

Received 22 December 2023

Accepted 18 June 2024

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

**Keywords:** 3D reconstruction and image processing; computational modeling; cryo-EM; denoising; neural networks.

**Supporting information:** this article has supporting information at [www.iucrj.org](http://www.iucrj.org)

# Refinement of cryo-EM 3D maps with a self-supervised denoising model: *crefDenoiser*

Ishaant Agarwal,<sup>a</sup> Joanna Kaczmar-Michalska,<sup>a,b</sup> Simon F. Nørrelykke<sup>c</sup> and Andrzej J. Rzepiela<sup>a\*</sup>

<sup>a</sup>Scientific Center for Optical and Electron Microscopy, ETH Zürich, 8093 Zürich, Switzerland, <sup>b</sup>Department of Computer Science, Wrocław University of Science and Technology, 50-370 Wrocław, Poland, and <sup>c</sup>Department of Systems Biology, Harvard Medical School, Boston, MA 02115, USA. \*Correspondence e-mail: [andrzejr@ethz.ch](mailto:andrzejr@ethz.ch)

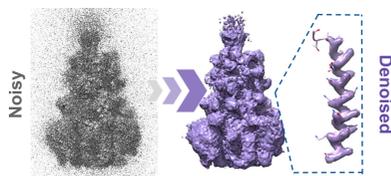
Cryogenic electron microscopy (cryo-EM) is a pivotal technique for imaging macromolecular structures. However, despite extensive processing of large image sets collected in cryo-EM experiments to amplify the signal-to-noise ratio, the reconstructed 3D protein-density maps are often limited in quality due to residual noise, which in turn affects the accuracy of the macromolecular representation. Here, *crefDenoiser* is introduced, a denoising neural network model designed to enhance the signal in 3D cryo-EM maps produced with standard processing pipelines. The *crefDenoiser* model is trained without the need for ‘clean’ ground-truth target maps. Instead, a custom dataset is employed, composed of real noisy protein half-maps sourced from the Electron Microscopy Data Bank repository. Competing with the current state-of-the-art, *crefDenoiser* is designed to optimize for the theoretical noise-free map during self-supervised training. We demonstrate that our model successfully amplifies the signal across a wide variety of protein maps, outperforming a classic map denoiser and following a network-based sharpening model. Without biasing the map, the proposed denoising method leads to improved visibility of protein structural features, including protein domains, secondary structure elements and modest high-resolution feature restoration.

## 1. Introduction

### 1.1. Noise sources and denoising in cryo-EM

Cryogenic electron microscopy (cryo-EM) is one of the leading methods to elucidate protein structures (Bai *et al.*, 2015; Cheng, 2018). In a cryo-EM experiment, a low-intensity electron beam must be used in order to minimize the organic sample degradation during imaging, resulting in noisy images. Thousands of these noisy images are collected and processed to improve the low, significantly below 1 (Frank & Al-Ali, 1975; Egelman, 2016), signal-to-noise ratio (SNR). This approach ultimately allows for modeling detailed atomic resolution 3D protein maps (Nakane *et al.*, 2020). Still, the remaining noise is one of the factors limiting the reconstructed map’s quality (Rosenthal & Henderson, 2003; Frangakis, 2021).

The low-intensity electron beam is responsible for shot noise in the cryo-EM images. Furthermore, the protein particle projections are modulated by structural noise. This noise appears due to the non-uniform surroundings of the imaged particles: for example, amorphous ice impurities and ice thickness fluctuations. The resulting 3D protein-density maps are also affected by errors in data processing: for instance, inaccuracies of 3D image alignment (Jiménez-Moreno *et al.*, 2021).



Published under a CC BY 4.0 licence

The most successful cryo-EM denoising method so far is simply processing and averaging a large number of images. The improvement of the reconstructed map as a function of the number of acquired images can be estimated (Rosenthal & Henderson, 2003). The relation is logarithmic, which sets improvement limits due to acquisition costs. Other methods, such as 2D micrograph denoising (Palovcak *et al.*, 2020; Bepler *et al.*, 2020), 3D map denoising (Ramlaul *et al.*, 2019; Tegunov *et al.*, 2021) and corrections for 3D image alignment (Jiménez-Moreno *et al.*, 2021), are an active area of development and testing.

## 1.2. Deep learning enhances cryo-EM data processing

Deep learning has found extensive use in image processing. Therefore, the adoption of neural network models in cryo-EM processing is broad and the application of new methods has often been straightforward (Chung *et al.*, 2022). For example, the popular YOLO object detection network (Redmon *et al.*, 2016) was adapted to pick protein particles from EM images as *crYOLO* (Wagner *et al.*, 2019). Denoising network models, developed for general-purpose image denoising and restoration, can also be used for contrast enhancement in 2D EM images (Bepler *et al.*, 2020; Lehtinen *et al.*, 2018; Batson & Royer, 2019). There are also a number of specialized methods for 3D model building [e.g. *CryoDRGN* (Zhong *et al.*, 2021), *3DFlex* (Punjani & Fleet, 2023), GMM-based methods (Chen *et al.*, 2024)] map post-processing [*DeepEMhancer* (Sanchez-Garcia *et al.*, 2021), *EMReady* (He *et al.*, 2023)], map analysis [*DeepRes* (Ramírez-Aportela *et al.*, 2019)] and atomistic model building [*Emap2sec* (Maddhuri Venkata Subramaniya *et al.*, 2019), *ModelAngelo* (Jamali *et al.*, 2022)], which are powered by neural networks.

## 1.3. 3D map sharpening and denoising

Cryo-EM 3D density maps show a loss of contrast at high resolutions. This is caused by the decay of high-frequency signal amplitudes, which are smaller than expected when compared with the reference X-ray scattering data (Rosenthal & Henderson, 2003). Contrast degradation is caused by imperfect imaging due to inherent instrument limitations in transmission electron microscopy (TEM) apparatus, including specimen movement and charging, radiation damage, inelastic electron scattering events, partial microscope coherence, and particle flexibility and heterogeneity, and also due to the limitations of data-processing methods (Henderson, 1992; Rosenthal & Henderson, 2003). To restore the degraded signal, global sharpening methods and, more recently, local sharpening methods have been developed. *LocScale* (Jakobi *et al.*, 2017) uses an atomic reference structure to locally correct signal amplitudes. *LocalDeblur* (Ramírez-Aportela *et al.*, 2020) performs deblurring based on the local resolution estimation. *LocSpiral* (Kaur *et al.*, 2021) uses the spiral phase transform to enhance high-resolution map features. Vargas *et al.* (2022) utilize a multiscale tubular filter to enhance post-processed maps. *DeepEMhancer* (Sanchez-Garcia *et al.*, 2021) is a network model trained on pairs of raw experimental and

sharpened maps. It uses *LocScale* (Jakobi *et al.*, 2017) maps as targets to mimic *LocScale*'s local sharpening effect without the need for atomic reference structures. *EMReady* is a similar method (He *et al.*, 2023), but trained on pairs of raw experimental maps and maps simulated from atomistic models. *EMReady* is optimized to match the ground-truth final post-processed maps but not necessarily to represent the raw experimental data optimally. Another widely used sharpening method is *phenix.auto\_sharpen* (Terwilliger *et al.*, 2018), which has also been employed in this article to visually compare maps (see Section 2.6). Another relevant method, *LAFTER* (Ramlaul *et al.*, 2019), is a classic local 3D map denoising algorithm based on two serial filters. *LAFTER* operates in both real and Fourier space. It compares independent half-set reconstructions to identify and retain shared features with power greater than the noise. *LAFTER* does not sharpen EM maps, it only denoises them, which makes it a suitable reference method for benchmarking our network-based denoising model.

## 1.4. Contributions

Entries in the Electron Microscopy Data Bank (EMDB) repository contain not only the final processed cryo-EM 3D maps but also, in many cases, 'half maps', which result from processing two randomly divided half-datasets. The two half-maps are used for the determination of the map resolution (Van Hell & Schatz, 2005; Rosenthal & Henderson, 2003), and can be used to illustrate how the map's SNR changes as a function of the signal frequency with Fourier shell correlation plots [FSC (Van Heel, 1987)] or by directly calculating power spectra of signal and noise components (Palovcak *et al.*, 2020). This type of data is suitable for training a neural network 3D map denoising model. The most natural setup would be the so-called noise-to-noise model (Lehtinen *et al.*, 2018), in which the first half-map is used as a denoising template and the second serves to calculate loss during the supervised model training. This is how the *M* software (Tegunov *et al.*, 2021) is, on the fly, training a map-specific model during a map refinement procedure. Here, we take advantage of existing theoretical analysis to further enhance the model's denoising power. Rosenthal & Henderson (2003) derive a relation between an ideal noise-free 3D map and a pair of two noisy half-maps as a function of FSC. In *Methods*, we outline how we employ this relation to optimize the denoising network in self-supervised training. We compare our model *crefDenoisier* with the recent 3D map denoiser *LAFTER* (Ramlaul *et al.*, 2019), the sharpening model *EMReady* (He *et al.*, 2023) and the pre-trained 3D *Topaz* denoising model (Bepler *et al.*, 2020; specific for cryo-electron tomography data, *Topaz<sup>Tom</sup>*), and analyze their denoising performance on the test maps set with a number of selected characteristics. Furthermore, we analyze the signal-to-noise enhancements as a function of signal frequency, and we show that *crefDenoisier* improves the SNR, without introducing large biases in the denoised maps. This is in contrast to the *EMReady* model, whose primary role is to

enhance maps by ingesting additional signal to the maps, rather than filtering the noise.

Finally, we provide examples of denoising with selected maps, where our processing provides insights into the usability and advantages of denoising.

## 2. Methods

### 2.1. Model optimization

The *crefDenoiser* model is trained using a loss function based on FSC and the statistical measure known as  $C_{\text{ref}}$ . FSC score is the most popular metric used in cryo-EM imaging to determine image and map quality (Rosenthal & Henderson, 2003). It measures the normalized cross-correlation between two volumes over corresponding shells in the Fourier domain. It quantifies the similarity of signals between two maps (or images when a 2D signal is analyzed) as a function of frequency. The FSC value between two map volumes is given by

$$\text{FSC}(s) = \frac{\sum_{r \in s} F_1(r) \cdot F_2^*(r)}{[\sum_{r \in s} |F_1(r)|^2 \cdot \sum_{r \in s} |F_2(r)|^2]^{1/2}}, \quad (1)$$

where  $F_1$  and  $F_2^*$  represent the Fourier transform and conjugate Fourier transform of the two volumes, and  $s$  is the shell being considered. The summation is performed over all frequency voxels  $r$  contained in the shell  $s$ . To calculate a scalar score, we integrate the FSC curve over all frequency shells up to the Nyquist frequency:

$$\text{FSC}_{\text{scalar}} = \int_0^f \text{FSC}(s) ds. \quad (2)$$

FSC values can range from +1 for perfectly correlated images to 0 for completely uncorrelated images. Negative values (up to  $-1$ ) imply a negative correlation. An FSC of  $-1$  would represent identical images with opposite contrasts (Penczek, 2020). In cryo-EM imaging, the  $\text{FSC}_{\text{half}}$  curve between half-dataset maps is used to determine the ‘gold standard’ resolution (Rosenthal & Henderson, 2003). The frequency at which the FSC curve first falls below a fixed value (usually 0.143) (Rosenthal & Henderson, 2003) is used as a resolution estimate.

$F_1$  and  $F_2$  in equation (1) can be represented by a common signal term and an additional noise term,  $F_1 = S + N_1$ ,  $F_2 = S + N_2$ , where  $N_1$  and  $N_2$  are realizations of noise  $N$ . With this,  $\text{FSC}_{\text{half}}$  becomes (Rosenthal & Henderson, 2003)

$$\text{FSC}_{\text{half}}(s) = \frac{\sum (S + N_1) \cdot (S + N_2)^*}{(\sum |S + N_1|^2 \sum |S + N_2|^2)^{1/2}} \approx \frac{\sum |S|^2}{\sum |S^2 + N^2|}, \quad (3)$$

when signal and noise are uncorrelated and data in the half-sets are on the same scale. Using the above notation, we can also write FSC between an ideal map and a map reconstructed from a complete dataset. The ideal map has no noise term, and the noise of the full-dataset map becomes  $N/\sqrt{2}$  when

compared with the half-dataset noise  $N$ . This so-called  $C_{\text{ref}}$  can be expressed as a function of  $\text{FSC}_{\text{half}}$  [substituting with the result of equation (3), see also Rosenthal & Henderson (2003)]:

$$C_{\text{ref}}(s) = \frac{\sum \left( S + \frac{N}{\sqrt{2}} \right) \cdot S^*}{\left( \sum \left| S + \frac{N}{\sqrt{2}} \right|^2 \sum |S|^2 \right)^{1/2}} \approx \left[ \frac{2 \times \text{FSC}_{\text{half}}(s)}{1 + \text{FSC}_{\text{half}}(s)} \right]^{1/2}. \quad (4)$$

Signal and noise in the above equations are uncorrelated only in a statistical sense (*i.e.* the expectation value of  $S \cdot N = 0$ ), and for a given map, noise realization might have a non-zero correlation with the signal (van Heel & Schatz, 2017, 2005). Furthermore, equation (4) has many solutions, in the sense that a noise-free map that fulfills equation (4) is not unique (Ramlaul *et al.*, 2019).

Our loss function is the mean absolute difference between the  $C_{\text{ref}}$  in equation (4) (calculated using readily available  $\text{FSC}_{\text{half}}$  curves) and  $\text{FSC}_{\text{FD}}$ , which is calculated between the average of half-maps (used as the network input map for denoising) and the denoised output map:

$$\mathcal{L}_C = \frac{1}{f} \sum_{s=0}^f |C_{\text{ref}}(s) - \text{FSC}_{\text{FD}}(s)|. \quad (5)$$

Calculating the loss, we assume that the average of two half-maps represents a map reconstructed from a complete dataset.

The  $\text{FSC}_{\text{FD}}$  between a noise-free map and the average of two half-set maps should completely overlap with  $C_{\text{ref}}$ . An  $\text{FSC}_{\text{FD}}$  value above  $C_{\text{ref}}$  indicates that the denoised map still contains some residual noise (under-denoised), while a value below  $C_{\text{ref}}$  points to a loss of signal. The lower the  $\mathcal{L}_C$ , the closer the  $\text{FSC}_{\text{FD}}$  of our network output is to the  $C_{\text{ref}}$ , indicating a more effective denoising operation. The loss function  $\mathcal{L}_C$  is differentiable since it is directly derived from the FSC function, which itself is differentiable (Kaczmar-Michalska *et al.*, 2022) and can thus be readily applied in gradient-based model training. The  $\mathcal{L}_C$  loss allows us to perform Fourier space based model optimization for the real-space theoretical noise-free map, even without actually having noise-free maps to drive the model training. The correlations of signal and noise realizations in the training maps should not limit the loss performance since the training is performed over many maps, and these correlations should average out.

### 2.2. Bias analysis

The denoising process might introduce a spurious bias signal to the map (Palovcak *et al.*, 2020). For example, the self-supervised *Noise2Void* model introduces checkerboard artefacts to the denoised images (Höck *et al.*, 2023). In general, bias manifestation can be of any form and can potentially harm the denoised map quality. Can the magnitude of bias be quantified and used to assess denoising model quality?

A noisy map consists of signal and noise,

$$M = S + N, \quad (6)$$

and a denoised map consists of signal, bias and some leftover noise,

$$D = S + B + N^d. \quad (7)$$

A variance of signal  $[\text{var}(S)]$ , noise  $[\text{var}(N)]$ , bias  $[\text{var}(B)]$  and leftover noise after denoising  $[\text{var}(N^d)]$  can be calculated from the noisy and denoised half-maps. We follow derivations provided by Palovcak *et al.* (2020) to show that these properties are readily calculable starting from covariances of noisy and denoised maps. Elementary relations between variance and covariance of variables are used in the calculations and are provided below. With variables  $X, Y, V$  and  $Z$ :

$$\text{var}(X) = \text{cov}(X, X), \quad (8)$$

$$\begin{aligned} \text{cov}(X, Y) &= E\{[X - E(X)][Y - E(Y)]\} \\ &= E(XY) - E(X)E(Y) \end{aligned} \quad (9)$$

and

$$\begin{aligned} \text{cov}(X + V, Y + Z) &= \text{cov}(X, Y) + \text{cov}(V, Y) \\ &\quad + \text{cov}(X, Z) + \text{cov}(V, Z), \end{aligned} \quad (10)$$

where  $\text{cov}(\cdot, \cdot)$  is the covariance and  $E(\cdot)$  is the expectation value. Equations (8) and (10) imply that

$$\text{var}(X + Y) = \text{var}(X) + \text{var}(Y) + 2\text{cov}(X, Y), \quad (11)$$

and if  $X$  and  $Y$  are independent,  $E(XY) = E(X)E(Y)$ , equation (9) implies that  $\text{cov}(X, Y) = 0$ . Assuming that  $S$  and  $N$  are independent, the variance of signal and noise in 3D cryo-EM maps,  $\text{var}(S)$  and  $\text{var}(N)$ , can be calculated using the noisy half-maps  $M_1$  and  $M_2$ :

$$\begin{aligned} \text{cov}(M_1, M_2) &= \text{cov}(S + N_1, S + N_2) \\ &= \text{cov}(S, S) + \text{cov}(S, N_2) + \text{cov}(N_1, S) \\ &\quad + \text{cov}(N_1, N_2) \\ &\approx \text{cov}(S, S) = \text{var}(S) \end{aligned} \quad (12)$$

and

$$\begin{aligned} &[\text{cov}(M_1, M_1)\text{cov}(M_2, M_2)]^{1/2} - \text{cov}(M_1, M_2) \\ &= [\text{cov}(S + N_1, S + N_1)\text{cov}(S + N_2, S + N_2)]^{1/2} \\ &\quad - \text{cov}(S + N_1, S + N_2) \\ &= \left\{ [\text{var}(S) + \text{var}(N_1) + 2\text{cov}(S, N_1)] \right. \\ &\quad \cdot [\text{var}(S) + \text{var}(N_2) + 2\text{cov}(S, N_2)] \left. \right\}^{1/2} - \text{var}(S) \\ &\quad - \text{cov}(S, N_2) - \text{cov}(N_1, S) - \text{cov}(N_1, N_2) \\ &\approx \left\{ [\text{var}(S) + \text{var}(N)]^2 \right\}^{1/2} - \text{var}(S) = \text{var}(N). \end{aligned} \quad (13)$$

Furthermore, assuming that  $B$  and  $N$ ,  $B$  and  $N^d$ , and  $S$  and  $N^d$  are independent, we can calculate the variance of  $B$  using the noisy  $(M_1, M_2)$  and denoised  $(D_1, D_2)$  half-maps:

$$\begin{aligned} &\text{cov}(D_1, D_2) + [\text{cov}(M_1, M_1)\text{cov}(M_2, M_2)]^{1/2} \\ &\quad - 2\text{cov}(M_1, D_2) - \text{var}(N) \\ &= \text{cov}(S + B + N_1^d, S + B + N_2^d) + [\text{cov}(S + N_1, S + N_1) \\ &\quad \cdot \text{cov}(S + N_2, S + N_2)]^{1/2} - 2\text{cov}(S + N_1, S + B + N_2^d) \\ &\quad - \text{var}(N) \\ &= \text{var}(S) + \text{var}(B) + 2\text{cov}(S, B) + \text{cov}(B, N_2^d) + \text{cov}(B, N_1^d) \\ &\quad + \text{cov}(S, N_2^d) + \text{cov}(S, N_1^d) + \text{cov}(N_1^d, N_2^d) \\ &\quad + \left\{ [\text{var}(S) + \text{var}(N_1) + 2\text{cov}(S, N_1)] \right. \\ &\quad \cdot [\text{var}(S) + \text{var}(N_2) + 2\text{cov}(S, N_2)] \left. \right\}^{1/2} \\ &\quad - 2[\text{var}(S) + \text{cov}(N_1, N_2^d) + \text{cov}(S, B) + \text{cov}(B, N_1) \\ &\quad + \text{cov}(S, N_2^d)] - \text{var}(N) \\ &\approx \text{var}(B). \end{aligned} \quad (14)$$

A similar derivation for  $N^d$  is provided below:

$$\begin{aligned} &[\text{cov}(D_1, D_1)\text{cov}(D_2, D_2)]^{1/2} - \text{cov}(D_1, D_2) \\ &= [\text{cov}(S + B + N_1^d, S + B + N_1^d) \\ &\quad \cdot \text{cov}(S + B + N_2^d, S + B + N_2^d)]^{1/2} \\ &\quad - \text{cov}(S + B + N_1^d, S + B + N_2^d) \\ &= [\text{cov}(S, S) + \text{cov}(B, B) + \text{cov}(N_1^d, N_1^d) + 2\text{cov}(S, B) \\ &\quad + 2\text{cov}(B, N_1^d) + 2\text{cov}(S, N_1^d)]^{1/2} \cdot [\text{cov}(S, S) + \text{cov}(B, B) \\ &\quad + \text{cov}(N_2^d, N_2^d) + 2\text{cov}(S, B) + 2\text{cov}(B, N_2^d) \\ &\quad + 2\text{cov}(S, N_2^d)]^{1/2} - \text{var}(S) - \text{var}(B) - 2\text{cov}(S, B) \\ &\quad - \text{cov}(N_1^d, N_2^d) - \text{cov}(B, N_2^d) - \text{cov}(B, N_1^d) \\ &\quad - \text{cov}(S, N_2^d) - \text{cov}(S, N_1^d) \\ &\approx \text{var}(N^d). \end{aligned} \quad (15)$$

The covariances can be calculated separately for each frequency shell  $s$  in 3D maps:

$$\text{cov}[F_1(s), F_2(s)] = \frac{1}{n} \sum_{r \in s} F_1(r) \cdot F_2^*(r), \quad (16)$$

where  $n$  is the number of voxels  $r$  in the shell  $s$ . Usefully, the *Electron Microscopy Data Analytical Toolkit (EMDA)* (Warshamanage *et al.*, 2022) provides routines to calculate 3D map covariances.

### 2.3. Network architecture

We use a 3D U-Net-like (Ronneberger *et al.*, 2015) model with five levels of depth in the contracting path and corresponding five levels in the expanding path. The overall architecture of *crefDenoiser* is enumerated below:

(i) Input layer. The model takes as input a 3D image with a single channel.

(ii) Contracting path. The contracting path consists of five blocks, each containing a 3D convolutional layer with 16 filters, a Leaky ReLU activation function and a 3D MaxPooling layer.

(iii) Bottleneck. The bottleneck consists of a 3D convolutional layer and a Leaky ReLU activation function. This part of the network is responsible for learning the most abstract features of the input data.

(iv) Expanding path. The expanding path also consists of five blocks, each containing a 3D UpSampling layer, a concatenation operation, two 3D convolutional layers and two Leaky ReLU activation functions. The concatenation operation combines the features learned in the contracting path with the upsampled output, allowing the network to use both local and global features for the reconstruction of the denoised image.

(v) Output layer. The final layer of the network is a 3D convolutional layer with a single filter, which outputs the denoised 3D image.

The total number of parameters in the model is 322 881, all of which are trainable.

Since cryo-EM images inherently capture the intricate 3D structures of macromolecules, this architecture is particularly well suited to the task due to its ability to effectively learn spatial hierarchies and extract features from the 3D data. Although the network is trained on patches of maps (see Section 2.4), it is fully convolutional and can denoise whole maps of any size without any architectural restrictions.

## 2.4. Data preparation

Our model was trained on data collected from the EMDB repository. All cryo-EM entries, with an associated mask and two half-maps attached, were downloaded from the online EMDB FTP server. Any entries with size mismatches between the two half-maps and/or the mask files were pruned. The remaining 3710 records, with resolutions in the range 1.22–9.9 Å, were used in constructing the training and test datasets.

All the half-map pairs were first independently masked and standardized to have a mean voxel value of 0 and an intensity standard deviation of 1. They were then randomly shuffled (as a pair) and split into patches of size  $96 \times 96 \times 96$ . Any patches that lay completely outside the masking region were removed. Those remaining were then divided in a 1:9 ratio to construct the test and training datasets. The training set finally contains 55 176 such pairs of half-map patches from 3386 maps, while the test dataset contains 6126 pairs from 324 maps. The model was not tuned on validation data, and we do not distinguish between the validation set and the test set. Trained model performance analysis was performed on 50 random maps selected from the test dataset.

## 2.5. Training process

The training process was conducted using the *Adam* (Kingma & Ba, 2014) optimizer ( $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ ,  $\epsilon = 10^{-8}$ ) with an initial learning rate of 0.0003. The learning rate was reduced exponentially with a decay rate of  $k = 0.7$  every ten epochs. The model was trained for 195 epochs with a batch size of 6 on three NVIDIA A100 80GB GPUs. The training time was  $\sim 120$  h. After each epoch, the model's performance was evaluated on the validation set, and the model weights were

saved. The training convergence analysis is shown in Fig. S2 of the supporting information.

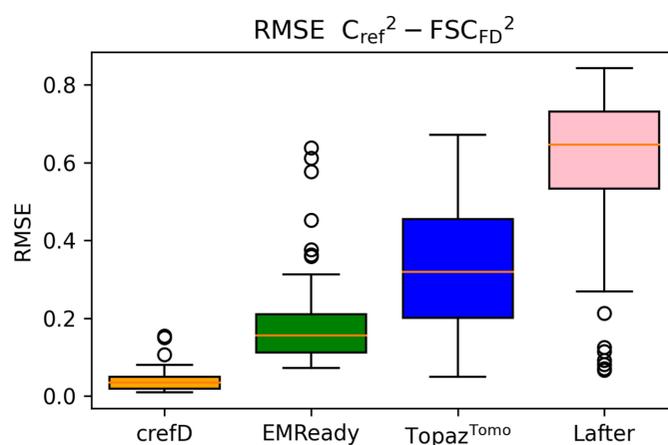
## 2.6. Map sharpening

$C_{\text{ref}}$ -denoised maps may be further sharpened. Here, the selected denoised maps were sharpened only to facilitate graphical comparison with the published maps, and were not used for any quantitative analysis. Local sharpening with *Phenix* software (Adams *et al.*, 2010), with the resolution threshold set to be slightly lower than the published resolution ( $-0.5$  Å), was chosen. The sharpening method was automatically selected using *phenix.auto\_sharpen* (Terwilliger *et al.*, 2018).

## 3. Results

### 3.1. Comparison with *EMReady*, *LAFTER* and *Topaz<sup>Tomo</sup>* methods

For a random set of masked denoised test maps ( $n = 50$ ), we calculated  $\text{FSC}_{\text{FD}}$  and compared it with the theoretical  $C_{\text{ref}}$ , analyzing root mean square difference between the two curves. Fig. 1 shows results for denoising with *crefDenoiser*, *LAFTER* and *EMReady*, as well as the *Topaz<sup>Tomo</sup>* 3D denoiser (Bepler *et al.*, 2020). As is evident from the plot, *crefDenoiser* has the smallest value (is the closest to  $C_{\text{ref}}$ ) of all the methods by a significant margin, which is not surprising since *crefDenoiser* was trained to minimize this map property. *EMReady* performs second best, while *Topaz<sup>Tomo</sup>* and *LAFTER* demonstrate lower performance. In Fig. S1,  $\text{FSC}_{\text{FD}}$  and  $C_{\text{ref}}$  for one representative map, EMD-23276 (Zhang *et al.*, 2021), are shown.  $\text{FSC}_{\text{FD}}$  for *crefDenoiser* follows closely the theoretical  $C_{\text{ref}}$  curve. The pattern visible for this map repeats in other test maps: *EMReady* shows deviations in



**Figure 1** Analysis of denoising performance for a test set of cryo-EM maps. Root mean square error (RMSE) between  $\text{FSC}_{\text{FD}}^2$  and  $C_{\text{ref}}^2$  curves is shown for 50 test masked maps processed using *crefDenoiser*, *EMReady*, *Topaz<sup>Tomo</sup>* and *LAFTER* as box and whisker plots. The solid central line depicts the median and the boxes represent the interquartile range. The whiskers span the distribution, excluding any outliers denoted by circles. RMSE is calculated for the squared values to ensure the numerical stability of calculations.

FSC<sub>FD</sub> from  $C_{ref}$  in lower frequencies (possibly due to the low accuracy of denoising lipids and other molecules not present in the atomistic structures used to construct reference maps for the *EMReady* model training), *LAFTER* shows rather large (as in Fig. S1) or rather low (see outliers in Fig. 1) RMSE to the  $C_{ref}$  curve, and *Topaz<sup>Tomo</sup>* under-denoises the high-frequency signal. The RMSE results of the FSC<sub>FD</sub> to  $C_{ref}$  curves alone are not enough to claim that the maps are close to the true biological reconstruction, since various denoised maps can minimize the RMSE measure. Further analysis of the denoised map properties is provided below.

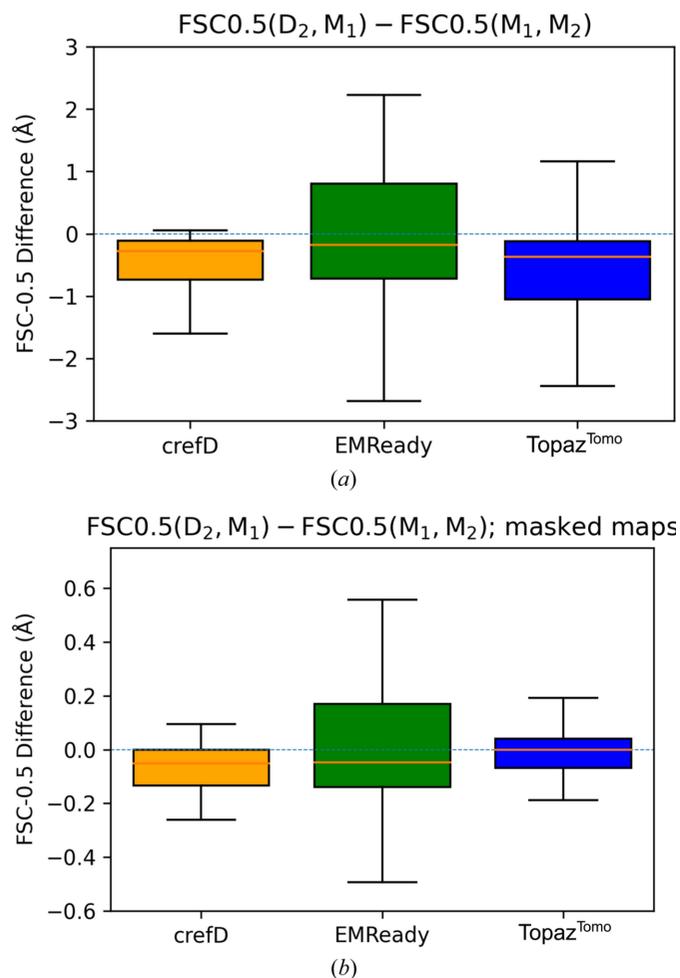
*EMReady* (He *et al.*, 2023) authors use the resolution at which the FSC between a pair of maps falls to one-half (*i.e.* FSC-0.5) as a metric for the model evaluation. Here, we compare the performance of *crefDenoisier* with the other denoisers using this same metric.

For this we calculated FSC-0.5 values for the noisy [FSC-0.5( $M_1, M_2$ )] and denoised-noisy half-map pairs [FSC-

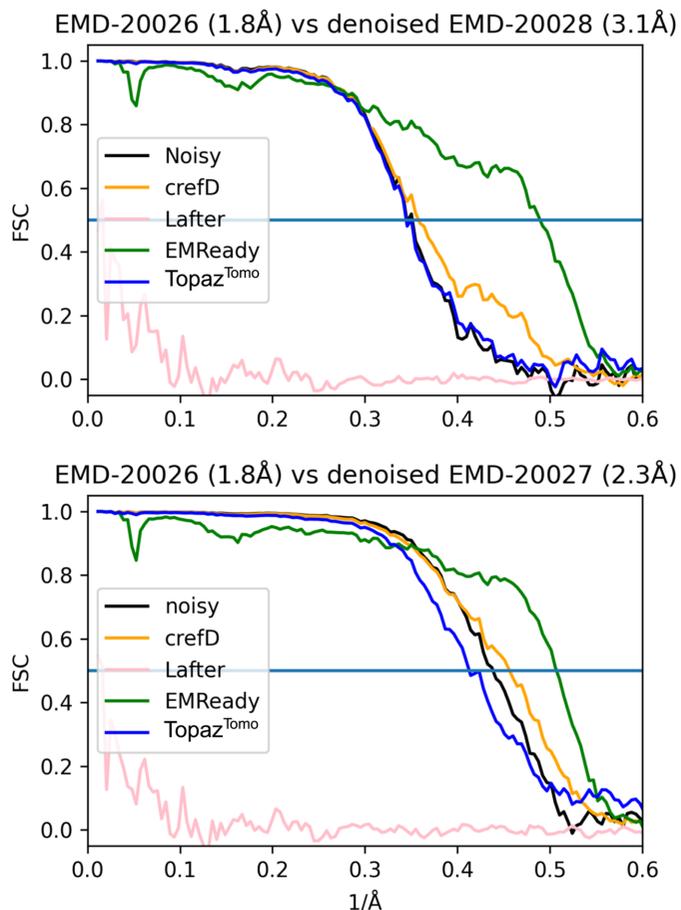
0.5( $D_2, M_1$ )] for the 50 test maps. *LAFTER* is not designed to denoise single half-maps, so it was excluded from this analysis. In Fig. 2, we present the change of FSC-0.5 after applying *crefDenoisier*, *EMReady* and *Topaz<sup>Tomo</sup>* models for masked and non-masked maps.

For the majority of the analyzed maps, the FSC-0.5 change is negative for all three methods when the analyzed maps are unmasked (with a median change around  $-0.2$  Å). A negative change suggests improved quality of the processed maps since the FSC-0.5 shifts to higher resolution values. When masked maps are used as inputs, *crefDenoisier* and *EMReady* facilitate a negative change, albeit smaller (with a median change of less than  $-0.1$  Å). *Topaz<sup>Tomo</sup>* does not manage to significantly affect the FSC-0.5 with the masked inputs.

The presented analysis suggests that the *crefDenoisier* model can perform half-map denoising, even though it was trained on the full-data density maps. We reason that the noisy maps from the large training set have a broad range of noise levels, and most of the half-maps fall within that range.

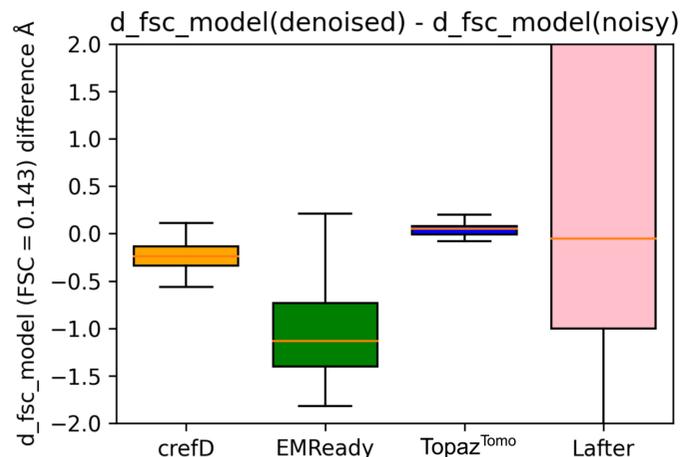


**Figure 2**  
Comparison of FSC-0.5 for maps denoised with the *crefDenoisier*, *EMReady* and *Topaz<sup>Tomo</sup>* methods. The difference in FSC-0.5 values between denoised-noisy half-maps and noisy-to-noisy half-maps ( $M_1, M_2$ ) for 50 EMDB test entries is shown as box-whisker plots for (a) non-masked and (b) masked maps. The plots depict distributions of FSC-0.5 difference for each method. The solid central line depicts the median and the boxes represent the interquartile range.



**Figure 3**  
Comparison of denoised maps with a higher-resolution map. Half-maps for human apoferritin EMD entries EMD-20027 (2.3 Å) and EMD-20028 (3.1 Å) were averaged (within entry), and the mean maps were subsequently denoised. Next, FSC curves to the mean map of the high-resolution entry, EMD-20026 (1.8 Å), were calculated. Maps EMD-20026, EMD-20027 and EMD-20028 were obtained by processing different fractions of the same single-particle dataset (Pintilie *et al.*, 2020). Denoising with *EMReady* and *crefDenoisier* improves FSC curves to the higher-resolution map for masked maps. The maps are part of the test set.

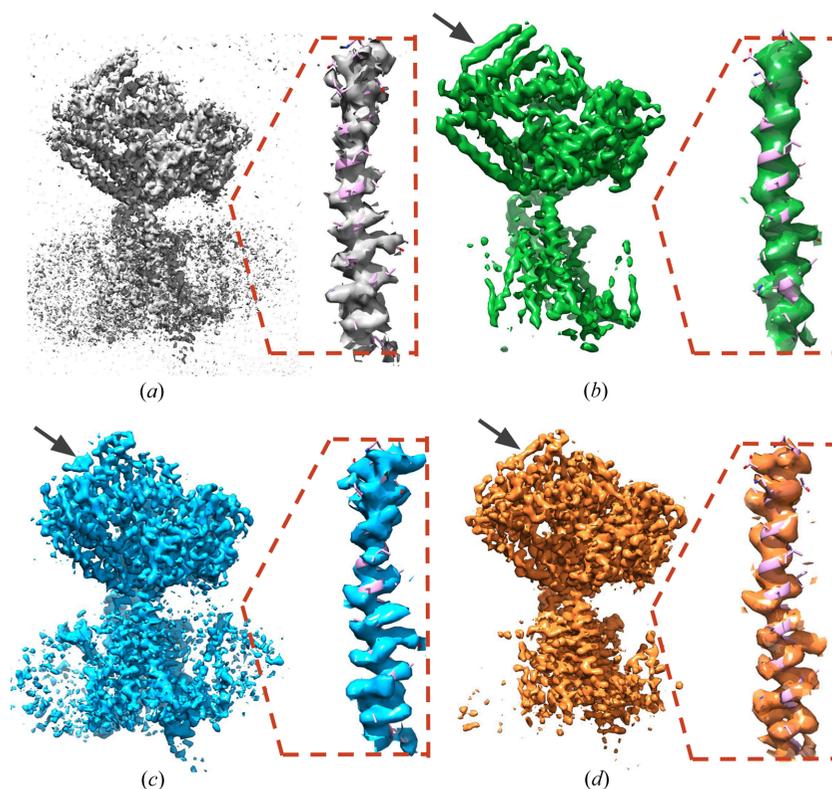
Next, we tested methods by comparing lower-resolution denoised maps with a high-resolution map. The analyzed apoferritin entries EMD-20026 (1.8 Å), EMD-20027 (2.3 Å) and EMD-20028 (3.1 Å) are reconstructed from the same dataset using different fractions of the acquired single-particle



**Figure 4**  
A comparison of the fit of denoised and noisy test-set maps with their published atomic models. The 33 test maps (from 50) for which automatic model extraction was successful were used in the analysis. The masked FSC = 0.143 values were directly computed using *phenix.mtriage* (Adams *et al.*, 2010).

images (Pintilie *et al.*, 2020). In Fig. 3, we show that denoising with *EMReady* and *crefDenoiser* improves FSC curves for both lower-resolution models (on masked maps). Mainly, the high-frequency part of the FSC curve is changed; the effect is significantly more pronounced for the *EMReady* model. *Topaz<sup>Tomo</sup>* gives little to no improvement for EMD-20028 and actually degrades map quality for EMD-20027 over most of the frequency range. *LAFTER* performs poorly for both maps. We further analyze whether *crefDenoiser* introduces any spurious densities in apoferritin maps in Fig. S3.

Unfortunately, the analysis for low-resolution denoised and high-resolution benchmark maps cannot be performed on a larger number of maps due to a lack of accessible data. However, we can approximate this analysis by comparing experimental cryo-EM maps with the maps calculated from atomic models. In Fig. 4 we demonstrate the difference in fit between 33 full maps from the test set (for which we extracted atomic models in a programmatic way) and their corresponding atomic models, before and after denoising. The fit is evaluated using the *d\_fsc\_model\_0143* metric from *phenix.mtriage* (Adams *et al.*, 2010). This measure signifies the resolution cutoff at which the FSC between the EM map and the atomic model falls below 0.143. We observe that denoising using *crefDenoiser* generates a modest improvement in the resolution cutoff (with a median change around  $-0.3$  Å) while *EMReady* maps demonstrate large similarity with the model



**Figure 5**  
Denoising a medium-resolution map. The map of Sec61 membrane channel from *Saccharomyces cerevisiae* (Itskanov *et al.*, 2021) and its fragment focused on one selected internal channel helix with the fitted atomistic model are visualized. This map is part of the test set and has a resolution of 4 Å. The noisy map, constructed as a mean of two half-maps, is presented in (a). The final published map is shown in (b). The noisy sharpened mean map is presented in (c). The denoised and sharpened mean map is presented in (d). The denoised map preserves lower-resolution motifs (the black-arrow marked  $\alpha$ -helix) and high-resolution details (the inset), while the noise is substantially reduced. Contouring was tuned to make the maps most similar.

maps (with a median change larger than  $-1 \text{ \AA}$ ). This result is not unexpected since *EMReady* was directly trained using maps simulated from the reference atomic models. On the other hand, *crefDenoisier* was exposed only to the experimental half-maps and still provides maps with improved similarity with the atomic model maps.

For *LAFTER* and *Topaz<sup>Tomo</sup>* processed maps, we do not observe an improved resolution cutoff for most of the test maps. *Topaz<sup>Tomo</sup>* has little to no effect on the fit, while *LAFTER* performs inconsistently and can sometimes degrade the fit by a large amount. For a large fraction of *LAFTER* processed maps, the `d_fsc_model_0143` metric could not be calculated.

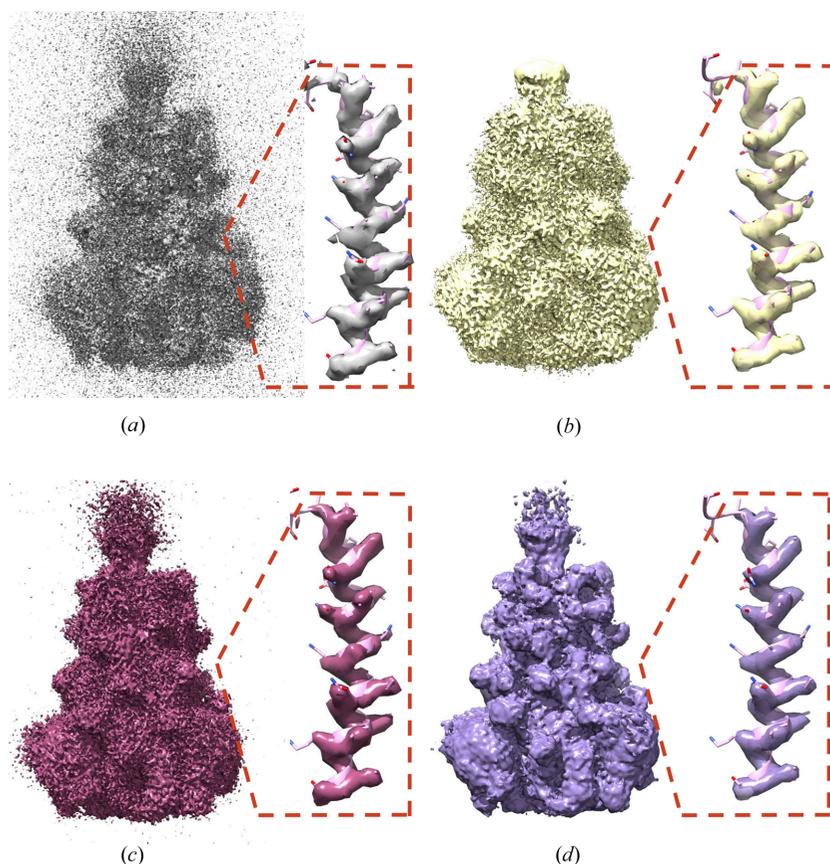
### 3.2. Comparison with published maps

In Figs. 5 and 6, we visually compare  $C_{\text{ref}}$ -denoised maps with the published final 3D cryo-EM maps, as deposited by authors in the EMDB repository. For example, in medium-resolution map EMD-22778 ( $4 \text{ \AA}$ ) of Sec61 membrane channel (Itskanov *et al.*, 2021), densities of some  $\alpha$ -helices improve after denoising (see Fig. 5), while noise due to lipid densities is mostly removed. To exclude that the effects are

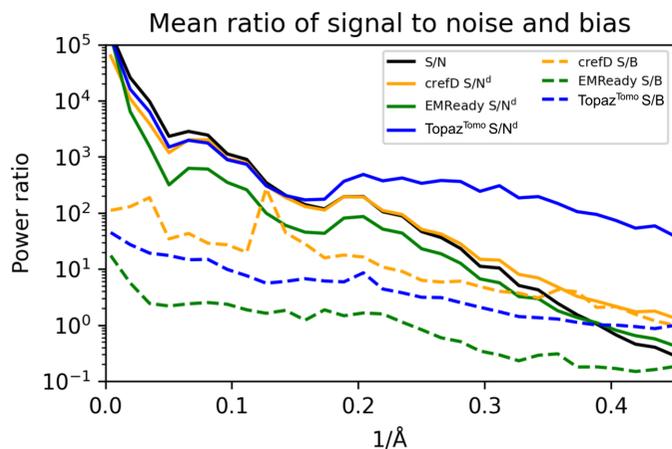
only due to sharpening, we also show noisy-sharpened maps. In the case of the SARS-CoV-1 Spike Protein map, EMD-34420 (Zhang *et al.*, 2023), shown in Fig. 6, denoising removes high-frequency noise that obscures an overview of the domain positions within the spike map [Fig. 6(d)], while the high-resolution information, in particular densities of amino acid side chains, is unaffected [Figs. 6(b) and 6(d)].

### 3.3. Signal, noise and bias in the denoised maps

In Fig. 7, we show ratios of signal-to-noise and signal-to-bias powers for the set of 50 test maps. The signal, noise and bias variances are computed as explained in Section 2.2. At first sight, the analysis shows unfavorable signal-to-bias ratios for the high-frequency range of the *EMReady* processed maps. This seems intuitive since the method modifies maps to include additional signal that is digested from the atomic model based ground truth (and both half-maps are modified in a similar manner by *EMReady*). The SNR is similar or worse when compared with noisy maps, suggesting that *EMReady*'s main action is not denoising. In effect, high-frequency noise is (visually) dampened in the *EMReady* processed maps by the much larger bias signal. In the case of the *crefDenoisier*, the



**Figure 6** Denoising high-frequency noise. The map of SARS-CoV-1 Spike Protein [EMD-34420 (Zhang *et al.*, 2023)] and its fragment focused on a single helix with the fitted atomistic model are presented. The map is part of the training set and has a resolution of  $2.99 \text{ \AA}$ . The noisy map, constructed as a mean of two half-maps, is presented in (a). The final map published in the EMDB repository is shown in (b). The noisy sharpened mean map is presented in (c). The denoised and sharpened mean map is presented in (d). Contouring was tuned to make the maps most similar. The high-frequency structural features of the published and denoised maps are similar (see the  $\alpha$ -helix); however, the denoised map provides a clear outlook of the overall architecture of the spike due to the removal of high-frequency noise.



**Figure 7**

Denoising bias. The ratio of signal to noise [ $\text{var}(S)/\text{var}(N)$ ] and signal to bias [ $\text{var}(S)/\text{var}(B)$ ] is plotted as a function of resolution for noisy and network processed maps. The mean values were calculated with 50 masked maps taken from the test set. Noisy half-maps and denoised half-maps were used to calculate the plotted characteristics, as described in the main text.

SNR is higher in the high-frequency range (denoising effect), while network-based bias is much lower when compared with *EMReady*. However, the overall analysis of data from Fig. 7 is confusing when the *Topaz<sup>Tomo</sup>* model SNR is considered. The plot suggests very effective denoising of the high-frequency signal by *Topaz<sup>Tomo</sup>*, while, with data presented in Fig. S1, we conclude that *Topaz<sup>Tomo</sup>* is a poor denoiser. To better understand the source of this phenomenon, we plot for two maps, signal and noise variances and covariance for two denoised half-maps in Fig. S4. The plot suggests that both noise and signal are dampened by the *Topaz<sup>Tomo</sup>* model in the high-frequency range. Since this effect is correlated, the noise and signal independence assumptions in equations (14) and (15) are not fulfilled, and the variance of noise and bias cannot be accurately computed.

#### 4. Discussion and conclusions

This article presents *crefDenoisier*, a self-supervised deep network model for denoising 3D cryo-EM density maps, and compares its performance with the *EMReady*, *LAFTER* and *Topaz<sup>Tomo</sup>* methods. To our knowledge, it is the first network-based model for denoising this type of 3D EM data in a self-supervised manner. The model is trained on  $\sim 3700$  experimental maps deposited in the EMDB repository to optimize an ideal noise-free map using the presented theory-based loss. We showcase the benefits of denoising in 3D EM map analysis on real-map examples and provide further data that confirm the improved quality of the denoised maps.

The recent sharpening model *EMReady* (He *et al.*, 2023) might, as a side effect, also perform map denoising, since it is trained to match simulated maps in which noise is not present. The presented *crefDenoisier*, on the other hand, is trained to maximize the consistency of raw cryo-EM data, *i.e.* half-maps, with the denoised map employing the  $C_{\text{ref}}$  based analytical loss. The presented analysis suggests that *EMReady* enhances

maps by ingesting additional signal, which we observe as bias in the analysis presented in Fig. 7. On the other hand, *crefDenoisier* restores maps by improving the SNR in the high-frequency range but without introducing additional large signal components, *i.e.* map biasing. The map enhancement due to SNR change is rather moderate, while the additional signal in *EMReady* maps moves them closer to higher-resolution maps and to atomic models, as shown in Figs. 3 and 4. Still, the observed map enhancement by *crefDenoisier* is not given *per se*. First, the model optimization (minimization of  $C_{\text{ref}}$  based loss) does not guarantee that the model restores the ‘true’ noise-free map because the loss is approximate and the space of solutions is degenerate. It is likely that, similar to models trained on a standard  $\mathcal{L}_2$  loss [see Menon *et al.* (2020)], *crefDenoisier* generates a ‘mean’ solution, here a 3D density map, that approximates all possible noise-free maps. Second, *crefDenoisier* is not able to restore the underlying signal if the corruption affects both half-maps similarly; for example, power loss of high-frequency signal in the half-maps.

The two other tested denoising methods, *LAFTER* and *Topaz<sup>Tomo</sup>*, do not enhance map quality in our analysis. For *Topaz<sup>Tomo</sup>*, quality enhancement was not expected, since the model was trained for cryo electron tomography data. However, analysis of *Topaz<sup>Tomo</sup>*-processed maps provides useful insights into the limitations of bias and noise power spectra analysis.

The  $C_{\text{ref}}$  loss is more sensitive to high-frequency signal than a standard  $\mathcal{L}_2$  based loss, which seems to be beneficial for model training (Fig. 1). ‘Noise-to-noise’ models, where one 3D half-map is used as denoising input and the second half-map is used to compute loss, could also be trained with a high-frequency focused loss, such as FSC [equation (1) (Kaczmar-Michalska *et al.*, 2022), see also Tegunov *et al.* (2021)], instead of an  $\mathcal{L}_2$  loss. However, the noise-to-noise setup, as used for example in the *M* software (Tegunov *et al.*, 2021), does not take advantage of both noisy maps during training (one is used to compute loss and the other is used as denoising input). In our model, we use the mean of both half-maps as input for denoising training, with the noise magnitude reduced by  $\sqrt{2}$ . Furthermore, the model is extensively trained on a relatively large dataset (compared with *M*) until convergence, resulting in a versatile denoiser. Training models with a larger training dataset, when it becomes available, might further reduce the leftover noise and improve the denoising results.

While in the presented analysis all the results were calculated for maps that were not part of the training set, it is important to highlight that in real-world applications, the maps requiring denoising could also be incorporated into the training process. This flexibility is made possible by *crefDenoisier*’s ability to train without the need for ground-truth clean maps (self-supervised learning) and has the potential to yield even better denoising results as the model can adapt to more specific noise patterns present in these maps.

We anticipate that the presented model could be beneficial during map analysis and processing steps. Furthermore, since it effectively improves SNR and introduces only low-level bias

in the processed maps, it could find applications as a regularizer in 3D map reconstruction pipelines.

## Acknowledgements

We would like to thank Daniel Böhringer for discussing the manuscript.

## Data availability

The code and model are available in the Github repository at <https://github.com/ajrzepiela/crefDenoisier>.

## Funding information

This work was performed with the support of the Swiss National Science Foundation SPARK grant nr CRSK-3-190804 to AJR and the Novartis FreeNovation grant to AJR and SFN.

## References

- Adams, P. D., Afonine, P. V., Bunkóczi, G., Chen, V. B., Davis, I. W., Echols, N., Headd, J. J., Hung, L.-W., Kapral, G. J., Grosse-Kunstleve, R. W., McCoy, A. J., Moriarty, N. W., Oeffner, R., Read, R. J., Richardson, D. C., Richardson, J. S., Terwilliger, T. C. & Zwart, P. H. (2010). *Acta Cryst.* **D66**, 213–221.
- Bai, X.-C., McMullan, G. & Scheres, S. H. W. (2015). *Trends Biochem. Sci.* **40**, 49–57.
- Batson, J. & Royer, L. (2019). *Proc. Mach. Learn. Res.* **97**, 524–533.
- Bepler, T., Kelley, K., Noble, A. J. & Berger, B. (2020). *Nat. Commun.* **11**, 5208.
- Chen, M., Schmid, M. F. & Chiu, W. (2024). *Nat. Methods*, **21**, 37–40.
- Cheng, Y. (2018). *Science*, **361**, 876–880.
- Chung, J., Durie, C. & Lee, J. (2024). *Life*, **12**, 1267.
- Egelman, E. H. (2016). *Biophys. J.* **110**, 1008–1012.
- Frangakis, A. S. (2021). *J. Struct. Biol.* **213**, 107804.
- Frank, J. & Al-Ali, L. (1975). *Nature*, **256**, 376–379.
- He, J., Li, T. & Huang, S.-Y. (2023). *Nat. Commun.* **14**, 3217.
- Heel, M. van & Schatz, M. (2005). *J. Struct. Biol.* **151**, 250–262.
- Heel, M. van & Schatz, M. (2017). *bioRxiv*, 224402.
- Henderson, R. (1992). *Ultramicroscopy*, **46**, 1–18.
- Höck, E., Buchholz, T.-O., Brachmann, A., Jug, F. & Freytag, A. (2023). *Computer Vision – ECCV 2022 Workshops*, edited by L. Karlinsky, T. Michaeli & K. Nishino, pp. 503–518. Cham: Springer.
- Itskanov, S., Kuo, K. M., Gumbart, J. C. & Park, E. (2021). *Nat. Struct. Mol. Biol.* **28**, 162–172.
- Jakobi, A. J., Wilmanns, M. & Sachse, C. (2017). *eLife*, **6**, e27131.
- Jamali, K., Kimanius, D. & Scheres, S. H. W. (2022). *arXiv:2210.00006*.
- Jiménez-Moreno, A., Štrelák, D., Filipovič, J., Carazo, J. M. & Sorzano, C. O. S. (2021). *J. Struct. Biol.* **213**, 107712.
- Kaczmar-Michalska, J., Hajizadeh, N. R., Rzepiela, A. J. & Nørrelykke, S. F. (2022). *arXiv:2201.03992*.
- Kaur, S., Gomez-Blanco, J., Khalifa, A. A. Z., Adinarayanan, S., Sanchez-Garcia, R., Wrapp, D., McLellan, J. S., Bui, K. H. & Vargas, J. (2021). *Nat. Commun.* **12**, 1240.
- Kingma, D. & Ba, J. (2014). *arXiv:1412.6980*.
- Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M. & Aila, T. (2018). *arXiv:1803.04189*.
- Maddhuri Venkata Subramaniya, S. R., Terashi, G. & Kihara, D. (2019). *Nat. Methods*, **16**, 911–917.
- Menon, S., Damian, A., Hu, S., Ravi, N. & Rudin, C. (2020). *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2434–2442. Piscataway: IEEE.
- Nakane, T., Kotecha, A., Sente, A., McMullan, G., Masiulis, S., Brown, P. M. G. E., Grigoras, I. T., Malinauskaitė, L., Malinauskas, T., Miehl, J., Uchański, T., Yu, L., Karia, D., Pechnikova, E. V., de Jong, E., Keizer, J., Bischoff, M., McCormack, J., Tiemeijer, P., Hardwick, S. W., Chirgadze, D. Y., Murshudov, G., Aricescu, A. R. & Scheres, S. H. W. (2020). *Nature*, **587**, 152–156.
- Palovcak, E., Asarnow, D., Campbell, M. G., Yu, Z. & Cheng, Y. (2020). *IUCrJ*, **7**, 1142–1150.
- Penczek, P. A. (2020). *IUCrJ*, **7**, 995–1008.
- Pintilie, G., Zhang, K., Su, Z., Li, S., Schmid, M. F. & Chiu, W. (2020). *Nat. Methods*, **17**, 328–334.
- Punjani, A. & Fleet, D. J. (2023). *Nat. Methods*, **20**, 860–870.
- Ramírez-Aportela, E., Mota, J., Conesa, P., Carazo, J. M. & Sorzano, C. O. S. (2019). *IUCrJ*, **6**, 1054–1063.
- Ramírez-Aportela, E., Vilas, J. L., Glukhova, A., Melero, R., Conesa, P., Martínez, M., Maluenda, D., Mota, J., Jiménez, A., Vargas, J., Marabini, R., Sexton, P. M., Carazo, J. M. & Sorzano, C. O. S. (2020). *Bioinformatics*, **36**, 765–772.
- Ramlaul, K., Palmer, C. M. & Aylett, C. H. S. (2019). *J. Struct. Biol.* **205**, 30–40.
- Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. (2016). *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788. Piscataway: IEEE.
- Ronneberger, O., Fischer, P. & Brox, T. (2015). *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pp. 234–241. Springer International Publishing.
- Rosenthal, P. B. & Henderson, R. (2003). *J. Mol. Biol.* **333**, 721–745.
- Sanchez-Garcia, R., Gomez-Blanco, J., Cuervo, A., Carazo, J. M., Sorzano, C. O. S. & Vargas, J. (2021). *Commun. Biol.* **4**, 874.
- Tegunov, D., Xue, L., Dienemann, C., Cramer, P. & Mahamid, J. (2021). *Nat. Methods*, **18**, 186–193.
- Terwilliger, T. C., Sobolev, O. V., Afonine, P. V. & Adams, P. D. (2018). *Acta Cryst.* **D74**, 545–559.
- Van Heel, M. (1987). *Ultramicroscopy*, **21**, 95–100.
- Vargas, J., Gómez-Pedrero, J. A., Quiroga, J. A. & Alonso, J. (2022). *Opt. Express*, **30**, 4515–4527.
- Wagner, T., Merino, F., Stabrin, M., Moriya, T., Antoni, C., Apeltaum, A., Hagel, P., Sitsel, O., Raisch, T., Prumbaum, D., Quentin, D., Roderer, D., Tacke, S., Siebolds, B., Schubert, E., Shaikh, T. R., Lill, P., Gatsogiannis, C. & Raunser, S. (2019). *Commun. Biol.* **2**, 218.
- Warshamanage, R., Yamashita, K. & Murshudov, G. N. (2022). *J. Struct. Biol.* **214**, 107826.
- Zhang, X., Johnson, R. M., Drulyte, I., Yu, L., Kotecha, A., Danev, R., Wootten, D., Sexton, P. M. & Belousoff, M. J. (2021). *Structure*, **29**, 963–974.e6.
- Zhang, X., Li, Z., Zhang, Y., Liu, Y., Wang, J., Liu, B., Chen, Q., Wang, Q., Fu, L., Wang, P., Zhong, X., Jin, L., Yan, Q., Chen, L., He, J., Zhao, J., Xiong, X. (2023). *Life Sci. Alliance*, **6**, e202201796.
- Zhong, E. D., Bepler, T., Berger, B. & Davis, J. H. (2021). *Nat. Methods*, **18**, 176–185.