, 2010.

[28] C. V. Sindelar and N. Grigorieff. An adaptation of the Wiener filter suitable for analyzing images of isolated single particles. *Journal of Structural Biology*, 176(1):60–74, 2011.

[29] A. Singer. Mathematics for cryo-electron microscopy. In *Proceedings of the International Congress of Mathematicians: Rio de Janeiro 2018*, pages 3995–4014. World Scientific, 2018.

[30] A. Singer and F. J. Sigworth. Computational methods for single-particle electron cryomicroscopy. *Annual Review of Biomedical Data Science*, 3:163–190, 2020.

[31] C. Sorzano, L. De La Fraga, R. Clackdoyle, and J. Carazo. Normalizing projection images: a study of image normalizing procedures for single particle three-dimensional electron microscopy. *Ultramicroscopy*, 101(2-4):129–138, 2004.

[32] M. S. Spilman, H. Guo, B. E. Bammes, L. Jin, and R. B. Bilhorn. Boosting contrast of cryo-EM images without a phase plate. *Microscopy and Microanalysis*, 21(S3):911–912, 2015.

[33] M. Su, H. Zhang, K. Schawinski, C. Zhang, and M. A. Cianfrocco. Generative adversarial networks as a tool to recover structural information from cryo-electron microscopy data. *BioRxiv*, page 256792, 2018.

[34] M. Vulović, R. B. Ravelli, L. J. van Vliet, A. J. Koster, I. Lazić, U. Lücken, H. Rullgård, O. Öktem, and B. Rieger. Image formation modeling in cryo-electron microscopy. *Journal of structural biology*, 183(1):19–32, 2013.

[35] W. Wong, X.-c. Bai, A. Brown, I. S. Fernandez, E. Hanssen, M. Condron, Y. H. Tan, J. Baum, and S. H. Scheres. Cryo-EM structure of the plasmodium falciparum 80s ribosome bound to the anti-protozoan drug emetine. *Elife*, 3:e03080, 2014.

[36] G. Wright, J. Andén, V. Bansal, J. Xia, C. Langfield, J. Carmichael, R. Brook, Y. Shi, A. Heimowitz, G. Pragier, I. Sason, A. Moscovich, Y. Shkolnisky, and A. Singer. ComputationalCryoEM/ASPIRE-Python: v0.9.2. Zenodo. 2022.

[37] H. Wu, X. Zhai, D. Lei, J. Liu, Y. Yu, R. Bie, and G. Ren. An algorithm for enhancing the image contrast of electron tomography. *Scientific reports*, 8(1):1–13, 2018.

[38] K. Zhang. Gctf: Real-time CTF determination and correction. *Journal of structural biology*, 193(1):1–12, 2016.

[39] Z. Zhao, Y. Shkolnisky, and A. Singer. Fast steerable principal component analysis. *IEEE transactions on computational imaging*, 2(1):1–12, 2016.

[40] Z. Zhao and A. Singer. Rotationally invariant image representation for viewing direction classification in cryo-EM. *Journal of structural biology*, 186(1):153–166, 2014.