Joint Angular Refinement and Reconstruction for Single-Particle Cryo-EM

Mona Zehni, Laurène Donati, Emmanuel Soubies, Zhizhen Zhao, and Michael Unser

***Abstract*— Single-particle cryo-electron microscopy (cryo-EM) reconstructs the three-dimensional (3D) structure of biomolecules from a large set of 2D projection images with random and unknown orientations. A crucial step in the single-particle cryo-EM pipeline is 3D refinement, which resolves a high-resolution 3D structure from an initial approximate volume by refining the estimation of the orientation of each projection. In this work, we propose a new approach that refines the projection angles on the continuum. We formulate the optimization problem over the density map and the orientations jointly. The density map is updated using the efficient alternating-direction method of multipliers, while the orientations are updated through a semi-coordinate-wise gradient descent for which we provide an explicit derivation of the gradient. Our method eliminates the requirement for a fine discretization of the orientation space and does away with the classical but computationally expensive template-matching step. Numerical results demonstrate the feasibility and performance of our approach compared to several baselines.**

***Index Terms*— Single-particle cryo-EM, joint reconstruction, continuous angular refinement, ADMM, gradient descent.**

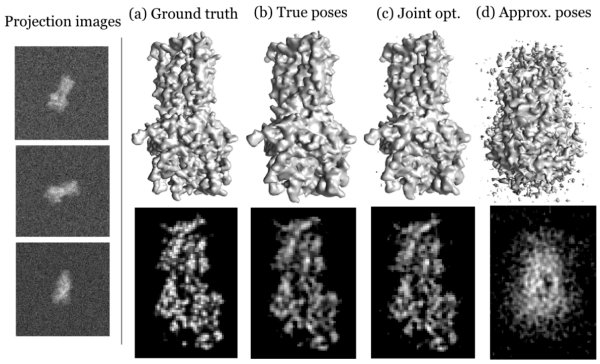

Fig. 1. Single particle analysis. Left: Samples of the noisy projection images used in this experiment. Right: Comparison between the central slices of (a) the ground-truth, (b) the reconstruction from the true orientations, (c) the reconstruction obtained with our joint refinement method, (d) the reconstruction from the unrefined initial orientations.

I. Introduction

SINGLE-PARTICLE cryo-electron microscopy (cryo-EM) aims at obtaining the three-dimensional (3D) atomic structures of biological macromolecules such as proteins or viruses. Replicates of a molecule of interest, in unknown orientations, are first imaged at cryogenic temperatures. From those 2D projections (Figure 1 left), one then reconstructs the 3D density map of the molecule (Figure 1 right), a computational process named "single-particle analysis" (SPA). The reconstruction task in SPA is extremely challenging due to the lack of knowledge on the projection directions, heavy noise and the blurring inherent with the point spread function (PSF) of the microscope. To tackle this difficulty, most methods start by estimating an *ab-initio* model from class-averaged particle images. Then, this initial model is refined iteratively until a high-resolution map is obtained, a task named "3D refinement".

A. Standard 3D Refinement Techniques

Currently, state-of-the-art refinement techniques [1]–[4] produce a high-resolution density map by alternating between

1) the reconstruction of the 3D density map for a given set of (however inaccurate) projection orientations;
2) the refinement of the projection orientations for all 2D particles based on the previously reconstructed 3D volume.

The reconstruction problem can be solved using different approaches such as algebraic methods [5], [6], weighted back-projection (WBP) [7], direct Fourier methods [8]–[10], and iterative regularized approaches [11]–[13].

In most SPA packages, direct Fourier methods based on the central-slice theorem are used. Those methods work adequately when the projections are sufficiently numerous and their speed is a key advantage. Unfortunately, their use is less appropriate in the presence of heavy noise, few projection measurements, or inaccurately known projection angles.

The past years have seen the appearance of more robust iterative schemes that formulate the 3D reconstruction problem

Manuscript received August 14, 2019; revised February 23, 2020; accepted March 18, 2020. Date of publication April 1, 2020; date of current version May 5, 2020. This work was supported by the GlobalBioIm under Grant ERC-692726, in part by the UIUC College of Engineering Strategic Research Initiatives, in part by UIUC under Grant NSF DMS-1854791. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Emanuele Salerno. *(Corresponding author: Mona Zehni.)*

Mona Zehni was with the Biomedical Imaging Group, École polytechnique fédérale de Lausanne (EPFL), 1015 Lausanne, Switzerland. She is now with the Department of Electrical and Computer Engineering, University of Illinois at Urbana–Champaign, Urbana, IL 61801 USA, and also with the Coordinate Science Laboratory, University of Illinois at Urbana–Champaign, Urbana, IL 61801 USA (e-mail: mzehni2@illinois.edu).

Laurène Donati and Michael Unser are with the Biomedical Imaging Group, École polytechnique fédérale de Lausanne (EPFL), 1015 Lausanne, Switzerland.

Emmanuel Soubies was with the Biomedical Imaging Group, École polytechnique fédérale de Lausanne (EPFL), 1015 Lausanne, Switzerland. He is now with the IRIT, Université de Toulouse, CNRS, 31000 Toulouse, France.

Zhizhen Zhao is with the Department of Electrical and Computer Engineering, University of Illinois at Urbana–Champaign, Urbana, IL 61801 USA, and also with the Coordinate Science Laboratory, University of Illinois at Urbana–Champaign, Urbana, IL 61801 USA.

Digital Object Identifier 10.1109/TIP.2020.2984313

1057-7149 © 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: EPFL LAUSANNE. Downloaded on May 08,2020 at 08:27:07 UTC from IEEE Xplore. Restrictions apply.

as a regularized optimization problem and enable the incorporation of prior knowledge on the underlying signal [14]. Their downside is that iterative schemes usually come with a prohibitive computational cost if not carefully engineered.

Several works have considerably improved this situation by proposing methods with increased speed. In particular, an important breakthrough came when a costly step of many reconstruction algorithms was shown to be quickly computable as a discrete convolution [13], [15], [16].

For the angular-refinement task, the most commonly used method is projection-matching [17], [18]. It compares every projection image against a finite set of clean templates obtained from the current estimate of the 3D density map and then assigns the angular parameters based on the closest match [19]. Projection-matching hence performs angular assignment on a discretized orientation space. As a consequence, the quality of the angular refinement depends on the fineness of the discretization and the quality of the density map used to generate the templates. One bottleneck is that a fine discretization comes at the cost of large set of templates, which leads to a computationally expensive procedure. Moreover, methods based on projection-matching were found to degrade significantly in low SNR regimes [20] or when errors occur in the estimation of the density map used for the generation of the clean templates [21].

Examples of joint-reconstruction methods that address 3D *ab initio* modeling are found in [22], [23]. In [22], a frequency-marching approach that increases the resolution of the reconstruction is proposed. This leads to a smaller computational overhead in projection-matching steps. In [23], the density map parameters are updated through gradient descent while the projection orientations are recovered through exhaustive search on an $SO(3)$ grid, followed by convex optimization. Although they provide efficient 3D *ab initio* modeling, these methods still suffer from the shortcomings of projection-matching.

Finally, a joint-reconstruction framework for 2D tomography with unknown projection orientations is proposed in [24]. The problem is solved through simulated annealing, which strongly limits its applicability to 3D tomography due to its high computational cost.

B. Maximum-Likelihood Methods

Scheres followed a Bayesian approach in [1] to formulate the 3D refinement problem as a maximum marginalized *a posterior* (MAP) estimation [25] that is solved by expectation maximization [26]. This method is less sensitive to the initial model and brings higher robustness in low SNR regimes. However, its high computational complexity limits its applicability.

Punjani *et al.* proposed a computationally efficient framework in [21]. They formulated the 3D refinement problem as a MAP estimation and solved it by stochastic average gradient descent. They also used importance sampling to further reduce the cost of computing the marginalized likelihood.

An advantage of maximum-likelihood-based methods is that they do not limit a particle image to a unique angular class. This leads to increased robustness in high-noise regimes compared to projection-matching procedures. However, they still involve some form of discretization of the projection orientations. In particular, they necessitate an overly fine discretization of the 3D orientation space, as well as a compactly supported grid over $\mathbb{R}^2$ for in-plane translations. Moreover, the marginalization process is usually computationally expensive.

C. Contributions

In this work, we present a refinement method for single-particle cryo-EM that jointly recovers the 3D density map and the orientation of each projection. This joint optimization problem is solved by letting the alternating-direction method of multipliers (ADMM) and gradient-descent steps take turns to update the density map and the orientations, respectively.

We use an explicit derivation of the gradient of the objective function (Theorem 1) to optimize the orientations over a continuous space. Hence, a key advantage of the proposed approach over usual methods is that it avoids one to resort to a fine discretization of $SO(3)$ for the orientations and $\mathbb{R}^2$ for the in-plane translations. Moreover, the computationally expensive step of projection-matching is skipped.

By using fast algorithms, we are able to efficiently refine 3D density maps from sets of projections with poor initial angular estimation. We illustrate in Figure 1 the type of refinements obtained with our joint-optimization framework, compared to a few baselines.

The paper is organized as follows: We describe in Section II the image-formation model. In Section III, we detail our joint-optimization framework. The experimental setup is described in Section IV and results are presented in Section V. Finally, we conclude this work in Section VI.

D. Notations

Sequences from $\mathbb{Z}^d \to \mathbb{R}$ are denoted by $\mathrm{c}[\cdot]$. Then, sequence samples are $\mathrm{c}[\mathbf{k}]$ with $\mathbf{k} = (k_1, \ldots, k_d) \in \mathbb{Z}^d$. Bold lowercase letters (*e.g.*, $\mathbf{c}$) represent vectors while bold uppercase letters are reserved for matrices (*e.g.*, $\mathbf{H}$). All vectors are assumed to be column vectors unless otherwise stated. The ℓ_1 and ℓ_2 norms of the vector $\mathbf{c} \in \mathbb{R}^N$ are defined as $\|\mathbf{c}\|_1 := \sum_{n=1}^N |\mathrm{c}_n|$ and $\|\mathbf{c}\|_2 := \left(\sum_{n=1}^N |\mathrm{c}_n|^2\right)^{\frac{1}{2}}$, respectively. The spaces $\ell_2(\mathbb{Z}^d)$ and $L_2(\mathbb{R}^d)$ contain finite-energy sequences and functions, respectively. The proximal operator of a convex functional $\mathcal{R} : \mathbb{R}^N \to \mathbb{R}$ is defined as $\mathrm{prox}_{\mathcal{R}}(\mathbf{z}; \mu) := \mathrm{argmin}_{\mathbf{s}} \left(\frac{1}{2}\|\mathbf{s} - \mathbf{z}\|_2^2 + \mu\mathcal{R}(\mathbf{s})\right)$, with $\mu \in \mathbb{R}^+$. The Fourier transform of f is $\widehat{f}$. The reflection of a function f is denoted $f^\vee = f(-\cdot)$. Finally, the projection orientations and the in-plane translations are referred to as "latent variables".

II. Cryo-EM Imaging Model

A. Imaging Model for a Single Orientation

Let $V \in L_2(\mathbb{R}^3)$ denote the 3D density map of a molecule and let $\Omega_{2\mathrm{D}} \subset \mathbb{Z}^2$ be the discretized projection domain (see Figure 2). The number of elements in $\Omega_{2\mathrm{D}}$ is $M = \sharp\Omega_{2\mathrm{D}}$.

Authorized licensed use limited to: EPFL LAUSANNE. Downloaded on May 08,2020 at 08:27:07 UTC from IEEE Xplore. Restrictions apply.

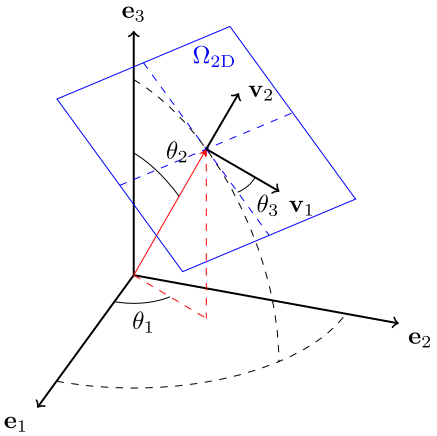

Fig. 2. 3D Geometry of the imaging model for an orientation $\theta = (\theta_1, \theta_2, \theta_3)$. The Euler angles θ_1, θ_2, and θ_3 represent the rotation, the tilt, and the in-plane rotation in the projection plane, respectively.

We model a cryo-EM projection image $g : \Omega_{2D} \to \mathbb{R}$ for an orientation $\theta = (\theta_1, \theta_2, \theta_3) \in [0, 2\pi) \times [0, \pi] \times [0, 2\pi)$ and an in-plane translation $\mathbf{t} = (t_1, t_2) \in \mathbb{R}^2$ as

$$g[\mathbf{m}] = (h * \mathcal{P}_\theta(V))(\mathbf{\Lambda m} - \mathbf{t}) + \varepsilon[\mathbf{m}], \tag{1}$$

where $\mathbf{\Lambda} = \mathbf{diag}(\Delta_1, \Delta_2)$ is a diagonal matrix formed out of the sampling steps Δ_1 and Δ_2 of the projection domain and $\varepsilon : \Omega_{2D} \to \mathbb{R}$ is an additive Gaussian white noise with zero mean and σ^2 variance. The operator $\mathcal{P}_\theta : L_2(\mathbb{R}^3) \to L_2(\mathbb{R}^2)$ is the projection operator for the orientation θ and $h \in L_2(\mathbb{R}^2)$ corresponds to the PSF of the microscope. The vectorization of g is $\mathbf{g} = (g[\mathbf{m}])_{\mathbf{m}\in\Omega_{2D}}$ so that $\mathbf{g} \in \mathbb{R}^M$. The same goes for ε and $\boldsymbol{\varepsilon}$.

B. Discretization

To discretize the 3D density map V, we follow a generalized sampling scheme [27] and define

$$V(\mathbf{x}) = \sum_{\mathbf{k}\in\mathbb{Z}^3} c[\mathbf{k}]\varphi(\mathbf{x} - \mathbf{k}), \ \forall \mathbf{x} \in \mathbb{R}^3, \tag{2}$$

where $\varphi \in L_2(\mathbb{R}^3)$ is a given basis function and $c[\cdot] \in \ell_2(\mathbb{Z}^3)$ is a sequence that contains the coefficients of V in the reconstructing space. The sampling step in the object domain is assumed to be equal to one, without loss of generality.

In this work, we choose φ to be the optimized Kaiser-Bessel window function (KBWF) [28]

$$\varphi(\mathbf{x}) = \begin{cases} \dfrac{\beta_a(\|\mathbf{x}\|)^m I_m\left(\alpha\beta_a(\|\mathbf{x}\|)\right)}{I_m(\alpha)}, & \|\mathbf{x}\| \in [0, a] \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

where $\beta_a(r) = \sqrt{1 - (r/a)^2}$, $a > 0$ is the support radius, $\alpha > 0$ the window taper, and I_m the modified Bessel function of order m. KBWFs are well suited for tomographic reconstruction in reason of their isotropy and compact support [13], [28]. Moreover, the x-ray transform of KBWF does not depend on the orientation θ and admits a closed-form expression [29]. It was shown in [28] that a KBWF represents functions very effectively when using specific parameter values (*e.g.*, $m = 2$, $a = 4$, and $\alpha = 19$).

Because the density map V is compactly supported, the sequence $c[\cdot] \in \ell_2(\mathbb{Z}^3)$ can be restricted to a finite number of nonzero coefficients $\mathbf{c} = (c[\mathbf{k}])_{\mathbf{k}\in\Omega_{3D}}$, where $\Omega_{3D} \subset \mathbb{Z}^3$ and $N = \sharp\Omega_{3D}$.

We then substitute (2) in (1), and use the linearity and the pseudo-translation invariance of the x-ray transform [30] to obtain a discrete version of the forward model, as in

$$g[\mathbf{m}] = \sum_{\mathbf{k}\in\Omega_{3D}} c[\mathbf{k}](h * \mathcal{P}_\theta(\varphi))(\mathbf{\Lambda m} - \mathbf{M}_{\theta^\perp}\mathbf{k} - \mathbf{t}) + \varepsilon[\mathbf{m}]. \tag{4}$$

Here, $\mathbf{M}_{\theta^\perp} \in \mathbb{R}^{2\times 3}$ is the orthogonal projector operator

$$\mathbf{M}_{\theta^\perp} = \begin{pmatrix} C_1C_2C_3 - S_1S_3 & C_3S_1 + C_1C_2S_3 & -C_1S_2 \\ -C_1S_3 - C_2C_3S_1 & C_1C_3 - C_2S_1S_3 & S_1S_2 \end{pmatrix}, \tag{5}$$

where $\forall i = \{1, 2, 3\}$, $C_i = \cos(\theta_i)$ and $S_i = \sin(\theta_i)$.

Finally, we write (4) as

$$\mathbf{g} = \mathbf{H}(\theta, \mathbf{t})\,\mathbf{c} + \boldsymbol{\varepsilon}, \tag{6}$$

where $\mathbf{H}(\theta, \mathbf{t}) \in \mathbb{R}^{M\times N}$ is the discrete imaging operator for orientation θ and in-plane translation $\mathbf{t}$.

C. Global Imaging Model

We now consider a set of P projection images (indexed as $\mathbf{g}_p$) such that $\mathbf{g} = \{\mathbf{g}_p \in \mathbb{R}^M\}_{p=1}^P$. Similarly, the set of projection orientations is defined as $\mathbf{\Theta} = \{\theta_p \in \Theta\}_{p=1}^P$ and the set of in-plane translations as $\mathbf{\Gamma} = \{\mathbf{t}_p \in \mathbb{R}^2\}_{p=1}^P$.

The global imaging model is thus given by

$$\mathbf{g} = \mathbf{H}(\mathbf{\Theta}, \mathbf{\Gamma})\,\mathbf{c} + \boldsymbol{\varepsilon}, \tag{7}$$

where

$$\mathbf{g} = \begin{bmatrix} \mathbf{g}_1 \\ \vdots \\ \mathbf{g}_P \end{bmatrix}, \quad \mathbf{H}(\mathbf{\Theta}, \mathbf{\Gamma}) = \begin{bmatrix} \mathbf{H}(\theta_1, \mathbf{t}_1) \\ \vdots \\ \mathbf{H}(\theta_P, \mathbf{t}_P) \end{bmatrix}, \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \boldsymbol{\varepsilon}_1 \\ \vdots \\ \boldsymbol{\varepsilon}_P \end{bmatrix} \tag{8}$$

For the sake of clarity we shall thereafter use the notations $\mathbf{H}^T\mathbf{H}(\mathbf{\Theta}, \mathbf{\Gamma}) = (\mathbf{H}(\mathbf{\Theta}, \mathbf{\Gamma}))^T\mathbf{H}(\mathbf{\Theta}, \mathbf{\Gamma})$ and $\mathbf{H}^T(\mathbf{\Theta}, \mathbf{\Gamma}) = (\mathbf{H}(\mathbf{\Theta}, \mathbf{\Gamma}))^T$.

III. Joint Angular Refinement and Reconstruction

A. Joint-Optimization Framework

Our goal is to jointly estimate the unknown variables in (7), which are the coefficients $\mathbf{c}$ of the density map, the projection orientations $\mathbf{\Theta}$, and the in-plane translations $\mathbf{\Gamma}$. To do so, we express the refinement procedure as a regularized least-squares minimization

$$(\hat{\mathbf{c}}, \hat{\mathbf{\Theta}}, \hat{\mathbf{\Gamma}}) \in \left\{ \arg\min_{\mathbf{c},\mathbf{\Theta},\mathbf{\Gamma}} \mathcal{J}(\mathbf{c}, \mathbf{\Theta}, \mathbf{\Gamma}) \right\}, \tag{9}$$

where

$$\mathcal{J}(\mathbf{c}, \mathbf{\Theta}, \mathbf{\Gamma}) = \frac{1}{2}\|\mathbf{g} - \mathbf{H}(\mathbf{\Theta}, \mathbf{\Gamma})\,\mathbf{c}\|_2^2 + \lambda\mathcal{R}(\mathbf{Lc}). \tag{10}$$

Authorized licensed use limited to: EPFL LAUSANNE. Downloaded on May 08,2020 at 08:27:07 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1 Joint-Optimization Framework

Require: $\mathbf{c}^0$,$\mathbf{\Theta}^0$,$\mathbf{\Gamma}^0$

1: $k = 0$
2: **while** not converged **do**
▷ Update the density map:
3: $\mathbf{c}^{k+1} = \text{ADMM}\left(\mathcal{J}(\cdot, \mathbf{\Theta}^k, \mathbf{\Gamma}^k), \mathbf{c}^k\right)$
▷ Update the latent variables:
4: $(\mathbf{\Theta}^{k+1}, \mathbf{\Gamma}^{k+1}) = \text{GD}\left(\mathcal{J}(\mathbf{c}^{k+1}, \cdot, \cdot), \mathbf{\Theta}^k, \mathbf{\Gamma}^k\right)$
5: $k \leftarrow k + 1$
6: **end while**

Return: $\mathbf{c}^k, \mathbf{\Theta}^k, \mathbf{\Gamma}^k$

Here, $\mathcal{R}$ is a sparsity-promoting functional and $\mathbf{L}$ a linear operator. Together, they are used to inject prior knowledge into the reconstruction process. As an example, setting $\mathbf{L} = \mathbf{\nabla}$ (i.e., the gradient operator) and $\mathcal{R} = \|\cdot\|_{2,1}$ leads to the popular total-variation (TV) regularization [31]. The regularization parameter λ controls the balance between the data-fidelity and the regularization terms.

To solve (9), we alternate between the minimization over $\mathbf{c}$ and the minimization over $\mathbf{\Theta}$ and $\mathbf{\Gamma}$. Although the objective function in (9) is convex with respect to $\mathbf{c}$, it is not convex with respect to the latent variables $\mathbf{\Theta}$ and $\mathbf{\Gamma}$. Moreover, it is smooth with respect to $\mathbf{\Theta}$ and $\mathbf{\Gamma}$, but usually not smooth (due to $\mathcal{R}$) with respect to $\mathbf{c}$. This dictates the choice of two different minimization procedures within the proposed alternating scheme.

For the minimization of $\mathcal{J}$ with respect to $\mathbf{c}$, we use ADMM [32], which allows us to split the problem into a sequence of simpler subproblems (see Section III-B). Then, taking benefit from the differentiability of $\mathcal{J}$ with respect to $\mathbf{\Theta}$ and $\mathbf{\Gamma}$, the latent variables are updated using gradient-descent with line-search (see Section III-C). The outline of this joint optimization procedure is given in Algorithm 1 and is implemented within the framework of the GlobalBioIm library[1] [33].

Note that, at Line 3 of Algorithm 1, we use the notation $\text{ADMM}\left(\mathcal{J}(\cdot, \mathbf{\Theta}^k, \mathbf{\Gamma}^k), \mathbf{c}^k\right)$ to refer to the minimization of $\mathcal{J}(\cdot, \mathbf{\Theta}^k, \mathbf{\Gamma}^k)$ using ADMM initialized with $\mathbf{c}^k$. We do the same for the gradient-descent algorithm (see Line 4).

B. Update of the Density Map

Given $\mathbf{\Theta}$ and $\mathbf{\Gamma}$, the reconstruction task consists in solving

$$\hat{\mathbf{c}} = \arg\min_{\mathbf{c}} \ \mathcal{J}(\mathbf{c}, \mathbf{\Theta}, \mathbf{\Gamma}). \tag{11}$$

To do so, we use the ADMM scheme proposed in [13]. The core idea is to split (11) by introducing an auxiliary variable $\mathbf{u}$ so that

$$\hat{\mathbf{c}} = \arg\min_{\mathbf{c}} \ \left(\frac{1}{2}\|\mathbf{g} - \mathbf{H}(\mathbf{\Theta}, \mathbf{\Gamma})\,\mathbf{c}\|_2^2 + \lambda\mathcal{R}(\mathbf{u})\right) \quad s.t.\ \mathbf{u} = \mathbf{L}\mathbf{c}. \tag{12}$$

Then, the ADMM algorithm alternates between three steps, as summarized in Algorithm 2.

[1] http://bigwww.epfl.ch/algorithms/globalbioim/

Algorithm 2 ADMM (Update of the Density Map)

Require: $\mathbf{g}$, $\mathbf{\Theta}$, $\mathbf{\Gamma}$, $\mathbf{c}^0$, $\rho > 0$

1: $\mathbf{u}^0 = \mathbf{L}\mathbf{c}^0$, $\tilde{\mathbf{u}}^0 = \mathbf{u}^0$
2: $k = 0$
3: **while** $k < K_{\text{ADMM}}$ **do**
4: $\mathbf{u}^{k+1} = \text{prox}_{\mathcal{R}}\left(\mathbf{L}\mathbf{c}^k - \tilde{\mathbf{u}}^k/\rho;\, \lambda/\rho\right)$
5: $\mathbf{b} = (\mathbf{H}(\mathbf{\Theta}, \mathbf{\Gamma}))^T\mathbf{g} + \rho\mathbf{L}^T(\mathbf{u}^{k+1} - \tilde{\mathbf{u}}^k/\rho)$
6: $\mathbf{c}^{k+1} = \left(\mathbf{H}^T\mathbf{H}(\mathbf{\Theta}, \mathbf{\Gamma}) + \rho\mathbf{L}^T\mathbf{L}\right)^{-1}\mathbf{b}$
7: $\tilde{\mathbf{u}}^{k+1} = \tilde{\mathbf{u}}^k + \rho\left(\mathbf{u}^{k+1} - \mathbf{L}\mathbf{c}^{k+1}\right)$
8: $k \leftarrow k + 1$
9: **end while**

Return: $\mathbf{c}^{K_{\text{ADMM}}}$

When TV regularization is used, the proximal operator at Line 4 admits a closed-form expression that can be computed efficiently [34]. Then, the linear step at Line 6 is solved iteratively using a conjugate-gradient algorithm together with a fast formulation of the $\mathbf{H}^T\mathbf{H}(\mathbf{\Theta}, \mathbf{\Gamma})$ term [13]. Finally, Line 7 corresponds to a simple update of the dual variable $\tilde{\mathbf{u}}$, while $\rho > 0$ is a penalty parameter.

For the sake of completeness, the full set of equations behind the reconstruction algorithm is provided in Appendix A.

C. Update of the Latent Variables

Let us first remark that the least-squares term in (10) can be written as

$$\frac{1}{2}\|\mathbf{g} - \mathbf{H}(\mathbf{\Theta}, \mathbf{\Gamma})\,\mathbf{c}\|_2^2 = \frac{1}{2}\sum_{p=1}^{P}\|\mathbf{g}_p - \mathbf{H}(\theta_p, \mathbf{t}_p)\,\mathbf{c}\|_2^2. \tag{13}$$

Hence, when $\mathbf{c}$ is fixed, the minimization of $\mathcal{J}(\mathbf{c}, \cdot, \cdot)$ amounts to solve

$$(\hat{\theta}_p, \hat{\mathbf{t}}_p) \in \left\{\arg\min_{\theta, \mathbf{t}} \ \mathcal{J}_p(\theta, \mathbf{t})\right\} \tag{14}$$

for all $p \in \{1, \ldots, P\}$, where $\mathcal{J}_p : (\theta, \mathbf{t}) \mapsto \mathbb{R}^+ \cup \{0\}$ is defined as

$$\mathcal{J}_p(\theta, \mathbf{t}) = \frac{1}{2}\|\mathbf{g}_p - \mathbf{H}(\theta, \mathbf{t})\,\mathbf{c}\|_2^2. \tag{15}$$

As the objective function $\mathcal{J}_p$ is differentiable, the minimization in (14) can be achieved using gradient-descent steps. Hence, we first need to compute the gradients

$$\mathbf{\nabla}_{\theta}\mathcal{J}_p(\theta, \mathbf{t}) = \left(\frac{\partial\mathcal{J}_p}{\partial\theta_1}(\theta, \mathbf{t}), \frac{\partial\mathcal{J}_p}{\partial\theta_2}(\theta, \mathbf{t}), \frac{\partial\mathcal{J}_p}{\partial\theta_3}(\theta, \mathbf{t})\right) \tag{16}$$

$$\mathbf{\nabla}_{\mathbf{t}}\mathcal{J}_p(\theta, \mathbf{t}) = \left(\frac{\partial\mathcal{J}_p}{\partial t_1}(\theta, \mathbf{t}), \frac{\partial\mathcal{J}_p}{\partial t_2}(\theta, \mathbf{t})\right). \tag{17}$$

The explicit expressions of these quantities are provided in Theorem 1.

Theorem 1: Let φ be an isotropic kernel and $\mathbf{H}(\theta, \mathbf{t}) \in \mathbb{R}^{M\times N}$ be defined by (4). Then, for $\upsilon \in \{\theta_1, \theta_2, \theta_3, t_1, t_2\}$, there exists $\mathbf{r}_\upsilon \in \mathbb{R}^N$ and $\mathbf{q}_\upsilon \in \mathbb{R}^N$ such that

$$\frac{\partial\mathcal{J}_p}{\partial\upsilon}(\theta, \mathbf{t}) = \frac{1}{2}\mathbf{c}^T\left(\mathbf{r}_\upsilon * \mathbf{c} - 2\mathbf{q}_\upsilon\right). \tag{18}$$

Moreover, $\forall\mathbf{k} \in \mathbf{\Omega}_{3\text{D}}$,

Authorized licensed use limited to: EPFL LAUSANNE. Downloaded on May 08,2020 at 08:27:07 UTC from IEEE Xplore. Restrictions apply.

Algorithm 3 GD (Update of the Latent Variables)

Require: $\alpha_{\boldsymbol{\theta}}^0 > 0$, $\alpha_{\mathbf{t}}^0 > 0$, $\eta \in (0,1)$, $\boldsymbol{\Theta}^0$, $\boldsymbol{\Gamma}^0$, $\mathbf{c}$

1: **for** $p = 1, \dots, P$
2: $k = 0$
3: **while** $k < K_{\mathrm{GD}}$
▷ Update $\boldsymbol{\theta}_p$
4: $\alpha_{\boldsymbol{\theta}} \leftarrow \alpha_{\boldsymbol{\theta}}^0$
5: $\boldsymbol{\theta}_p^{k+1} = \boldsymbol{\theta}_p^k - \alpha_{\boldsymbol{\theta}} \boldsymbol{\nabla}_{\boldsymbol{\theta}} \mathcal{J}_p(\boldsymbol{\theta}_p^k, \mathbf{t}_p^k)$
6: **while** $\mathcal{J}_p(\boldsymbol{\theta}_p^{k+1}, \mathbf{t}_p^k) > \mathcal{J}_p(\boldsymbol{\theta}_p^k, \mathbf{t}_p^k)$
7: $\alpha_{\boldsymbol{\theta}} \leftarrow \eta \alpha_{\boldsymbol{\theta}}$
8: $\boldsymbol{\theta}_p^{k+1} = \boldsymbol{\theta}_p^k - \alpha_{\boldsymbol{\theta}} \boldsymbol{\nabla}_{\boldsymbol{\theta}} \mathcal{J}_p(\boldsymbol{\theta}_p^k, \mathbf{t}_p^k)$
9: **end while**
▷ Update $\mathbf{t}_p$
10: $\alpha_{\mathbf{t}} \leftarrow \alpha_{\mathbf{t}}^0$
11: $\mathbf{t}_p^{k+1} = \mathbf{t}_p^k - \alpha_{\mathbf{t}} \boldsymbol{\nabla}_{\mathbf{t}} \mathcal{J}_p(\boldsymbol{\theta}_p^{k+1}, \mathbf{t}_p^k)$
12: **while** $\mathcal{J}_p(\boldsymbol{\theta}_p^{k+1}, \mathbf{t}_p^{k+1}) > \mathcal{J}_p(\boldsymbol{\theta}_p^{k+1}, \mathbf{t}_p^k)$
13: $\alpha_{\mathbf{t}} \leftarrow \eta \alpha_{\mathbf{t}}$
14: $\mathbf{t}_p^{k+1} = \mathbf{t}_p^k - \alpha_{\mathbf{t}} \boldsymbol{\nabla}_{\mathbf{t}} \mathcal{J}_p(\boldsymbol{\theta}_p^{k+1}, \mathbf{t}_p^k)$
15: **end while**
16: $k \leftarrow k + 1$
17: **end while**
18: **end for**

Return: $\boldsymbol{\Theta}^{K_{\mathrm{GD}}}, \boldsymbol{\Gamma}^{K_{\mathrm{GD}}}$

- if $\upsilon = \theta_i$ for $i \in \{1, 2, 3\}$,

$$r_\upsilon[\mathbf{k}] = \frac{1}{\det(\boldsymbol{\Lambda})} \left(\frac{\partial \mathbf{M}_{\boldsymbol{\theta}^\perp}}{\partial \theta_i} \mathbf{k} \right)^T \boldsymbol{\nabla} \left(\psi * \psi^\vee \right) (\mathbf{M}_{\boldsymbol{\theta}^\perp} \mathbf{k}), \quad (19)$$

$$q_\upsilon[\mathbf{k}] = \frac{1}{\det(\boldsymbol{\Lambda})} \left(\frac{\partial \mathbf{M}_{\boldsymbol{\theta}^\perp}}{\partial \theta_i} \mathbf{k} \right)^T \boldsymbol{\nabla} (g_p * \psi^\vee) (\mathbf{M}_{\boldsymbol{\theta}^\perp} \mathbf{k} + \mathbf{t}), \quad (20)$$

- if $\upsilon = t_j$ for $j \in \{1, 2\}$,

$$r_\upsilon[\mathbf{k}] = 0, \quad (21)$$

$$q_\upsilon[\mathbf{k}] = \frac{1}{\det(\boldsymbol{\Lambda})} \frac{\partial (g_p * \psi^\vee)}{\partial y_j} (\mathbf{M}_{\boldsymbol{\theta}^\perp} \mathbf{k} + \mathbf{t}), \quad (22)$$

where $\psi : \mathbf{y} = (y_1, y_2) \mapsto (h * \mathcal{P}(\varphi))(\mathbf{y})$, $\frac{\partial \mathbf{M}_{\boldsymbol{\theta}^\perp}}{\partial \theta_i} \in \mathbb{R}^{2\times 3}$ contains the entry-wise derivatives of $\mathbf{M}_{\boldsymbol{\theta}^\perp}$, and g_p denotes the continuous counterpart of $\mathbf{g}_p$ (*i.e.* interpolated values).

The proof is given in Appendix B and includes details on the gradients of $\psi * \psi^\vee$ and $g_p * \psi^\vee$. In particular, we show that they depend on $\mathcal{P}(\varphi)$ and $\frac{\partial \mathcal{P}(\varphi)}{\partial y_j}$ whose expressions are provided in Proposition 1 for the specific case of the KBWF φ in (3).

Proposition 1: For the KBWF φ given in (3), we have

$$\mathcal{P}(\varphi)(\mathbf{y}) = a A \, \beta_a(\|\mathbf{y}\|)^{m+\frac{1}{2}} I_{m+\frac{1}{2}} \left(\alpha \beta_a(\|\mathbf{y}\|) \right), \quad (23)$$

$$\frac{\partial \mathcal{P}(\varphi)}{\partial y_\upsilon}(\mathbf{y}) = -\frac{\alpha y_\upsilon A}{a} \beta_a(\|\mathbf{y}\|)^{m-\frac{1}{2}} I_{m-\frac{1}{2}}(\alpha \beta_a(\|\mathbf{y}\|)), \quad (24)$$

where $A = \frac{\sqrt{2\pi/\alpha}}{I_m(\alpha)}$.

The proof is given in Appendix C.

Equipped with those gradient expressions, we deploy a semi-coordinate-wise gradient-descent to solve (14), as summarized in Algorithm 3. At each iteration, the parameters $\boldsymbol{\theta}$ and $\mathbf{t}$ are updated sequentially, which allows for the use of different stepsizes between orientation and in-plane translation. This is crucial to account for the different dynamics between these two variables. Moreover, we use adaptive steps that are selected according to a backtracking line-search method [35], [36]. Given an initial value, the step is decreased through the parameter $\eta \in (0, 1)$ until the cost that corresponds to the updated variable is smaller than its current value (conditions checked in Steps 6 and 12).

Finally, to further accelerate the update of the latent variables, we divide the projection set $\{1, \dots, P\}$ into mini-batches and process them in parallel. It is the separability of the objective function in (14), related to the independence of projection images, that makes this parallelization possible.

D. Computational Complexity

We compare the computational complexity of the proposed latent variable update to that of projection matching. Let n and m be such that $N = n^3$ and $M = m^2$ (*i.e.* , $\mathbf{c} \in \mathbb{R}^{n\times n\times n}$ and $\mathbf{g}_p \in \mathbb{R}^{m\times m}$).

1) Projection Matching: Each iteration of projection matching consists of two steps.

1) *Generation of Clean Templates.* Given the current density map $\mathbf{c}$, evaluate the forward model $\mathbf{H}(\boldsymbol{\theta}, \mathbf{0}_{\mathbb{R}^2})\, \mathbf{c}$ for $N_{\theta_1} N_{\theta_2}$ different values of $\boldsymbol{\theta} = (\theta_1, \theta_2, 0)$ obtained by sampling $[0, 2\pi)$ with N_{θ_1} points and $[0, \pi]$ with N_{θ_2} points. Denoting by C_H the cost of one evaluation of the forward model, the computational complexity of this step is $O(N_{\theta_1} N_{\theta_2} C_H)$.
2) *Matching Projection Images* $\mathbf{g}_p$. Each projection image $\mathbf{g}_p$ ($p \in \{1, \dots, P\}$) is compared against $N_{\theta_1} N_{\theta_2}$ clean templates. This requires rotation and in-plane translation alignment whose complexity is $O(m^2 \log m)$ if done efficiently using polar Fourier transform [37], spherical harmonics [38], or steerable basis functions [39].

The overall complexity of template matching is thus given by $O(N_{\theta_1} N_{\theta_2} (P m^2 \log m + C_H))$. The cost of C_H depends on its implementation. An efficient way to compute it can, for example, rely on the Fourier-slice theorem and the use of non-uniform FFT. This strategy roughly requires one 3D-FFT of the volume $\mathbf{c}$, one interpolation step to extract the central slice perpendicular to the projection direction, and one inverse 2D-FFT of this slice. This gives $C_H = n^3 \log n + m^2 + m^2 \log m$. With such an implementation, the overall complexity would thus be $O(N_{\theta_1} N_{\theta_2} (P m^2 \log m + n^3 \log n))$.

2) Proposed Update Scheme: According to (18) in Theorem 1, the evaluation of the partial derivative $\partial \mathcal{J}_p / \partial \upsilon$ can be done at the cost of a 3D convolution (only required when $\upsilon = \theta_i$), a component-wise subtraction, and a scalar product. This gives a complexity of $O(n^3 \log n)$. To this has to be added the cost of computing $\mathbf{r}_\upsilon$ and $\mathbf{q}_\upsilon$ in Theorem 1. First, let us remark that $\frac{\partial \mathbf{M}_{\boldsymbol{\theta}^\perp}}{\partial \theta_i}$ is known in closed form from (5). Hence, the complexity of computing $\frac{\partial \mathbf{M}_{\boldsymbol{\theta}^\perp}}{\partial \theta_i} \mathbf{k}$ for all $\mathbf{k} \in \Omega_{3\mathrm{D}}$, is $O(n^3)$. Then, we distinguish two situations:

Authorized licensed use limited to: EPFL LAUSANNE. Downloaded on May 08,2020 at 08:27:07 UTC from IEEE Xplore. Restrictions apply.

- *Explicit Expressions of* $\nabla\left(\psi * \psi^{\vee}\right)$ *and*[2] $\nabla(g_p * \psi^{\vee})$ *are Known*. Given Θ and $\mathbf{t}$, the computation of $\mathbf{r}_v$ and $\mathbf{q}_v$ amounts to their sampling at points $\mathbf{M}_{\Theta^{\perp}}\mathbf{k}$ (or $\mathbf{M}_{\Theta^{\perp}}\mathbf{k}+\mathbf{t}$), for $\mathbf{k} \in \Omega_{3\mathrm{D}}$, followed by a scalar product with $\frac{\partial \mathbf{M}_{\Theta^{\perp}}}{\partial \theta_i}\mathbf{k}$, resulting in an overall complexity of $O(n^3)$.
- *Explicit Expressions of* $\nabla\left(\psi * \psi^{\vee}\right)$ *and* $\nabla(g_p * \psi^{\vee})$ *are Not Known*. Due to their independence upon Θ and $\mathbf{t}$, the relevant quantities can thus be evaluated *once* (optionally upsampled) on the grid $\Omega_{2\mathrm{D}}$ using (39) together with Proposition 1 and discrete convolutions (complexity of $O(m^2 \log m)$). Having this precomputed quantity saved as a lookup table, the evaluation of $\nabla\left(\psi * \psi^{\vee}\right)$ and $\nabla(g_p * \psi^{\vee})$ at points $\mathbf{M}_{\Theta^{\perp}}\mathbf{k}$ (or $\mathbf{M}_{\Theta^{\perp}}\mathbf{k} + \mathbf{t}$) is done by interpolation. Hence, here again, the computational complexity is $O(n^3)$.

Considering that there are P projection images and that K_{GD} iterations of gradient descent are performed at each update of the latent variables (see Algorithm 1), we obtain an overall complexity of $O(PK_{\mathrm{GD}}n^3 \log n)$.

Finally, given that K_{GD} is typically small (for example in our experiments $K_{\mathrm{GD}} = 3$) and that it is recommended [20] to set $N_{\theta_1}N_{\theta_2}$ in the order of n^2 to maintain a precise estimation of projection angles, the proposed method offers an interesting improvement in runtime over projection matching.

IV. Experiments

A. Datasets

We test our algorithm on two synthetic datasets. The first dataset corresponds to the Holliday junction complex (HJC) density map, while the second corresponds to the Human patched 1 (PTCH1) protein. For each dataset, we generate the synthetic ground truth V from the submitted density map [40], [41], along with the associated atomic model in the Protein Data Bank using Chimera [42]. The sizes of the HJC and PTCH1 volumes used in our simulations are $(90\times90\times90)$ and $(84 \times 84 \times 84)$, with voxel sizes of 2.867Å and 1.8Å, respectively. We also synthesize a higher resolution version of HJC with size $124 \times 124 \times 124$ and voxel size 2Å. The first two volumes are used in our proof of concept simulations; the last volume is used in an experiment that mimics more realistic cryo-EM conditions (section V.E).

From those ground truths, we then generate P projection images according to the image-formation model provided in (1). We sample the orientation space using P points in an equi-distributed fashion over $\{(\theta_{1,p}, \theta_{2,p})\}_{p=1}^{P}$. The in-plane rotations are generated by uniformly sampling P points on a $[0, 2\pi)$ interval. To perform in-plane translations, we move the center of the projection images randomly by at most $m_{\mathbf{t}}$ pixels in either horizontal or vertical directions. In our experiments, we use at most 20000 projection images to demonstrate the feasibility of our method. Finally, the projection images are corrupted with additive Gaussian noise with zero mean and variance σ^2. The average signal-to-noise ratio (SNR) across all projection images is then given by $\mathrm{SNR}_{\mathrm{data}} = \frac{1}{P}\sum_{p=1}^{P}\frac{\|\mathbf{g}_p^{\star}\|_2^2}{\sigma^2}$ where $\mathbf{g}_p^{\star}$ correspond to noiseless measurements.

[2]Note that $\frac{\partial(g_p * \psi^{\vee})}{\partial y_j}$ is nothing else than the jth component of $\nabla(g_p * \psi^{\vee})$.

B. Initial Density Map, Orientations, and In-Plane Translations

To generate an approximate density map from which to start the refinement procedure, we use the initial density map generated by 3D *ab-initio* model in Relion [43]. For the projection orientations, we consider two possible initializations.

- Model *Init-1*: We add a zero-mean random variable uniformly distributed in $[-e_\theta, e_\theta]$ to the ground-truth orientations, *i.e.*, $\Theta_p^{\mathrm{init}} = \Theta_p^{\mathrm{true}} + \varepsilon_{\theta,p,j} \sim \mathrm{Unif}(-e_\theta, e_\theta)$, $p \in \{1, \ldots, P\}$ and $j \in \{1, 2, 3\}$.
- Model *Init-2*: We use projection-matching (or another angular assignment method) to assign the initial projection orientations Θ_p^{init}, $p \in \{1, \ldots, P\}$. For this initialization, we use the angular assignments from the 3D ab-initio modeling in Relion.

The in-plane translations $\{\mathbf{t}_p\}_{p=1}^{P}$ are all initialized by zeros. With our notations we have that $\mathbf{\Theta}^{\mathrm{init}} = \{\Theta_p^{\mathrm{init}}\}_{p=1}^{P}$, $\mathbf{\Theta}^{\mathrm{true}} = \{\Theta_p^{\mathrm{true}}\}_{p=1}^{P}$, $\mathbf{\Gamma}^{\mathrm{init}} = \{\mathbf{0}\}_{p=1}^{P}$, and $\mathbf{\Gamma}^{\mathrm{true}} = \{\mathbf{t}_p^{\mathrm{true}}\}_{p=1}^{P}$.

C. Tuning of the Hyper Parameters

The parameters that need tuning are λ, ρ, and K_{ADMM}, as used in the update of the density map (Algorithm 2), and $\alpha_{\boldsymbol{\theta}}$, $\alpha_{\boldsymbol{t}}$, and K_{GD}, as introduced in the update of the latent variables (Algorithm 3). In our experiments, we use $K_{\mathrm{ADMM}} = 2$ or $K_{\mathrm{ADMM}} = 5$, along with $K_{\mathrm{GD}} = 3$, $\alpha_{\Theta} = 10^{-7}$, $\alpha_{\boldsymbol{t}} = 10^{-5}$, and $\eta = 0.25$. The parameters λ and ρ grow like σ. We use the same set of parameters for the two molecules. Similar to [13], the parameters of the KBWF used in the expansion of the volume in (2)-(3) are $a = 4$, $\alpha = 19$, and $m = 2$.

D. Metrics

To assess the quality of reconstruction, we use the Fourier shell correlation (FSC) between the reconstructed volume V^{rec} and the ground-truth V^{gt}, as defined by

$$\mathrm{FSC}(r) = \frac{\sum_{r_i \in \mathbf{r}} \hat{V}^{\mathrm{rec}}(r_i)\hat{V}^{\mathrm{gt}}(r_i)^*}{\sqrt{\sum_{r_i \in \mathbf{r}} |\hat{V}^{\mathrm{rec}}(r_i)|^2 \sum_{r_i \in \mathbf{r}} |\hat{V}^{\mathrm{gt}}(r_i)|^2}}. \quad (25)$$

where $\mathbf{r} = \{(x_i, y_i, z_i) : |\sqrt{x_i^2 + y_i^2 + z_i^2} - r| \leq \varepsilon_r\}$, for $\varepsilon_r > 0$, denotes the set of all points in the discrete Fourier domain that lie in a spherical shell with inner radius $r - \varepsilon_r$ and outer radius $r + \varepsilon_r$, centered at origin. The FSC thus computes the correlation between two corresponding spherical shells of the density maps in the Fourier domain. Moreover, we use the SNR metric defined as $\mathrm{SNR}(V^{\mathrm{gt}}, V^{\mathrm{rec}}) = 20\log_{10}\frac{\|V^{\mathrm{gt}}\|_2}{\|V^{\mathrm{gt}} - V^{\mathrm{rec}}\|_2}$.

To assess the quality of the 3D orientation refinements, we visualize the deviations of the refined angles from their ground-truth values. In other words, we examine the distribution of $\{\theta_{i,p}^{\mathrm{true}} - \theta_{i,p}^{\mathrm{rec}}\}_{p=1}^{P}$ for $i \in \{1, 2, 3\}$ and compare it to $\{\theta_{i,p}^{\mathrm{true}} - \theta_{i,p}^{\mathrm{init}}\}_{p=1}^{P}$. When the difference between the angles is small (up to some global rotations), the distribution of the differences is more concentrated around zero. On the contrary,

Authorized licensed use limited to: EPFL LAUSANNE. Downloaded on May 08,2020 at 08:27:07 UTC from IEEE Xplore. Restrictions apply.

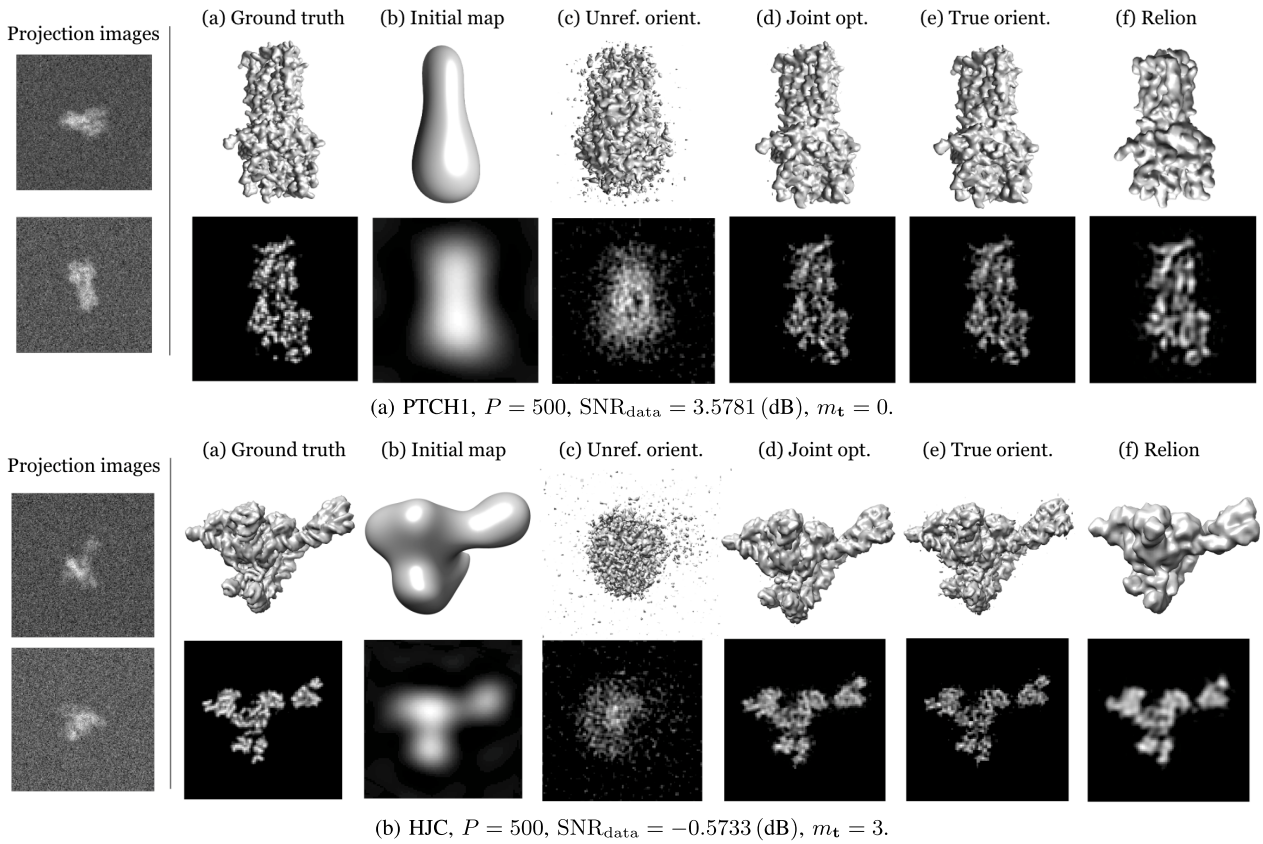

Fig. 3. Reconstructions of PTCH1 and HJC. Left: Samples of the noisy projection images. Top row: 3D structures. Bottom row: Intensity maps of the central slice of the structures. The presented volumes are (a) the ground truth, (b) the initial map, (c) the reconstruction with unrefined projection orientations, (d) the output of our joint refinement approach, (e) the reconstruction with the true projection orientations, (f) the output of Relion after post-processing. For both experiments, the latent variables are initialized following the *Init-1* model with $e_\theta = 0.7$ [rad].

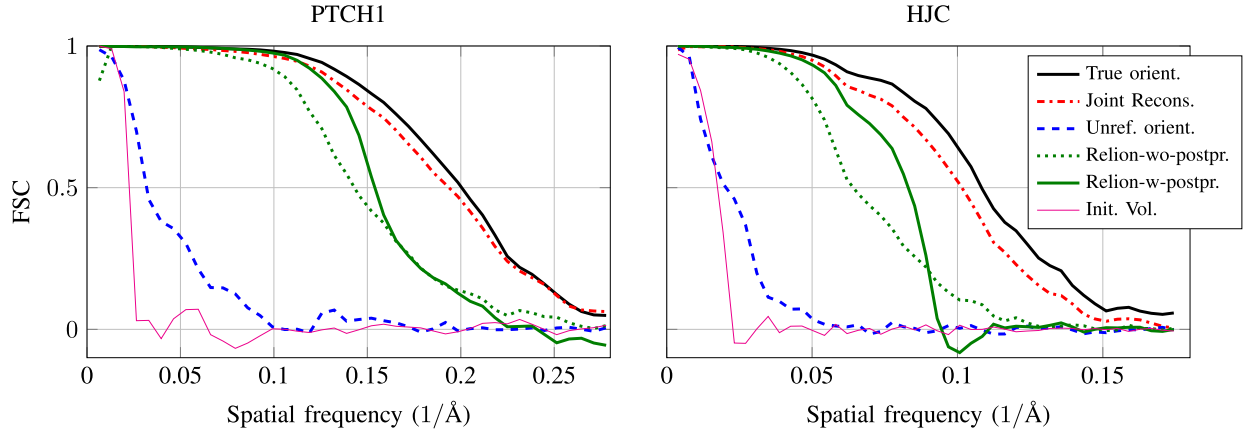

Fig. 4. Comparison between the FSC of the density maps obtained from several baselines and the ground-truth density map. Relion-w-postpr (green solid curve) and Relion-wo-postpr (green dashed curve) refer to the Relion results with and without post-processing, respectively. Note that Relion-wo-postpr is obtained after averaging the two half-maps. The experimental setups are identical to the ones used in Figure 3.

the distribution is more expanded for angles that are further away from their ground-truth values.

E. Compared Methods

We compare our joint-optimization method to the following approaches:

1) *Reconstruction with Unrefined Orientations*. We do not refine the initial angles and directly reconstruct the density map. This gives us an indication of the quality of reconstruction prior to the refinement procedure.
2) *Reconstruction with True Orientations*. We reconstruct the density map with the ground-truth orientations and

Authorized licensed use limited to: EPFL LAUSANNE. Downloaded on May 08,2020 at 08:27:07 UTC from IEEE Xplore. Restrictions apply.

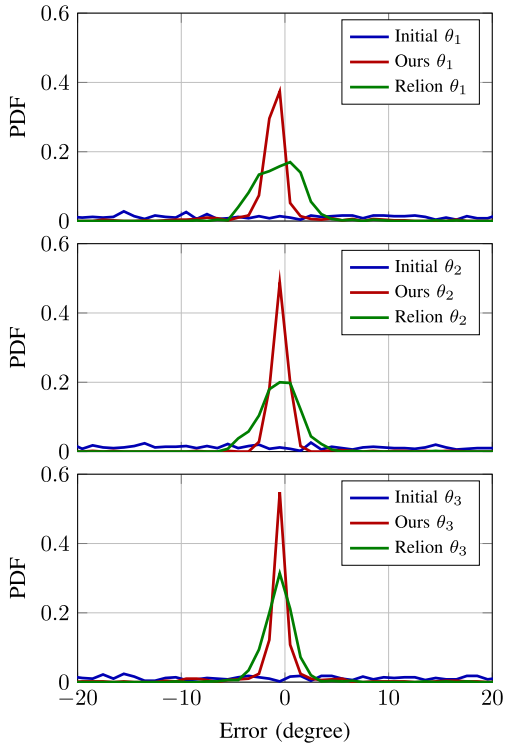

Fig. 5. Probability density function (PDF) of the differences between the true and refined projection orientations by our method $\{\theta_{i,p}^{\text{true}} - \theta_{i,p}^{\text{rec}}\}_{p=1}^{P}$ (red curves), the true and refined projection orientations by Relion $\{\theta_{i,p}^{\text{true}} - \theta_{i,p}^{\text{Relion}}\}_{p=1}^{P}$ (green curves), as well as the true and the initial projection orientations $\{\theta_{i,p}^{\text{true}} - \theta_{i,p}^{\text{init}}\}_{p=1}^{P}$ (blue curves). The experimental setup is identical to Figure 3 (HJC). The x-axis is truncated between -20 and 20 degrees for the sake of clarity.

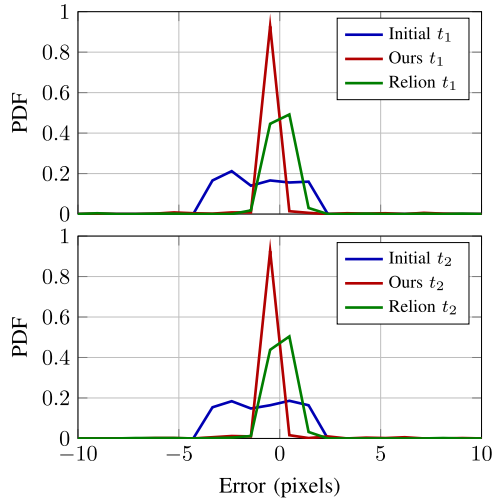

Fig. 6. Probability density function (PDF) of the differences between the true and refined in-plane translations by our method $\{\mathbf{t}_{i,p}^{\text{true}} - \mathbf{t}_{i,p}^{\text{rec}}\}_{p=1}^{P}$ (red curves), the true and refined in-plane translations by Relion $\{\mathbf{t}_{i,p}^{\text{true}} - \mathbf{t}_{i,p}^{\text{Relion}}\}_{p=1}^{P}$ (green curves), as well as the true and the initial in-plane translations $\{\mathbf{t}_{i,p}^{\text{true}} - \mathbf{t}_{i,p}^{\text{init}}\}_{p=1}^{P}$ (blue curves). The experimental setup is identical to Figure 3 (HJC).

in-plane translations. This serves as an oracle benchmark that allows us to quantify the improvement brought by our refinement procedure.

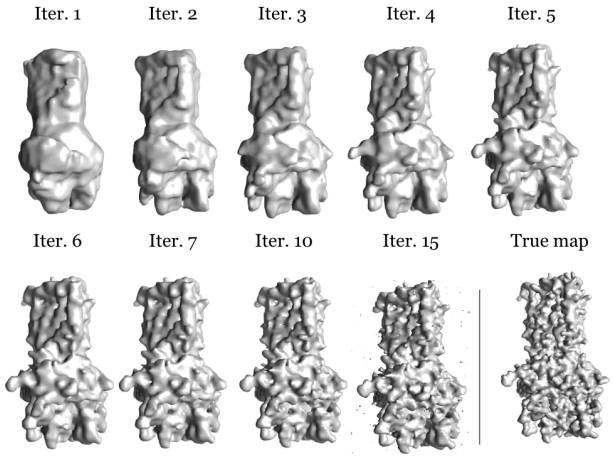

Fig. 7. Evolution through iterations for the refinement of PTCH1. The experimental setup is the same as in Figure 3.

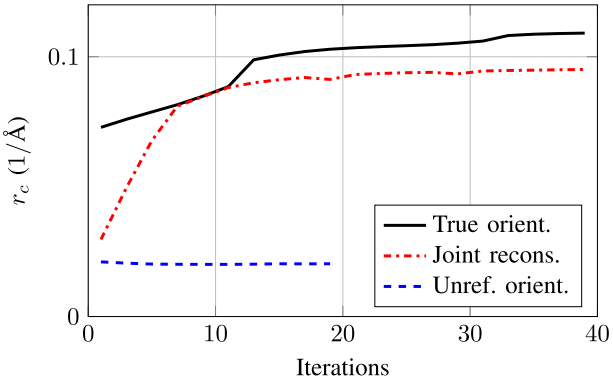

Fig. 8. Evolution of r_c, the radial frequency at which the FSC equals 0.5. Dash-dotted curve: Our joint-reconstruction framework. Solid curve: Reconstruction with true orientation projections. Dashed curve: Reconstruction with unrefined orientation projections. The experimental setup is identical to Figure 3 (HJC).

3) *Reconstruction with the Relion package* [1]. We run the *3D auto-refine* function in Relion (version 2.1.0). The default parameters of this function (*e.g.*, *Initial angular sampling* and *Local searches from auto sampling*) are used. For the particular experiments in which the in-plane translations are zero, the *Initial offset range* and *Initial offset step* parameters are set to their minimum values, which are 0 and 0.1, respectively. Otherwise, they are set to 4 and 0.5, respectively. Note that, to reduce the impact of noise when using Relion, we mask the projection images with a soft circular mask of a diameter that is proportional to the support of the density map.

All reconstructions from Relion are post-processed. We first apply a tight soft mask that embeds the maps. We then low-pass filter the volumes with a cut-off frequency that corresponds to the gold-standard FSC between the two half-maps; this is done using the post-processing function in Relion.

Authorized licensed use limited to: EPFL LAUSANNE. Downloaded on May 08,2020 at 08:27:07 UTC from IEEE Xplore. Restrictions apply.

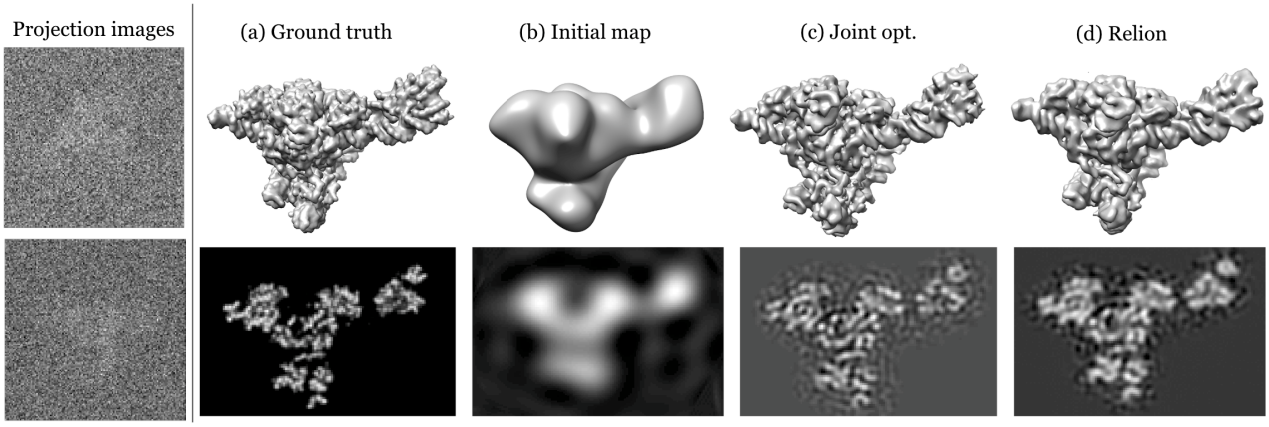

Fig. 9. Reconstructions of HJC. Left: Samples of the noisy projection images. Top row: 3D structures. Bottom row: Intensity maps of the central slice of the structures. The presented volumes are (a) the ground truth, (b) the initial map, (c) the output of our joint refinement approach after post-processing, (d) the output of Relion after post-processing. For this experiments, the latent variables are initialized following the *Init-2* model using the results from Relion 3D ab-initio modeling. The parameters of this experiment are: $P = 20000$, $\text{SNR}_{\text{data}} = -14.2\,\text{dB}$, $m_{\mathbf{t}} = 3$.

V. Results

A. Visual Comparison

We compare in Figure 3 the refined maps obtained using our join optimization scheme (Figure 3-(d)) and the other methods (Figure 3-(c,e,f)). In that experiment, the latent variables are initialized following the *Init-1* model.

As expected, the reconstruction fails when the unrefined 3D orientations are used (Figure 3-(c)). This confirms that angular refinement is required to achieve a successful reconstruction. Predictable as well is the fact that a perfect knowledge of the true 3D poses leads to a successful reconstruction (Figure 3-(e)). It can be clearly seen that the results of our method (Figure 3-(d)) closely resemble the reconstructed map resulting from perfect knowledge of the latent variables. This shows the ability of our method to appropriately refine the density map and the latent variables.

Figure 3-(f) contains the 3D density map refined by Relion. We observe that the map refined through our method is more similar to the ground-truth density map than the Relion output.

B. FSC Curves

The FSC curves of the reconstructed maps are sketched in Figure 4. These curves confirm that our joint-optimization approach (red dash-dotted curve) is able to appropriately refine the low-resolution initial map. Its performance indeed closely approaches that of the reconstruction with perfect knowledge of 3D orientations (solid curve). Moreover, our framework outperforms the Relion outcome with and without post-processing (green curves).

C. Quality of Angular Refinement

In Figure 5, one finds the probability density function (PDF) of the differences between 1) the true and initial projection orientations (blue curve), 2) the true and refined projection orientations by our method (red curve), and 3) the true and refined projection orientations by Relion (green curve). The optimal PDF is obtained when all the differences are zero, up to a global rotation. The corresponding curve resembles a delta function that is one at zero, and zero elsewhere. Based on this, we observe that our proposed method performs well in recovering the projection orientations and outperforms Relion.

Figure 6 compares the PDF of the difference between 1) the true and initial in-plane translations (blue curve), 2) the true and refined in-plane translations by our method (red curve), and 3) the true and refined in-plane translations by Relion (green curve). Here as well, the figure demonstrates the ability of our method to refine in-plane translations, and its increased performance compared to Relion.

D. Convergence Results

The evolution of the density map during refinement is presented in Figure 7. The convergence in terms of resolution of our framework and of two other baselines are shown in Figure 8, where r_c marks the radial frequency at which the FSC between the true and the reconstructed density map equals 0.5.

When the 3D projection orientations are perfectly known (solid curve), the reconstruction process achieves a high-resolution map in twenty iterations. A key result is that our framework (dash-dotted curve) is able to converge to an almost equally-high resolution map starting from less-than-ideal 3D projection orientations. Once again, we observe failure when reconstructing with the unrefined set of projection orientations. This further confirms that refinement of the latent variables is vital to achieve a high-quality reconstruction of the map.

E. Simulation of a Real Scenario

We then mimic a real scenario in which the output of the 3D *ab initio* method provided by Relion is used to initialize both the density map and the projection orientations. The in-plane translations are initialized with zeros. We use the same HJC

Authorized licensed use limited to: EPFL LAUSANNE. Downloaded on May 08,2020 at 08:27:07 UTC from IEEE Xplore. Restrictions apply.

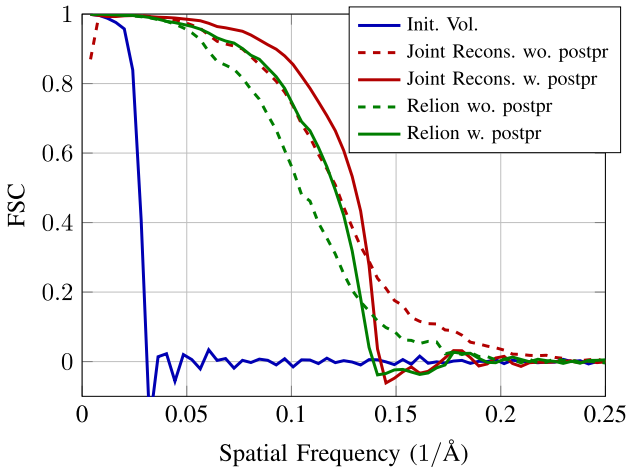

Fig. 10. Comparison between the FSC of the density maps obtained from several baselines and the ground-truth density map. Relion-w-postpr (green solid curve) and Relion-wo-postpr (green dashed curve) refer to the Relion results with and without post-processing, respectively. Note that, Relion-wo-postpr is obtained after averaging the two half-maps. The experimental setups are identical to the ones used in Figure 9.

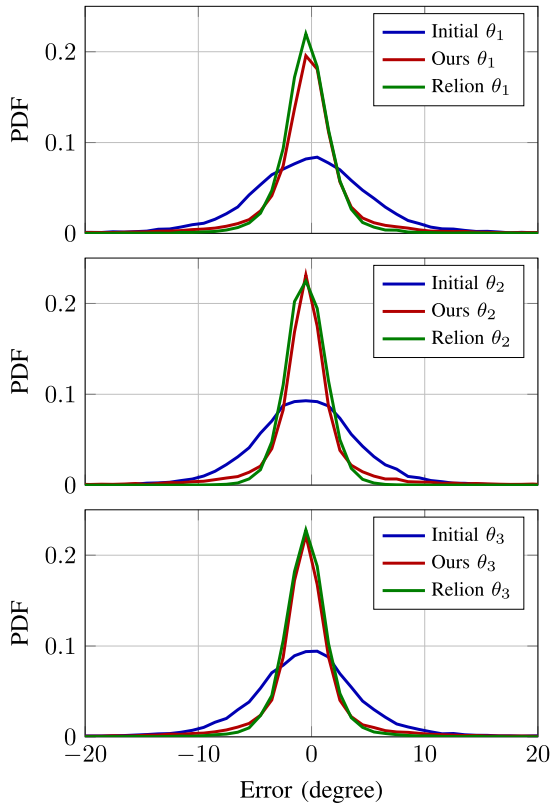

Fig. 11. Probability density function (PDF) of the differences between the true and refined projection orientations by our method $\{\theta_{i,p}^{\text{true}} - \theta_{i,p}^{\text{rec}}\}_{p=1}^{P}$ (red curves), the true and refined projection orientations by Relion $\{\theta_{i,p}^{\text{true}} - \theta_{i,p}^{\text{Relion}}\}_{p=1}^{P}$ (green curves), as well as the true and the initial projection orientations $\{\theta_{i,p}^{\text{true}} - \theta_{i,p}^{\text{init}}\}_{p=1}^{P}$ (blue curves). The experimental setup is identical to Figure 9.

structure to synthesize a volume with size $124 \times 124 \times 124$ and with a voxel size of 2Å. The number of projection images is $20,000$ and the average SNR of the projection images

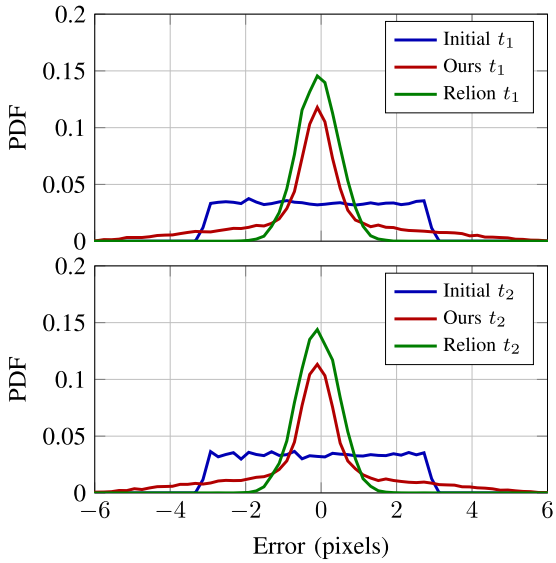

Fig. 12. Probability density function (PDF) of the differences between the true and refined in-plane translations by our method $\{\mathbf{t}_{i,p}^{\text{true}} - \mathbf{t}_{i,p}^{\text{rec}}\}_{p=1}^{P}$ (red curves), the true and refined in-plane translations by Relion $\{\mathbf{t}_{i,p}^{\text{true}} - \mathbf{t}_{i,p}^{\text{Relion}}\}_{p=1}^{P}$ (green curves), as well as the true and the initial in-plane translations $\{\mathbf{t}_{i,p}^{\text{true}} - \mathbf{t}_{i,p}^{\text{init}}\}_{p=1}^{P}$ (blue curves). The experimental setup is identical to Figure 9.

is -14.2dB. Examples of projection images are provided in Fig. 9 (left most column).

We split the projection dataset in two halves and refine each half separately, starting from the same initial volume. Independent refinement of the two halves is a common approach in practice and has two main goals. First, by comparing the two refined half maps against one another, a convergence criterion is obtained. More precisely, we stop the refinement when the FSC between the two half maps fails to improve from one iteration to the next. Second, it reduces overfitting, especially in high-noise regimes.

A visual comparison of the density maps refined by our method and by Relion is presented in Fig. 9 (c)-(d). Both results are post-processed by combining the half-maps and filtering out frequencies beyond the gold-standard FSC by applying a soft tight mask.

A quantitative comparison between our method and Relion based on FSC is provided in Fig. 10. Our method outperforms Relion both with and without post processing.

To assess the quality of the refined latent variables, we compute the PDF of the errors between the refined and the ground-truth projection orientations and in-plane translations. We do this for both our method and Relion, and display the comparisons in Fig. 11 and 12. We observe that the proposed method enjoys comparable performance with Relion for the refinement of the projection orientations and in-plane translations.

Hence, while the proposed angular refinement offers a substantial gain over Relion in the proof-of-concept experiments, the difference here is less significant for a larger volume and a noisier regime. However, our framework could

Authorized licensed use limited to: EPFL LAUSANNE. Downloaded on May 08,2020 at 08:27:07 UTC from IEEE Xplore. Restrictions apply.

be further improved in several ways. An option would be to add regularization for the latent variable estimation. The proposed framework could also be combined with the multi-scale approach proposed in [13] to perform angle refinements at coarser scales, which demonstrates increased robustness to noise. We expect that those extensions would further improve the performance of the method while keeping an attractive numerical complexity as demonstrated in Section III-D. These extensions are to be addressed in future works.

VI. Conclusion

We propose a variational 3D refinement framework for single-particle cryo-electron microscopy that jointly refines the density map and the 3D projection orientations. The refinement of the orientations on the continuum does away with the computationally expensive projection-matching steps. We take alternating steps to update the density map and the latent variables. Steps of the classical method known as the alternating-direction method of multipliers are used to update the density map, while the latent variables are updated through gradient-descent. Our results demonstrate the ability of our framework to refine an approximate map from inaccurate 3D projection orientations. In addition, we show that the resolution of the refined map using our method closely approaches that of the map reconstructed with perfect knowledge of the 3D orientations.

Appendix

A. Fast Reconstruction With ADMM

We use the ADMM scheme proposed in [13]. To that end, we introduce an auxiliary variable $\mathbf{u}$ and constrain its value by setting $\mathbf{u} = \mathbf{L}\mathbf{c}$, as done in (12). The form of the augmented-Lagrangian function used in the ADMM procedure is thus given by

$$\mathcal{L}_{\mathbf{\Theta},\mathbf{\Gamma}}(\mathbf{c},\mathbf{u},\tilde{\mathbf{u}}) = \frac{1}{2}\|\mathbf{g}-\mathbf{H}(\mathbf{\Theta},\mathbf{\Gamma})\,\mathbf{c}\|_2^2 + \lambda\mathcal{R}(\mathbf{u}) + \tilde{\mathbf{u}}^T(\mathbf{L}\mathbf{c}-\mathbf{u}) + \frac{\rho}{2}\|\mathbf{L}\mathbf{c}-\mathbf{u}\|_2^2, \quad (26)$$

where $\tilde{\mathbf{u}}$ is the Lagrangian multiplier that corresponds to the constraint $\mathbf{u} = \mathbf{L}\mathbf{c}$ and ρ is the penalty parameter. Then, the ADMM algorithm alternates between a minimization of $\mathcal{L}$ with respect to $\mathbf{u}$, a minimization of $\mathcal{L}$ with respect to $\mathbf{c}$, and an update of the dual variable $\tilde{\mathbf{u}}$.

The minimization of $\mathcal{L}$ with respect to $\mathbf{u}$ (Step 4 in Algorithm 2) results in

$$\mathbf{u}^{k+1} = \arg\min_{\mathbf{u}} \left(\lambda\mathcal{R}(\mathbf{u}) + \frac{\rho}{2}\|\mathbf{L}\mathbf{c}^k - \mathbf{u} + \tilde{\mathbf{u}}^k/\rho\|_2^2\right), \quad (27)$$

where one recognizes the proximity operator of $\mathcal{R}$. Hence,

$$\mathbf{u}^{k+1} = \mathrm{prox}_{\frac{\lambda}{\rho}\mathcal{R}}(\mathbf{L}\mathbf{c}^k - \tilde{\mathbf{u}}^k/\rho). \quad (28)$$

Then, the objective function involved in the update of $\mathbf{c}$ at Line 2 of Algorithm 2 is

$$\mathcal{L}_{\mathbf{\Theta},\mathbf{\Gamma}}(\mathbf{c},\mathbf{u}^{k+1},\tilde{\mathbf{u}}^k) = \frac{1}{2}\|\mathbf{g}-\mathbf{H}(\mathbf{\Theta},\Gamma)\,\mathbf{c}\|_2^2 + \frac{\rho}{2}\|\mathbf{L}\mathbf{c}-\mathbf{u}^{k+1}+\tilde{\mathbf{u}}^k/\rho\|_2^2, \quad (29)$$

which is a convex quadratic function of $\mathbf{c}$. Its minimization yields the linear system of equations

$$\left(\mathbf{H}^T\mathbf{H}(\mathbf{\Theta},\mathbf{\Gamma}) + \rho\mathbf{L}^T\mathbf{L}\right)\mathbf{c}^{k+1} = \rho\mathbf{L}^T(\mathbf{u}^{k+1}-\tilde{\mathbf{u}}^k/\rho) + \mathbf{H}^T(\mathbf{\Theta},\mathbf{\Gamma})\mathbf{g}. \quad (30)$$

We solve it in terms of $\mathbf{c}$ using a conjugate-gradient method. Note that, for the x-ray operator $\mathbf{H}(\mathbf{\Theta},\mathbf{\Gamma})$, the quantity $\mathbf{H}^T\mathbf{H}(\mathbf{\Theta},\mathbf{\Gamma})\mathbf{c}$ can be efficiently computed at the cost of one FFT and one inverse FFT [13], [15], [16]. Indeed, we have that

$$\mathbf{H}^T\mathbf{H}(\mathbf{\Theta},\mathbf{\Gamma})\mathbf{c} = \mathbf{w}(\mathbf{\Theta}) * \mathbf{c}, \quad (31)$$

where the kernel $\mathbf{w}(\mathbf{\Theta}) \in \mathbb{R}^N$ is given by, $\forall\mathbf{k} \in \Omega_{3\mathrm{D}}$,

$$\left[\mathbf{w}(\mathbf{\Theta})\right]_{\mathbf{k}} = \frac{1}{\det(\mathbf{\Lambda})}\sum_{p=1}^{P}\left(\psi_{\theta_p} * \psi_{\theta_p}^{\vee}\right)(\mathbf{M}_{\theta_p{}^{\perp}}\mathbf{k}), \quad (32)$$

with $\psi_{\theta_p} = h * \mathcal{P}_{\theta_p}(\varphi)$ a function that maps $\mathbb{R}^2$ to $\mathbb{R}$. It is worth to mention that the kernel $\mathbf{w}(\mathbf{\Theta})$ does not depend on the in-plane translations $\mathbf{\Gamma}$.

A similar strategy can be deployed to efficiently evaluate the quantity $\mathbf{H}^T(\mathbf{\Theta},\mathbf{\Gamma})\mathbf{g}$. Let g_p be the continuous version of the measurements $\mathbf{g}_p$ (*i.e.* , $g_p[\mathbf{m}] = g_p(\mathbf{\Lambda}\mathbf{m})$), which can for instance be obtained via some interpolation of the elements of $\mathbf{g}_p$. We then have that [13]

$$\left[\mathbf{H}^T(\mathbf{\Theta},\mathbf{\Gamma})\mathbf{g}\right]_{\mathbf{k}} = \frac{1}{\det(\mathbf{\Lambda})}\sum_{p=1}^{P}\left(g_p * \psi_{\theta_p}^{\vee}\right)(\mathbf{M}_{\theta_p{}^{\perp}}\mathbf{k} + \mathbf{t}_p). \quad (33)$$

The interest of (33) is that $g_p * \psi_{\theta_p}^{\vee}$ can be precomputed on a fine grid using discrete 2D convolutions. Then, each term in the sum (33) comes at the price of an interpolation of this precomputed quantity.

Finally, the Lagrange multiplier $\tilde{\mathbf{u}}$ in ADMM is updated through a simple gradient-ascent step (Step 7 in Algorithm 2).

B. Proof of Theorem 1

Let us expand $\mathcal{J}_p(\theta,\mathbf{t})$ in (15) as

$$\mathcal{J}_p(\theta,\mathbf{t}) = \frac{1}{2}\mathbf{c}^T\mathbf{H}^T\mathbf{H}(\theta,\mathbf{t})\mathbf{c} - \mathbf{c}^T\mathbf{H}^T(\theta,\mathbf{t})\mathbf{g}_p + \frac{1}{2}\|\mathbf{g}_p\|^2 = \frac{1}{2}\mathbf{c}^T(\mathbf{w}(\theta) * \mathbf{c}) - \mathbf{c}^T\mathbf{H}^T(\theta,\mathbf{t})\mathbf{g}_p + \frac{1}{2}\|\mathbf{g}_p\|^2, \quad (34)$$

where $\mathbf{w}(\theta)$ corresponds to one term in the sum (32). Moreover, because φ is isotropic, we have that $\mathcal{P}_{\theta}(\varphi) = \mathcal{P}(\varphi)$, a quantity that does not depend on θ. Hence,

$$\left[\mathbf{w}(\theta)\right]_{\mathbf{k}} = \frac{1}{\det(\mathbf{\Lambda})}\left(\psi * \psi^{\vee}\right)(\mathbf{M}_{\theta^{\perp}}\mathbf{k}), \quad (35)$$

with $\psi = h * \mathcal{P}(\varphi)$.

Then, from (34), one easily sees that, for all $\upsilon \in \{\theta_1, \theta_2, \theta_3, t_1, t_2\}$,

$$\frac{\partial\mathcal{J}_p}{\partial\upsilon}(\theta,\mathbf{t}) = \frac{1}{2}\mathbf{c}^T\left(\mathbf{r}_{\upsilon} * \mathbf{c} - 2\mathbf{q}_{\upsilon}\right), \quad (36)$$

where

$$\mathbf{r}_{\upsilon} = \frac{\partial\mathbf{w}(\theta)}{\partial\upsilon} \text{ and } \mathbf{q}_{\upsilon} = \frac{\partial\mathbf{H}^T(\theta,\mathbf{t})\mathbf{g}_p}{\partial\upsilon}. \quad (37)$$

We now distinguish two cases.

Authorized licensed use limited to: EPFL LAUSANNE. Downloaded on May 08,2020 at 08:27:07 UTC from IEEE Xplore. Restrictions apply.

1) Case $v = \theta_i$ for $i \in \{1, 2, 3\}$: From (35) and the chain rule, we get that

$$r_v[\mathbf{k}] = \frac{1}{\det(\mathbf{\Lambda})} \left(\frac{\partial \mathbf{M}_{\theta^\perp}}{\partial \theta_i} \mathbf{k} \right)^T \nabla \left(\psi * \psi^\vee \right) (\mathbf{M}_{\theta^\perp} \mathbf{k}), \quad (38)$$

where $\frac{\partial \mathbf{M}_{\theta^\perp}}{\partial \theta_i} \in \mathbb{R}^{2\times 3}$ contains the entry-wise derivatives with respect to θ_i of the matrix $\mathbf{M}_{\theta^\perp}$ given in (5). Moreover, from the definition of $\psi : \mathbf{y} \mapsto (h * \mathcal{P}(\varphi))(\mathbf{y})$, with $\mathbf{y} = (y_1, y_2) \in \mathbb{R}^2$, and from the derivation property of the convolution, we have that

$$\nabla \left(\psi * \psi^\vee \right) = \begin{pmatrix} \frac{\partial h}{\partial y_1} * \mathcal{P}(\varphi) * \psi^\vee \\ \frac{\partial h}{\partial y_2} * \mathcal{P}(\varphi) * \psi^\vee \end{pmatrix} = \begin{pmatrix} h * \frac{\partial \mathcal{P}(\varphi)}{\partial y_1} * \psi^\vee \\ h * \frac{\partial \mathcal{P}(\varphi)}{\partial y_2} * \psi^\vee \end{pmatrix}. \quad (39)$$

Note that we could have also differentiated $\psi^\vee$ (instead of h or $\mathcal{P}(\phi)$).

For $\mathbf{q}_v$, we get from (33) that

$$\begin{aligned} q_v[\mathbf{k}] &= \frac{1}{\det(\mathbf{\Lambda})} \frac{\partial (g_p * \psi^\vee)(\mathbf{M}_{\theta^\perp} \mathbf{k} + \mathbf{t})}{\partial \theta_i} \\ &= \frac{1}{\det(\mathbf{\Lambda})} \left(\frac{\partial \mathbf{M}_{\theta^\perp}}{\partial \theta_i} \mathbf{k} \right)^T \nabla (g_p * \psi^\vee)(\mathbf{M}_{\theta^\perp} \mathbf{k} + \mathbf{t}), \quad (40) \end{aligned}$$

where $\nabla \left(g_p * \psi^\vee \right)$ is obtained in the same way as (39), with differentiation on $\psi^\vee$ instead of g_p.

2) Case $v = t_j$ for $j \in \{1, 2\}$: As $\mathbf{w}(\boldsymbol{\theta})$ does not depend on the in-plane translation $\mathbf{t}$, we have that $\mathbf{r}_v = \mathbf{0}_{\mathbb{R}^N}$. For $\mathbf{q}_v$, as in (40), we get that

$$q_v[\mathbf{k}] = \frac{1}{\det(\mathbf{\Lambda})} \frac{\partial (g_p * \psi^\vee)}{\partial y_j} (\mathbf{M}_{\theta^\perp} \mathbf{k} + \mathbf{t}). \quad (41)$$

C. Proof of Proposition 1

The closed-form expression of the x-ray transform of the KBWF φ in (3) is provided in [29] as

$$\mathcal{P}(\varphi)(\mathbf{y}) = \frac{a\sqrt{2\pi/\alpha}}{I_m(\alpha)} \beta_a(\|\mathbf{y}\|)^{m+\frac{1}{2}} I_{m+\frac{1}{2}} \left(\alpha \beta_a(\|\mathbf{y}\|) \right), \quad (42)$$

where $\beta_a(r) = \sqrt{1 - (r/a)^2}$ and I_m is the modified Bessel function of order m. Now, let us introduce the function $f(u) = (\alpha u)^{m+\frac{1}{2}} I_{m+\frac{1}{2}}(\alpha u)$ whose derivative is $f'(u) = \alpha (\alpha u)^{m+\frac{1}{2}} I_{m-\frac{1}{2}}(\alpha u)$. Then, we can write (42) as

$$\mathcal{P}(\varphi)(\mathbf{y}) = \frac{a\sqrt{2\pi/\alpha}}{I_m(\alpha)} \frac{1}{\alpha^{m+\frac{1}{2}}} f(\beta_a(\|\mathbf{y}\|)) \quad (43)$$

and, for all $v \in \{1, 2\}$, obtain that

$$\frac{\partial \mathcal{P}(\varphi)}{\partial y_v}(\mathbf{y}) = \frac{a\sqrt{2\pi/\alpha}}{I_m(\alpha)\alpha^{m+\frac{1}{2}}} \frac{y_v}{\|\mathbf{y}\|} \beta_a'(\|\mathbf{y}\|) f'(\beta_a(\|\mathbf{y}\|)). \quad (44)$$

Finally, the injection of f' and $\beta_a'(r) = \left(-\frac{r}{a^2} \left(1 - (r/a)^2 \right)^{-\frac{1}{2}} \right) = \left(-\frac{r}{a^2 \beta_a(r)} \right)$ into (44) leads to

$$\begin{aligned} \frac{\partial \mathcal{P}(\varphi_{\alpha,a})}{\partial y_v}(\mathbf{y}) &= -\frac{a\sqrt{2\pi/\alpha}}{I_m(\alpha)\alpha^{m+\frac{1}{2}}} \frac{y_v}{\|\mathbf{y}\|} \frac{\|\mathbf{y}\| \alpha (\alpha \beta_a(\|\mathbf{y}\|))^{m+\frac{1}{2}}}{a^2 \beta_a(\|\mathbf{y}\|)} \\ &\quad \times I_{m-\frac{1}{2}}(\alpha \beta_a(\|\mathbf{y}\|)), \\ &= -\frac{\alpha y_v \sqrt{2\pi/\alpha}}{a I_m(\alpha)} \beta_a(\|\mathbf{y}\|)^{m-\frac{1}{2}} \\ &\quad \times I_{m-\frac{1}{2}}(\alpha \beta_a(\|\mathbf{y}\|)), \quad (45) \end{aligned}$$

which completes the proof.

ACKNOWLEDGMENT

The authors would like to thank Dr. M. Nilchian for his constructive feedback on the project.

REFERENCES

[1] S. H. W. Scheres, "RELION: Implementation of a Bayesian approach to cryo-EM structure determination," *J. Struct. Biol.*, vol. 180, no. 3, pp. 519–530, Dec. 2012.

[2] A. Punjani, J. L. Rubinstein, D. J. Fleet, and M. A. Brubaker, "CryoSPARC: Algorithms for rapid unsupervised cryo-EM structure determination," *Nature Methods*, vol. 14, no. 3, pp. 290–296, Mar. 2017.

[3] J. M. de la Rosa-Trevín *et al.*, "Xmipp 3.0: An improved software suite for image processing in electron microscopy," *J. Struct. Biol.*, vol. 184, no. 2, pp. 321–328, Nov. 2013.

[4] N. Grigorieff, "FREALIGN: High-resolution refinement of single particle structures," *J. Struct. Biol.*, vol. 157, no. 1, pp. 117–125, Jan. 2007.

[5] R. Gordon, R. Bender, and G. T. Herman, "Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and X-ray photography," *J. Theor. Biol.*, vol. 29, no. 3, pp. 471–481, Dec. 1970.

[6] P. Gilbert, "Iterative methods for the three-dimensional reconstruction of an object from projections," *J. Theor. Biol.*, vol. 36, no. 1, pp. 105–117, Jul. 1972.

[7] M. Radermacher, "Weighted back-projection methods," in *Electron Tomography*, J. Frank, Ed. New York, NY, USA: Springer, 2006, pp. 245–273.

[8] P. A. Penczek, R. Renka, and H. Schomberg, "Gridding-based direct Fourier inversion of the three-dimensional ray transform," *J. Opt. Soc. Amer. A, Opt. Image Sci.*, vol. 21, no. 4, p. 499, Apr. 2004.

[9] V. Abrishami, J. R. Bilbao-Castro, J. Vargas, R. Marabini, J. M. Carazo, and C. O. S. Sorzano, "A fast iterative convolution weighting approach for gridding-based direct Fourier three-dimensional reconstruction with correction for the contrast transfer function," *Ultramicroscopy*, vol. 157, pp. 79–87, Oct. 2015.

[10] J. M. Carazo, C. O. S. Sorzano, J. Otón, R. Marabini, and J. Vargas, "Three-dimensional reconstruction methods in single particle analysis from transmission electron microscopy data," *Arch. Biochemistry Biophys.*, vol. 581, pp. 39–48, Sep. 2015.

[11] M. Nilchian, C. Vonesch, P. Modregger, M. Stampanoni, and M. Unser, "Iterative FBP for improved reconstruction of X-ray differential phase-contrast tomograms," in *Proc. IEEE 10th Int. Symp. Biomed. Imag.*, San Francisco CA, USA, Apr. 2013, pp. 1248–1251.

[12] N. Jaitly, M. A. Brubaker, J. L. Rubinstein, and R. H. Lilien, "A Bayesian method for 3D macromolecular structure inference using class average images from single particle electron microscopy," *Bioinformatics*, vol. 26, no. 19, pp. 2406–2415, Oct. 2010.

[13] L. Donati, M. Nilchian, C. O. S. Sorzano, and M. Unser, "Fast multiscale reconstruction for cryo-EM," *J. Struct. Biol.*, vol. 204, no. 3, pp. 543–554, Dec. 2018.

[14] C. O. S. Sorzano *et al.*, "A survey of the use of iterative reconstruction algorithms in electron microscopy," *BioMed Res. Int.*, vol. 2017, pp. 1–17, 2017.

[15] A. H. Delaney and Y. Bresler, "A fast and accurate Fourier algorithm for iterative parallel-beam tomography," *IEEE Trans. Image Process.*, vol. 5, no. 5, pp. 740–753, May 1996.

Authorized licensed use limited to: EPFL LAUSANNE. Downloaded on May 08,2020 at 08:27:07 UTC from IEEE Xplore. Restrictions apply.

[16] C. Vonesch, L. Wang, Y. Shkolnisky, and A. Singer, "Fast wavelet-based single-particle reconstruction in cryo-EM," in *Proc. IEEE Int. Symp. Biomed. Imaging, From Nano Macro*, Mar. 2011, pp. 1950–1953.
[17] P. A. Penczek, R. A. Grassucci, and J. Frank, "The ribosome at improved resolution: New techniques for merging and orientation refinement in 3D cryo-electron microscopy of biological particles," *Ultramicroscopy*, vol. 53, no. 3, pp. 251–270, Mar. 1994.
[18] T. S. Baker and R. H. Cheng, "A model-based approach for determining orientations of biological macromolecules imaged by cryoelectron microscopy," *J. Struct. Biol.*, vol. 116, no. 1, pp. 120–130, Jan. 1996.
[19] J. B. Heymann and D. M. Belnap, "Bsoft: Image processing and molecular modeling for electron microscopy," *J. Struct. Biol.*, vol. 157, no. 1, pp. 3–18, Jan. 2007.
[20] F. J. Sigworth, "Principles of cryo-EM single-particle image processing," *Microscopy*, vol. 65, no. 1, pp. 57–67, Feb. 2016.
[21] A. Punjani, M. A. Brubaker, and D. J. Fleet, "Building proteins in a day: Efficient 3D molecular structure estimation with electron cryomicroscopy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 706–718, Apr. 2017.
[22] A. Barnett, L. Greengard, A. Pataki, and M. Spivak, "Rapid solution of the cryo-EM reconstruction problem by frequency marching," *SIAM J. Imag. Sci.*, vol. 10, no. 3, pp. 1170–1195, Jan. 2017.
[23] Y. Michels, E. Baudrier, and L. Mazo, "Radial function based Ab-initio tomographic reconstruction for Cryo electron microscopy," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 1178–1182.
[24] B. Ben Cheikh, É. Baudrier, and G. Frey, "A tomographical reconstruction method from unknown direction projections for 2D gray-level images," *Pattern Recognit. Lett.*, vol. 86, pp. 49–55, Jan. 2017.
[25] S. H. W. Scheres, "A Bayesian view on cryo-EM structure determination," *J. Mol. Biol.*, vol. 415, no. 2, pp. 406–418, Jan. 2012.
[26] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Stat. Soc., Ser. B (Methodol.)*, vol. 39, no. 1, pp. 1–38, 1977.
[27] M. Unser, "Sampling-50 years after Shannon," *Proc. IEEE*, vol. 88, no. 4, pp. 569–587, Apr. 2000.
[28] M. Nilchian, J. P. Ward, C. Vonesch, and M. Unser, "Optimized Kaiser–Bessel window functions for computed tomography," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3826–3833, Nov. 2015.
[29] R. M. Lewitt, "Multidimensional digital image representations using generalized Kaiser–Bessel window functions," *J. Opt. Soc. Amer. A, Opt. Image Sci.*, vol. 7, no. 10, p. 1834, Oct. 1990.
[30] F. Natterer, *The Mathematics of Computerized Tomography*, Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2001.
[31] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D, Nonlinear Phenomena*, vol. 60, nos. 1–4, pp. 259–268, Nov. 1992.
[32] S. Boyd, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2010.
[33] E. Soubies *et al.*, "Pocket guide to solve inverse problems with GlobalBioIm," *Inverse Problems*, vol. 35, no. 10, Oct. 2019, Art. no. 104006.
[34] P. L. Combettes and J.-C. Pesquet, "A proximal decomposition method for solving convex variational inverse problems," *Inverse Problems*, vol. 24, no. 6, Dec. 2008, Art. no. 065014.
[35] L. Armijo, "Minimization of functions having lipschitz continuous first partial derivatives," *Pacific J. Math.*, vol. 16, no. 1, pp. 1–3, Jan. 1966.
[36] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. New York, NY, USA: Springer, 2006.
[37] Z. Yang and P. A. Penczek, "Cryo-EM image alignment based on nonuniform fast Fourier transform," *Ultramicroscopy*, vol. 108, no. 9, pp. 959–969, Aug. 2008.
[38] J. Lee, P. C. Doerschuk, and J. E. Johnson, "Exact reduced-complexity maximum likelihood reconstruction of multiple 3-D objects from unlabeled unoriented 2-D projections and electron microscopy of viruses," *IEEE Trans. Image Process.*, vol. 16, no. 12, pp. 2865–2878, Dec. 2007.
[39] Z. Zhao and A. Singer, "Rotationally invariant image representation for viewing direction classification in cryo-EM," *J. Structural Biol.*, vol. 186, no. 1, pp. 153–166, Apr. 2014.
[40] G. Laxmikanthan *et al.*, "Structure of a holliday junction complex reveals mechanisms governing a highly regulated DNA transaction," *eLife*, vol. 5, May 2016, Art. no. e14313.
[41] X. Qi, P. Schmiege, E. Coutavas, J. Wang, and X. Li, "Structures of human patched and its complex with native palmitoylated sonic hedgehog," *Nature*, vol. 560, no. 7716, pp. 128–132, Aug. 2018.
[42] E. F. Pettersen *et al.*, "UCSF Chimera-A visualization system for exploratory research and analysis," *J. Comput. Chem.*, vol. 25, no. 13, pp. 1605–1612, 2004.
[43] R. Fernandez-Leiro and S. H. W. Scheres, "A pipeline approach to single-particle processing in Relion," *Acta Crystallographica Sect. D, Struct. Biol.*, vol. 73, no. 6, pp. 496–502, 2017.

Authorized licensed use limited to: EPFL LAUSANNE. Downloaded on May 08,2020 at 08:27:07 UTC from IEEE Xplore. Restrictions apply.