Journal of Structural Biology 213 (2021) 107712

Contents lists available at ScienceDirect

Journal of Structural Biology

journal homepage: www.elsevier.com/locate/yjsbi

DeepAlign, a 3D alignment method based on regionalized deep learning for Cryo-EM

A. Jiménez-Moreno [a], D. Střelák [a,b,d], J. Filipovič [d], J.M. Carazo [a,*], C.O.S. Sorzano [a,c,*]

[a] Centro Nac. Biotecnología (CSIC), c/Darwin, 3, 28049 Cantoblanco, Madrid, Spain
[b] Faculty of Informatics, Masaryk University, Botanická 68a, 662 00 Brno, Czech Republic
[c] Univ. San Pablo – CEU, Campus Urb. Montepríncipe, 28668 Boadilla del Monte, Madrid, Spain
[d] Institute of Computer Science, Masaryk University, Botanická 68a, 60200 Brno, Czech Republic

ARTICLE INFO

Keywords:
3D alignment
3D reconstruction
Cryo-EM
Deep learning
Machine learning

ABSTRACT

Cryo Electron Microscopy (Cryo-EM) is currently one of the main tools to reveal the structural information of biological specimens at high resolution. Despite the great development of the techniques involved to solve the biological structures with Cryo-EM in the last years, the reconstructed 3D maps can present lower resolution due to errors committed while processing the information acquired by the microscope. One of the main problems comes from the 3D alignment step, which is an error-prone part of the reconstruction workflow due to the very low signal-to-noise ratio (SNR) common in Cryo-EM imaging. In fact, as we will show in this work, it is not unusual to find a disagreement in the alignment parameters in approximately 20–40% of the processed images, when outputs of different alignment algorithms are compared.

In this work, we present a novel method to align sets of single particle images in the 3D space, called DeepAlign. Our proposal is based on deep learning networks that have been successfully used in plenty of problems in image classification. Specifically, we propose to design several deep neural networks on a regionalized basis to classify the particle images in sub-regions and, then, make a refinement of the 3D alignment parameters only inside that sub-region. We show that this method results in accurately aligned images, improving the Fourier shell correlation (FSC) resolution obtained with other state-of-the-art methods while decreasing computational time.

1. Introduction

Single Particle Analysis (SPA) for Cryo Electron Microscopy (Cryo-EM) has become one of the major tools to reveal the three-dimensional (3D) structure of macromolecules at high resolution, allowing to understand molecular interactions and being crucial to start understanding the function of biological ensembles (Nogales, 2016). When high resolution is achieved in the reconstructed 3D maps, it is possible to recover a great amount of biological information. However, it is common to find 3D maps or part of them with lower resolutions, which is due to errors in the reconstruction procedure (Henderson, 1992), among other problems.

One of the most complicated steps in a common workflow to obtain a 3D reconstructed map is the highly error-prone 3D alignment step. The goal of the 3D alignment is to find parameters describing orientation and position in a 3D sphere for every particle image. These parameters are: the in-plane rotation and the shift translations in both axis of the 2D projection, and then two angles to orient the projection in the 3D sphere (commonly named *rotation* and *tilt*). These five parameters completely define the orientation of every particle image in the 3D space.

The 3D alignment step is affected by the very low Signal-to-Noise Ratio (SNR) that reduces the accuracy in the obtained alignment parameters, which results in artifacts in the reconstructed map. Moreover, as this step is an optimization problem in a high-dimensional space, common statistical approaches can easily get stuck in local minima. As we will demonstrate later, it is common to find a disagreement of more than 10° in the alignment parameters obtained with different alignment algorithms in approximately 20–40% of the particle images.

1.1. State-of-the-art

We can find several ways to tackle the 3D alignment in the literature,

* Corresponding authors at: Centro Nac. Biotecnología (CSIC), c/Darwin, 3, 28049 Cantoblanco, Madrid, Spain (C.O.S. Sorzano).
E-mail addresses: carazo@cnb.csic.es (J.M. Carazo), coss@cnb.csic.es (C.O.S. Sorzano).

https://doi.org/10.1016/j.jsb.2021.107712
Received 2 July 2020; Received in revised form 2 February 2021; Accepted 21 February 2021
Available online 4 March 2021

(e.g. Penczek et al., 1992; Penczek et al., 1994; Scheres et al., 2005; Scheres et al., 2007; Scheres, 2012; Elmlund et al., 2013; Vargas et al., 2014; Sorzano et al., 2015; Punjani et al., 2017; Sorzano et al., 2018; Sorzano et al., 2018). The standard approach to the 3D alignment problem was the so-called "Projection Matching" (Penczek et al., 1992; Penczek et al., 1994). Then, statistical tools as Maximum Likelihood (ML), Maximum *a posteriori (MAP), and Bayesian prior methods started to be a relevant way to face the alignment problem, following in the footsteps of* Sigworth (1998) *where ML was firstly used for Cryo-EM.* Scheres et al. (2005), Scheres et al. (2007) *and* Scheres (2012) *presented alignment procedures based on ML and Bayesian reconstruction, in which the particle images can take all projection directions with different weights, which were calculated from a Bayesian prior on the distribution of noise and signal coefficients. This method solved the optimization problem in a greedy way, starting from an initial estimation of the 3D map to be reconstructed. In* Elmlund et al. (2013) *a similar optimisation probabilistic approach was proposed but in a non-greedy way, in which an image could be assigned to a subset of so-called feasible directions, using different weights calculated from a heuristically determined function, which could help to avoid local minima.* Vargas et al. (2014) *described also a statistical approach focused on trying to avoid local minima by reducing the search space using image subsets, randomly assigning orientations, and checking which of the assignments was more successful.* Sorzano et al. (2015) *considered the alignment problem as a weighted least squares optimisation based on the concept of statistical significance, rather than a closed form optimisation of a given functional under a simplified set of assumptions. Novel ML implementations based on branch-and-bound technique, stochastic gradient descent, and GPU processing have gained much attention, significantly reducing the processing time (*Punjani et al., 2017*).* Sorzano et al. (2018) *proposed to use the statistical significance as weight, instead of using the likelihood, and recommended an angular assignment in which each image receives a single angular orientation, unlike some previous works. Other works, (e. g.,* Sorzano et al., 2018*), took the approach of generating many different volumes (preferably with different algorithms) and ranking the volumes according to their fit to the experimental data.*

Despite the availability of all these methods, current practice shows that, due to the previously mentioned problems, there are situations in which the approaches above fail to produce a satisfactory result and more robust techniques are still needed.

Our method presents a new framework based on deep learning to manage the 3D alignment problem. Deep learning is a machine learning technique, derived from neural networks, able to learn from multiple levels of feature representation. In the last years, it has become a revolutionary tool in computer vision, e.g. image classification, object recognition, and tracking. In Cryo-EM, deep learning is being used already for particle picking, or annotation of different parts in the reconstructed structure of proteins, (e.g. Wang et al., 2016; Li et al., 2016; Zhu et al., 2017; Chen et al., 2017; Sanchez-Garcia et al., 2018; Wagner et al., 2018; Zhang et al., 2019). There are some attempts to use deep learning in the 3D reconstruction step, (e. g. Gupta et al., 2020; Zhong et al., 2019; Zhong et al., 2020. Gupta et al., 2020) used a generative adversarial network to learn the 3D density map whose projections are the most consistent with the given input particle set. However, this approach was not able to produce a sufficiently accurate 3D map to resolve the biological structure. Zhong et al. (2019) and Zhong et al. (2020) presented one of the first successful approach for Cryo-EM reconstruction based on deep learning, specifically a variational autoencoder is used to find out discrete states as well as continuous conformational changes. Thus, this method was able to manage 3D heterogeneity; however, the particle orientation needed to be previously determined by other technique. Therefore, to the best of our knowledge, our proposal is one of the first methods based on deep learning dealing with the 3D alignment process.

1.2. Introduction to DeepAlign

In this work, we present DeepAlign, a new proposal built on Convolutional Neural Networks (CNNs), that have revolutionized the field of neural networks for image processing, as they have boosted the performance in a large variety of tasks. The CNNs are designed with the first part of convolutional layers devoted to extracting several levels of features based on a non-linear filtering process. The second part of layers is dedicated to the classification itself, generating a label for the input image knowing the features previously calculated in the convolutional part of the network (more details will be given in the following section). Unlike common machine learning approaches, which typically use handcrafted filters to extract the features, CNNs have the ability to learn these filters on its own by means of the feature extraction layers.

Moreover, our proposal is built on a regionalized basis. Creating only one network to predict the location of the particle images in the whole 3D sphere can result in a very high-complexity network due to the difficulty of this task. Instead, we propose to divide the 3D projection sphere, which means the angular space of orientations in 3D, into non-overlapping regions and create a simpler deep neural network in each region to detect if the experimental image comes or not from that region, which can be done with high accuracy. Following this reasoning, we obtain so many deep neural networks as regions and, for every particle image, we calculate the probability of that image coming from each region, and select that with the highest probability. The final alignment parameters (rotation, tilt, and in-plane angle and shifts) are finally obtained running a simplified alignment procedure based on correlation only in the region of interest.

Additionally, taking into account the high disagreement that can be found in the alignment parameters obtained with different algorithms, we also propose a consensus tool. The idea is to select only those particle images in which the angular differences between alignment methods are low, so it is more likely that these images are accurately assigned. Building the 3D reconstructed map taking into account only those images, could avoid the appearance of artifacts and improve the obtained resolution.

2. Methods

2.1. Regionalized deep learning approach

Our deep learning proposal relies on CNNs, which have successfully proved their usefulness in a variety of problems related to image processing.

In a CNN, the convolutional layers are able to successfully capture the spatial dependencies in an image through the application of consecutive filters of different sizes, going from basic features, like edges or corners, to detailed features more specific to the problem to be solved. The filter kernels are the values to be learned in the training process. A convolution operation will be applied between image and filters to obtain the features present in the image. In other words, the CNN network can be trained to understand the characteristics of the image better than other approaches.

The fully connected layers in the second part of the network is a way of learning a non-linear function in the feature space, weighting the features obtained with the previous convolutional part. The output of these layers are real values that will be converted into a label (or probability). In this way, the network classifies the input image into that class with the highest probability.

Specifically, the design of our networks is as follows:

- The size of the input layer is that of the input particle images. This can be downsampled to avoid memory overload and to alleviate the computational burden, while we try to preserve the main details of the images that are decisive to properly train the networks.

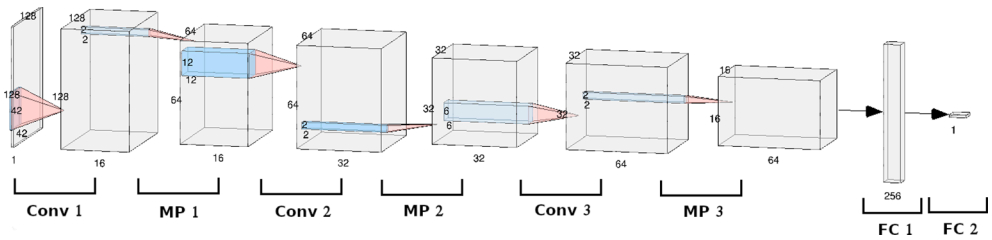


Fig. 1. Network design. For an input image of size 128 × 128, the first convolutional layer (Conv 1) is created with 16 filters of size 42 × 42, then max pooling of size 2 × 2 is applied (MP 1), the second convolutional layer (Conv 2) has 32 filters of size 12 × 12, another max pooling layer follows (MP 2), and the last convolutional layer (Conv 3) has 64 filters of 6 × 6 followed by the last max-pooling (MP 3). The first fully connected layer (FC 1) has a size of 256 neurons and the output layer (FC 2) with one neuron will give us the classification probability.

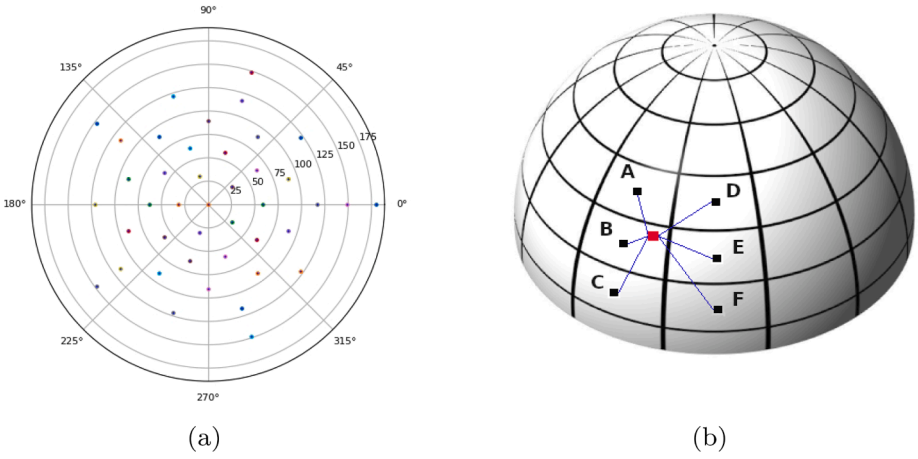


Fig. 2. (a) Top view of region centers shown in dots, example with regions separated 30°. (b) Illustrative example of the labeling for a particle: the distance between the particle (red point) and all the region centers (for clearness just six regions are drawn, A, B, C, D, E and F) is calculated, the minimum distance give the label for the particle (B in this example).

- Three convolutional layers are applied with kernel sizes adapted to the input (1/3, 1/10, and 1/20 of the input size, respectively). The number of filters is 16 for the first layer, 32 for the second, and 64 for the last one.
- In between every convolutional layer, a normalization and max-pooling with size 2 × 2 (which will halve the input in both spatial dimensions) are carried out.
- A dropout layer is included to prevent overfitting after the convolutional part. This layer randomly drops a fraction of input units at

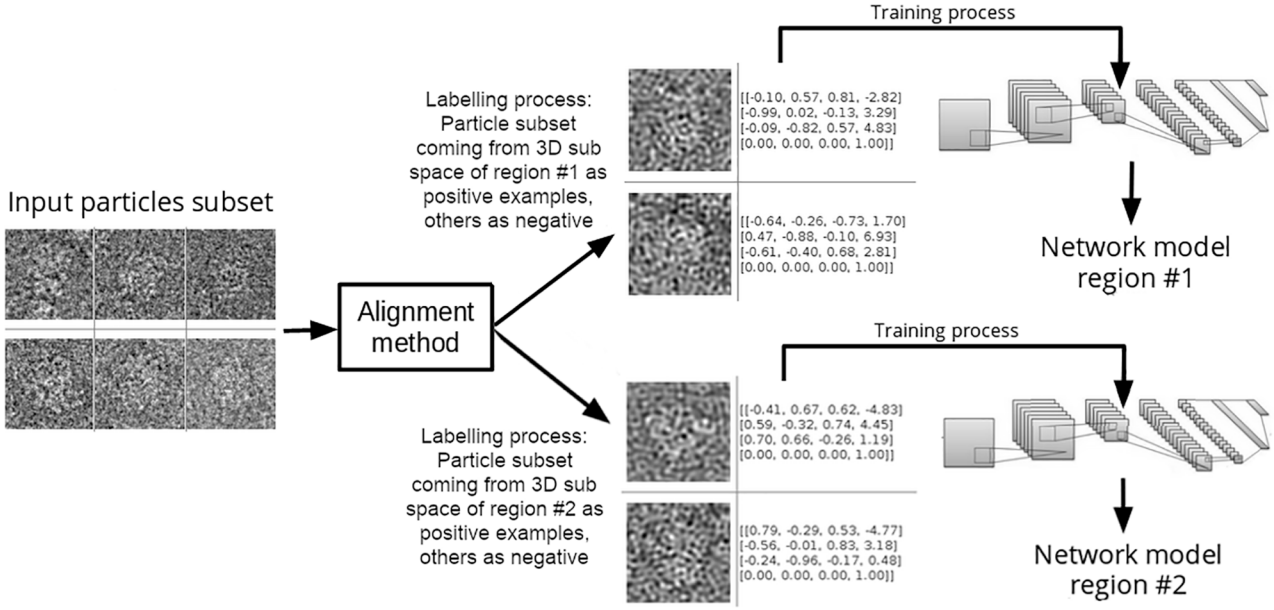


Fig. 3. A schematic representation of the training process. Every particle has an angular assignment and can be assigned to a specific region. The subset of particles assigned to every region will be used for training that network model.

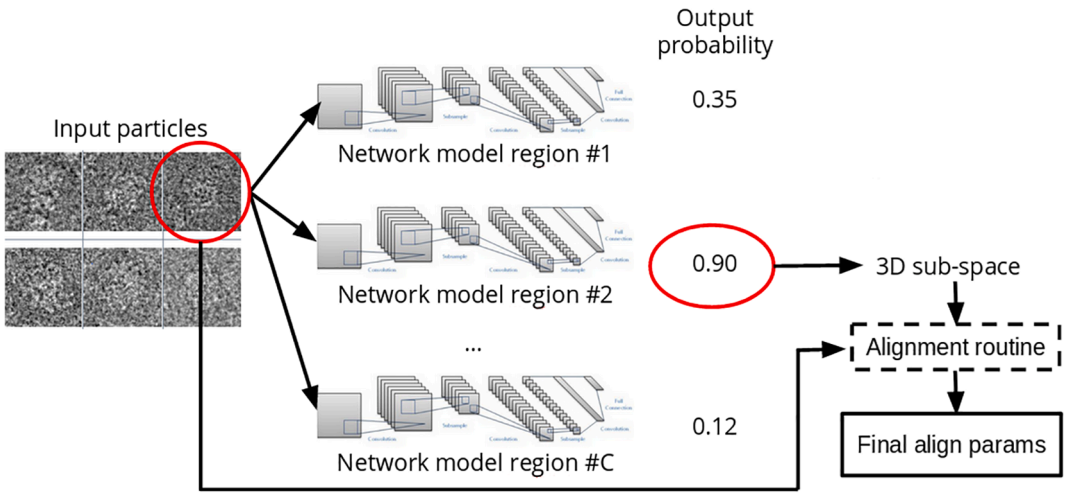


Fig. 4. A schematic representation of the prediction process.

each update during training time. In our design, that fraction is fixed to 0.2.

- We used two fully connected layers, the first with 256 neurons and the second with just one neuron (the output layer with only one neuron will give us the output probability). The first fully connected layer uses a Rectified Linear Unit (ReLU) activation function, whilst the second layer uses a sigmoid.
- The optimizer for the training process is Kingma et al. (2014) with a learning rate of 0.002.

A graphical representation of the network design can be found in Fig. 1.

CNNs identical to the one explained above, are set up to work on a regionalized basis. The idea of working by regions is the following: we need to find for every particle image the set of alignment parameters to place that image in the 3D sphere. However, to predict with only one network the whole set of parameters could be a very difficult task, considering the low SNR and the high probability of finding a local minimum in the solution space. For that reason, we decided to simplify the problem and divide the 3D sphere into non-overlapping smaller regions (an example is shown in Fig. 2(b)). For every region, a unique CNN is trained to give the probability of a particle image belonging to that region (but not the specific projection direction corresponding to that particle image), which is a simpler problem that can be managed with the low complexity CNN described above. The selected region for a particle image is the one with the highest output probability. The final alignment parameters are obtained running an alignment method based on correlation only in the selected region, which reduces the complexity burden.

2.2. CNNs training

To train the CNNs we need a set of particle images with the associated label of the region where the image comes from. To this end, a small random subset (approximately 10% of the input size) of the input particle set must be aligned with another method, (e.g., Sorzano et al., 2018; Scheres, 2012), or (Punjani et al., 2017). Then, knowing the alignment parameters and using the distance to the center of regions (Fig. 2), the image label will be the region whose center is closer to the image. To train a CNN, we take the subset of images assigned to that region as positive labels and all the remaining ones as negative labels. This results in a very unbalanced number of images for every label, which can be problematic in the training process. Therefore, we build balanced sets by randomly sampling these two sets to a final equal size.

Moreover, we use a data augmentation procedure to increase the power of the network to recognize particles in different in-plane orientations. During the data augmentation we take a training image (particle image) and we repeat it several times with random rotations and shifts in the in-plane parameters. In Fig. 3 a schematic representation of the training process is shown.

Regarding the accuracy of the training process, although CNNs are known for being robust to mislabelling and we can expect good behavior from them (Rolnick et al., 2018), it is key to check how the error rate evolves during the training process. To obtain a low error rate on the validation set is the way to know if the training is correct. As we will show in the Results section, to achieve a 3D reconstruction in the mid-range of resolutions with, approximately, a 10% of the particle images, was enough in our test cases to get a proper training set, even in challenging cases with very noisy images. On the other hand, if higher reliability in the angular assignment of the training set is required, a higher percentage of images can be used to train, or a consensus before the training could be applied. This means, to use two different algorithms to assign the angles for the training particles, selecting then the subset with coincident angles, which could assure us to have very accurate assignments. Also, Sorzano et al. (2018) can be used to build a reconstruction in a particular range of resolutions, as this method has an option to select the target resolution and work in that range.

2.3. Predicting image label and obtaining final alignment parameters

Fig. 4 shows a summary of the prediction and final alignment steps. Once the CNNs for every 3D region are ready, prediction can be carried out for the whole input set of particle images. Every image is presented to all CNNs and the output probabilities are gathered. The region with the highest output probability is selected for each image. In this way, the algorithm locates for each image a narrow 3D region from which it likely comes. To find out the alignment parameters, we run an alignment method based on correlation; specifically, this method is a GPU version of the significant assignment of Xmipp, (Sorzano et al., 2015; Sorzano et al., 2018). This alignment is carried out in every region of interest and with the particle subset assigned to it. This greatly reduces the search space as the number of comparisons between input images and reprojections of the reference volume, which is the most expensive part of any 3D angular assignment algorithm, is divided by the number of 3D regions.

In some cases, several of the highest CNN output probabilities could have similar values, which could point out to regions where it is difficult to distinguish between them. To manage this situation, we give the option to select the number of regions to be considered per image. That is, several regions (those with the highest output probabilities) can be selected for one particle image and, then, the alignment algorithm will

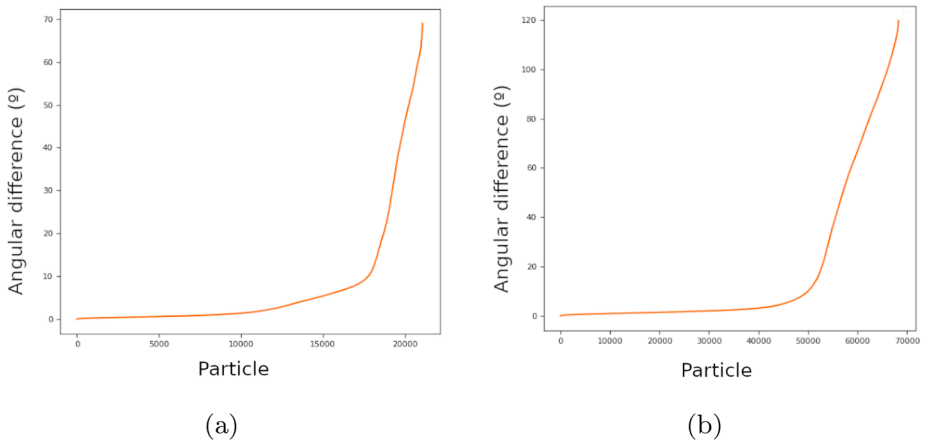


Fig. 5. Angular differences sorted from lower to higher. (a) Xmipp Highres vs Relion for *proteasome*. (b) Relion vs DeepAlign for *ribosome*.

be in charge of selecting the best 3D location, that could be inside any of the available regions. This is also a way to minimize classification errors by the deep learning approach, even the errors coming from the mislabelling in the training process (these labels come from the alignment parameters obtained by another method that will have some error percentage) will be reduced thanks to the possibility of using several regions to align. Although there is a tradeoff between classification error rate and complexity burden to take into account.

2.4. Complexity optimization

The training and final alignment steps are responsible for the main complexity burden in our proposal. All the deep learning procedures included in this method are developed using Keras library (Chollet et al., 2015) and exploit its GPU implementation.

The training step depends on the number of regions considered (as it is equal to the number of CNNs) that, on its turn, depends on the region size and the symmetry, as only the non-symmetric part of the 3D space is considered. Moreover, this step is parallelized at GPU level, as training of every CNN is completely independent of each other. So, when several GPUs are available, these tasks can be divided among them. We must highlight that the complexity of the training per region does not depend on the number of input particle images, as we use data augmentation to keep the same training set size. The whole training step has a complexity that depends on the parameters of the training, e.g. number of epochs and batch size, that can be selected by the user, and the size of the training images.

The prediction is carried out for the whole set of input particles and considering all the available regions, so its runtime depends on both. Anyway, this time is clearly lower than the one required for the training step. Thus, for simplicity, we decided not to parallelize it at GPU level.

The final alignment based on correlation is also implemented in GPU and, as in the training step, the alignment for every region is independent of the other ones, so it can be also additionally parallelized at GPU level. Thus, the alignment in every region can be executed in different GPUs.

The method presented in this paper was implemented in Xmipp (de la Rosa-Trevín et al., 2013) and included in Scipion (de la Rosa-Trevín et al., 2016).

2.5. Consensus tool

A comparison in angular assignments between several methods is presented in Fig. 5. Specifically, we plot the angular differences, from lowest to highest, between the angles obtained for every particle image with Xmipp Highres (Sorzano et al., 2018) and Relion (Scheres, 2012) for structure *T20S proteasome* in part (a) of the figure, and between (Scheres, 2012) and DeepAlign for structure *Plasmodium falciparum* 80S ribosome in part (b). These results show that approximately 70–80% of images have angular assignments that differ in less than 10°. Thus, there is a significant number of images in which the angular differences highly increases up to very high values, indicating around 20–30% of images cannot be accurately aligned (however, we have seen this value to go up to 40–50% for some datasets). Obviously, if these significant disagreements are translated in wrongly assigned images, the obtained resolution for the 3D reconstructed map could be damaged.

The consensus tool presented in this work aims to solve the previous problem. If we have several alignment results for every particle image, we can check if the different methods give similar solutions or not. In the case of low angular differences, there is no evidence that the particles come from different directions. Otherwise, when the angular differences are large, the probability of a wrong assignment could be significant. The consensus tool is in charge of discarding images for which the angular difference is above some user-defined threshold. Images for which two or more angular assignment algorithms agree in their orientation, are used to refine the 3D map. This procedure could improve the obtained resolution as we are discarding particle images that do not contain enough information to be properly located.

A possible caveat of any consensus tool comes from the comparison using similar techniques, as they can discover similar local minimum. Since most of the available techniques to carry out the alignment process rely on ML approaches, we can expect a similar behavior among them in terms of accuracy. Therefore, most of the subset with the wrongly assigned images could have similar statistical characteristics, and the same holds true for the subset of well assigned images. DeepAlign is based on a completely different approach. Its hits and miss subset will have a different statistical basis, giving extra information to select the particle image subset that will likely be correctly aligned.

3. Results

In this section, we present the results obtained with DeepAlign in comparison with other methods in the state-of-the-art, specifically Xmipp Highres (Scheres, 2012), Relion (Sorzano et al., 2018) (v3.0), and CryoSparc (Punjani et al., 2017) (v2.14). The structures *Plasmodium falciparum* 80S ribosome (with codes 10028 in EMPIAR and 2660 in EMDB databases), T20S proteasome (with codes 10025 in EMPIAR and 6287 in EMDB), and SARS-CoV-2 Spike (Melero et al., in press) have been used. The GPUs used were GeForce RTX 2080 Ti with 11 GB of memory, and the CPUs were Intel(R) Xeon(R) Silver 4114 at 2.20 GHz.

3.1. Plasmodium falciparum 80S ribosome

The tests with this structure were carried out with a distance of 30°

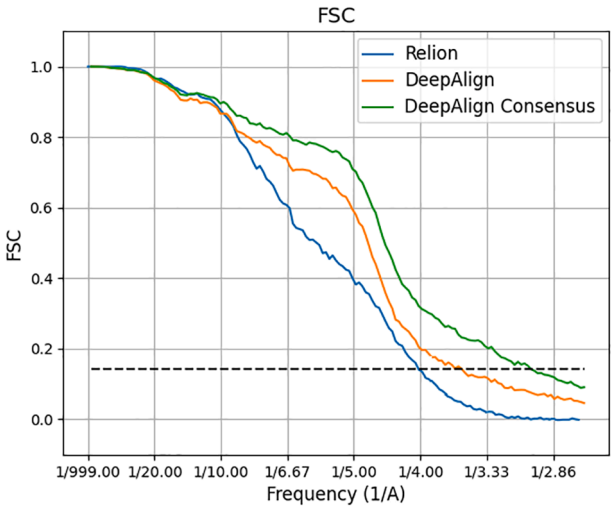


Fig. 6. FSC curves obtained for *ribosome*. Relion 4.0 Å, DeepAlign 3.5 Å, and DeepAlign consensus 2.9 Å are compared.

between region centers. As this structure presents no symmetry, the total number of regions considered was 42. The number of experimental images was 85,012 with a size of 300 × 300 pixels and a pixel size of 1.34 Å/pixel. The original achieved resolution for this structure was 3.2 Å (Wong et al., 2014). A target resolution of 10 Å was used to rescale the input particle images and volume, thus the image size was reduced to 120 × 120 pixels. From this input set, we randomly took 5,000 images for training that were aligned with Relion and labeled according to the region from which they come. Data augmentation procedure generates a total of 10,000 images per region, applying random in-plane rotations and shifts to every image. Thus, in the training of a CNN we have 10,000 positive labeled images and 410,000 negative examples, which is a very unbalanced set. To solve this, in every batch generated during the training, we maintain the same proportion of positive and negative examples. Additionally, we selected the two best regions per image, to find the optimal location inside them.

The number of epochs for training the CNNs was 10, and the batch size was 128. The training process of a region started with a loss (measured with mean absolute error) near 0.9 and accuracy of 0.5 (corresponding to random predictions), in the first epoch a loss of 0.2 and accuracy of 0.8 were already achieved, and the training process was finished with a loss of 0.1 and accuracy of 0.9 on the validation set. So, we were able to generate a proper training set with which the network can learn the alignment parameters quite fast.

The training time (without taking into account the required time by Relion to align the 5,000 images) per region took 20 min on average, the prediction step 40 min in total, and the alignment inside the two selected regions 1 min per region. Running the process in two GPUs, the whole algorithm required approximately 9 h (additional steps, such as data read/write and preprocessing took additional 30 min). After running a local refinement (using Xmipp Highres) we got a resolution of 3.8 Å. It must be taken into account that we started with the information of 5,000 aligned particles as input to our method that would give rise to a very rough 3D map estimation of around 10 Å. Relion running also in two GPUs took 20 h to converge and obtained a resolution of 4.0 Å. If we run one more local refinement step of the DeepAlign results, we reached a processing time of also 20 h, the same as Relion, but the obtained resolution was 3.5 Å. The consensus tool was tested in this example comparing the alignment angles obtained with our proposal and Relion and selecting the particle subset with a difference between them in less than 5°. This, reduced the number of particle images in approximately 27,000 images (from 85,012 to 57,886) which is over 30%. Then, a local refinement using only this subset was carried out. We obtained a resolution of 2.9 Å compared to the previous 3.5 Å. This result indicates that

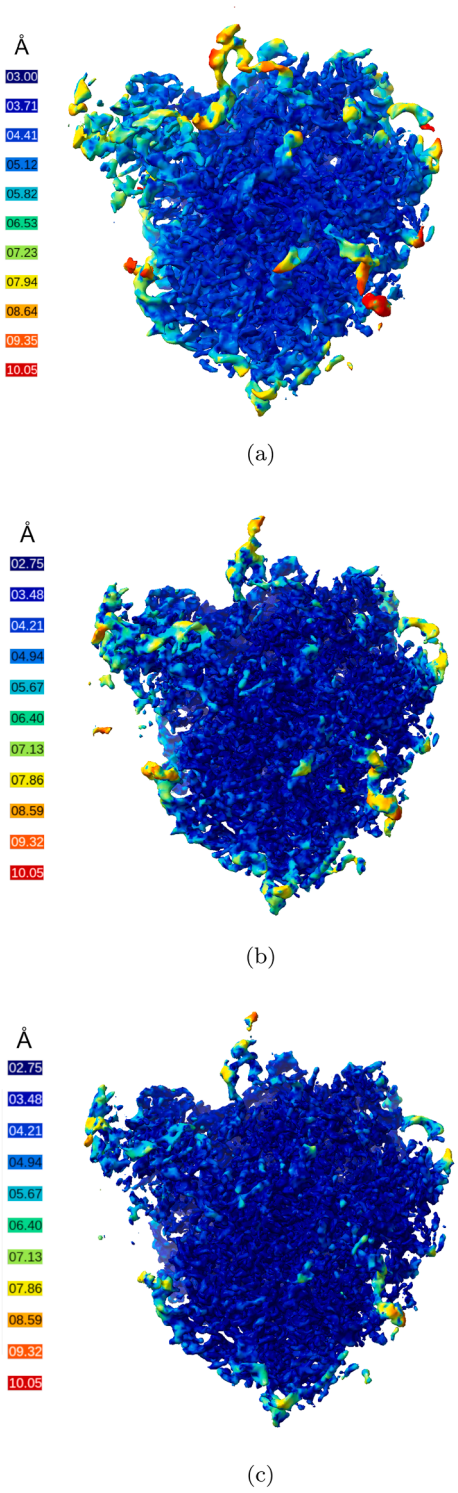


Fig. 7. Local resolution of the reconstructed 3D maps for *ribosome*. (a) Relion, (b) DeepAlign, and (c) DeepAlign consensus.

a lower number of images with an accurate alignment leads to better reconstruction than using a bigger set of particles containing misaligned or noisy images. The Fourier shell correlation (FSC) curves are presented in Fig. 6.[1]

[1] To measure the FSC after using Xmipp Highres or a local refinement based on this method, we have disabled the post-processing options of this method, thus we obtain FSC curves comparable to the other approaches.

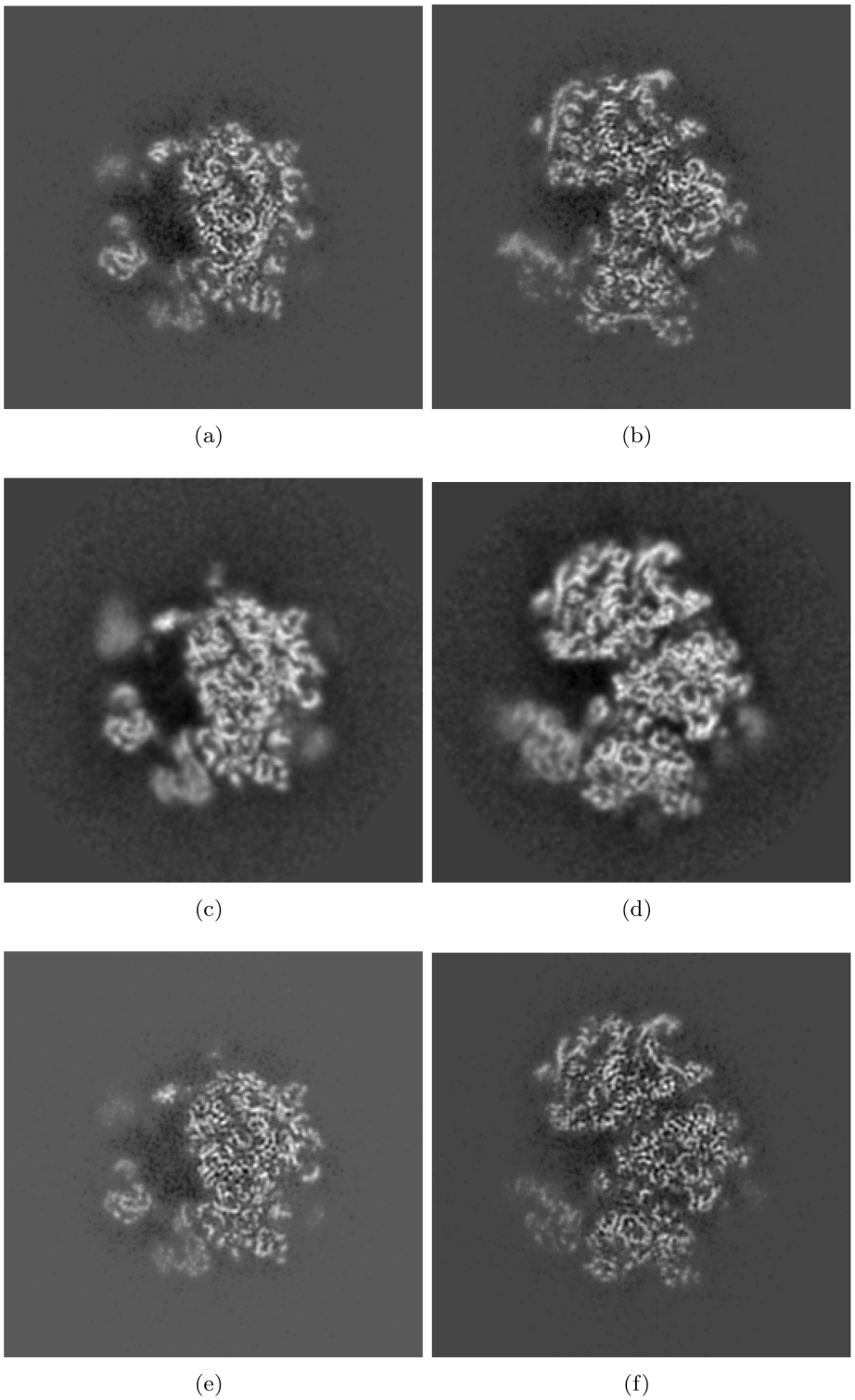


Fig. 8. Central slices of the reconstructed 3D map for *ribosome*. (a) and (b) Z-axis and Y-axis with DeepAlign (3.5 Å). (c) and (d) Z-axis and Y-axis with Relion autorefine (4.0 Å). (e) and (f) Z-axis and Y-axis with DeepAlign consensus tool (2.9 Å).

The local resolution was also measured using Monores (Vilas et al., 2018) and comparing the three considered approaches, obtaining the results presented in Fig. 7. This analysis confirms the trend of the FSC curves, our proposals were able to obtain better resolution in most of the voxels of the structure, specially improving in the inner part of the structure from the obtained 3.0 Å with Relion to 2.75 Å with DeepAlign and consensus. Some selected slices taken from the three reconstructed 3D maps are presented in Fig. 8.

Finally, we represent the 3D structures obtained with DeepAlign consensus tool in comparison with Relion in Fig. 9. (a) and (b) represent the whole structure where some densities started to appear in the outer areas of the structure that in the map obtained with Relion are lost (red circles in Fig. 9(b)). Parts (c) and (d) of the figure show a zoomed area on a helix with the deposited atomic model (PDBPDB3j7j79) fitted in it. As it can be seen, after the post-processing and fitting steps similar results are achieved with both methods. We used Refmac (Murshudov et al.,

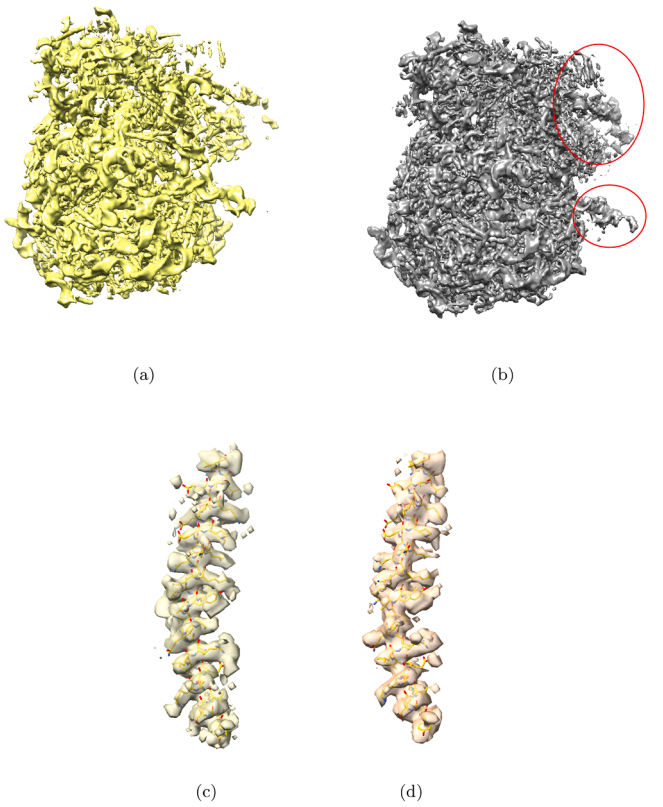


Fig. 9. 3D reconstructed maps for *ribosome*. (a) and (b) Whole 3D maps reconstructed by Relion and DeepAlign, respectively. (c) and (d) Zoom in a specific helix for Relion and DeepAlign reconstructions, respectively, with the atomic model fitted.

2011) to refine the fitting, obtaining an average Fourier shell correlation of 0.45 with DeepAlign and 0.43 with Relion (note that the model only corresponds to one sub-unit), confirming that both methods are able to perform similarly.

3.2. T20S proteasome

This structure presents a dihedral symmetry (D7) with a size of 400 × 400 pixels and a pixel size of 0.66 Å/pixel. The original achieved resolution was 2.8 Å (Campbell et al., 2015). The distance between region centers was 20° which generates 92 regions to cover the whole 3D space but only 9 were actually assigned to the asymmetric part of the molecule. The algorithm was configured to select the best two regions to find out the location for each particle. We had 26,230 experimental particles as input, from which only 3,000 (randomly selected) were used to carry out the training process. These 3,000 images were aligned and afterwards labeled with Xmipp Highres. With data augmentation we were able to generate 10,000 positive examples and 80,000 negative examples to train every network. These sets were balanced during the generation of the batches for the training, as in the previous example. The target resolution was 4 Å, so the image size was reduced to 197 × 197. All the remaining steps and parameters to make the training stayed as in the previous example.

On average, the training process of a region started with a loss near 0.9 and accuracy of 0.5 (corresponding to random predictions), in the first epoch a loss of 0.2 and 0.85 of accuracy were achieved and, the training process was finished with a loss of 0.02 and an accuracy of 0.99 on the validation set.

In this example, the new alignment parameters obtained with DeepAlign and locally optimized lead to a reconstructed 3D map with a resolution of 2.9 Å, compared to the 3.3 Å obtained with Xmipp Highres.

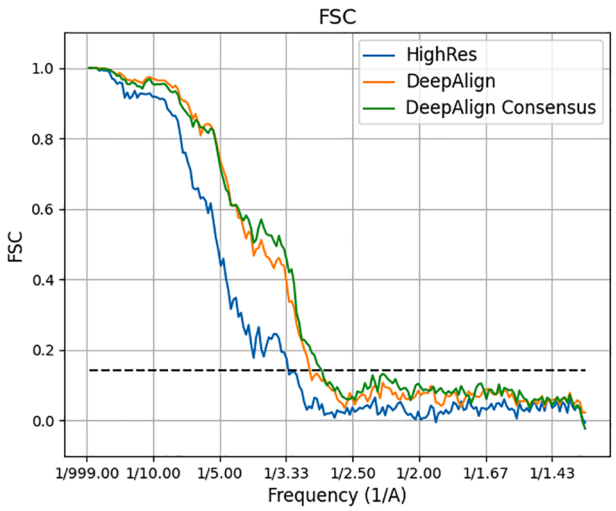


Fig. 10. FSC curves obtained for *proteasome*. Xmipp Highres 3.3 Å, DeepAlign 2.9 Å, and DeepAlign consensus 2.7 Å are compared.

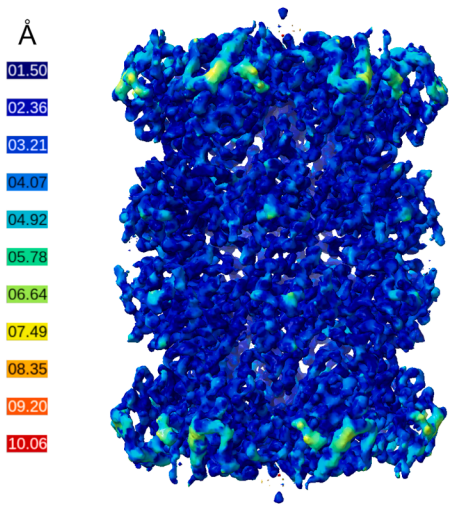
The consensus tool was run with the subset of images for which the angular difference was lower than 5°. This, reduced the input set of particle images from 26,230 to 19,086, a 28% of reduction. After a local refinement, the achieved resolution was 2.7 Å. The local resolution analysis with Monores also showed that Xmipp Highres and our proposals were able to obtain a high resolution reconstruction in most of the areas of the structure, but DeepAlign and the consensus tool got some improvements. These results prove that our method was able to find a slightly better solution. Figs. 10–12 show the obtained FSCs, the local resolution, and some slices taken from the reconstructed 3D maps.

Xmipp Highres needed more than 2 days using 24 cores to make the whole alignment process. The proposed method took 30 min, on average, to train every region (without taking into account the time to firstly align 3,000 images with Xmipp Highres). Thus, using 5 GPUs training in parallel, DeepAlign was able to complete the training process in just 1 h. The prediction time took 10 min. Finally, the step to obtain the final alignment parameters took 4 min per region, on average, so a total of barely 10 min in 5 GPUs aligning in parallel. The entire process was done in 1 h and a half using 5 GPUs.

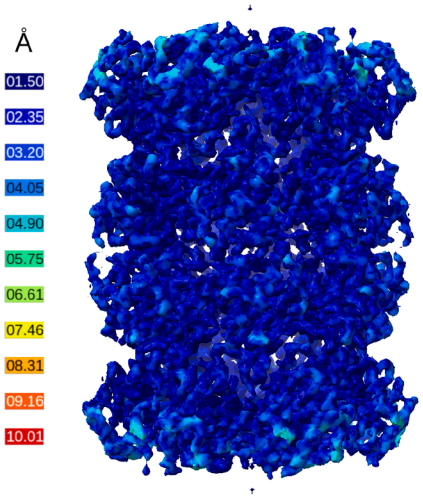
The 3D maps obtained with Xmipp Highres and with DeepAlign consensus tool can be seen in Fig. 13. The whole 3D structures for both methods are presented in (a) and (b). Sharper details showed up in the DeepAlign reconstruction and some new densities appeared in the outer central part of the macromolecule (highlighted with red circles). A zoomed area on a pair of helices is shown in Fig. 13(c,d), showing slightly sharper details in the areas expected to correspond with side chains. In this example, there is no atomic model included in the deposited data, so it is not included in the analysis.

3.3. SARS-CoV-2 Spike

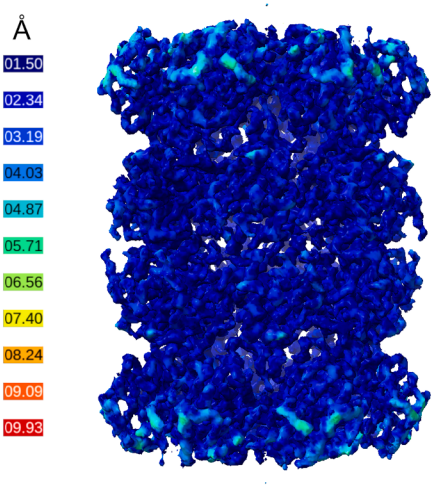
In this test case, our goal is to check if our proposal is able to achieve results comparable to other state-of-the-art approaches with a more challenging data set. We use the SARS-CoV-2 Spike data set (Melero et al., in press) whose characteristics are: size of 400 × 400 pixels, pixel size of 1.05 Å/pixel, and no symmetry. We considered a distance between regions of 30°, which results in 42 regions, and a target resolution of 4 Å to rescale the input particle images and volume to a size of 314 × 314 pixels. The data set consisted of 36,558 images, from which we randomly took 5,000 for the training. The alignment of the training set was carried out with CryoSparc. As in the previous test, data augmentation was used to generate a more complete training set with 10,000 images per region, and balanced sets were generated during the creation batches for the training. The remaining parameters were kept as in the



(a)



(b)



(c)

Fig. 11. Local resolution of the reconstructed 3D maps for *proteasome*. (a) Xmipp HighRes, (b) DeepAlign, and (c) DeepAlign consensus.

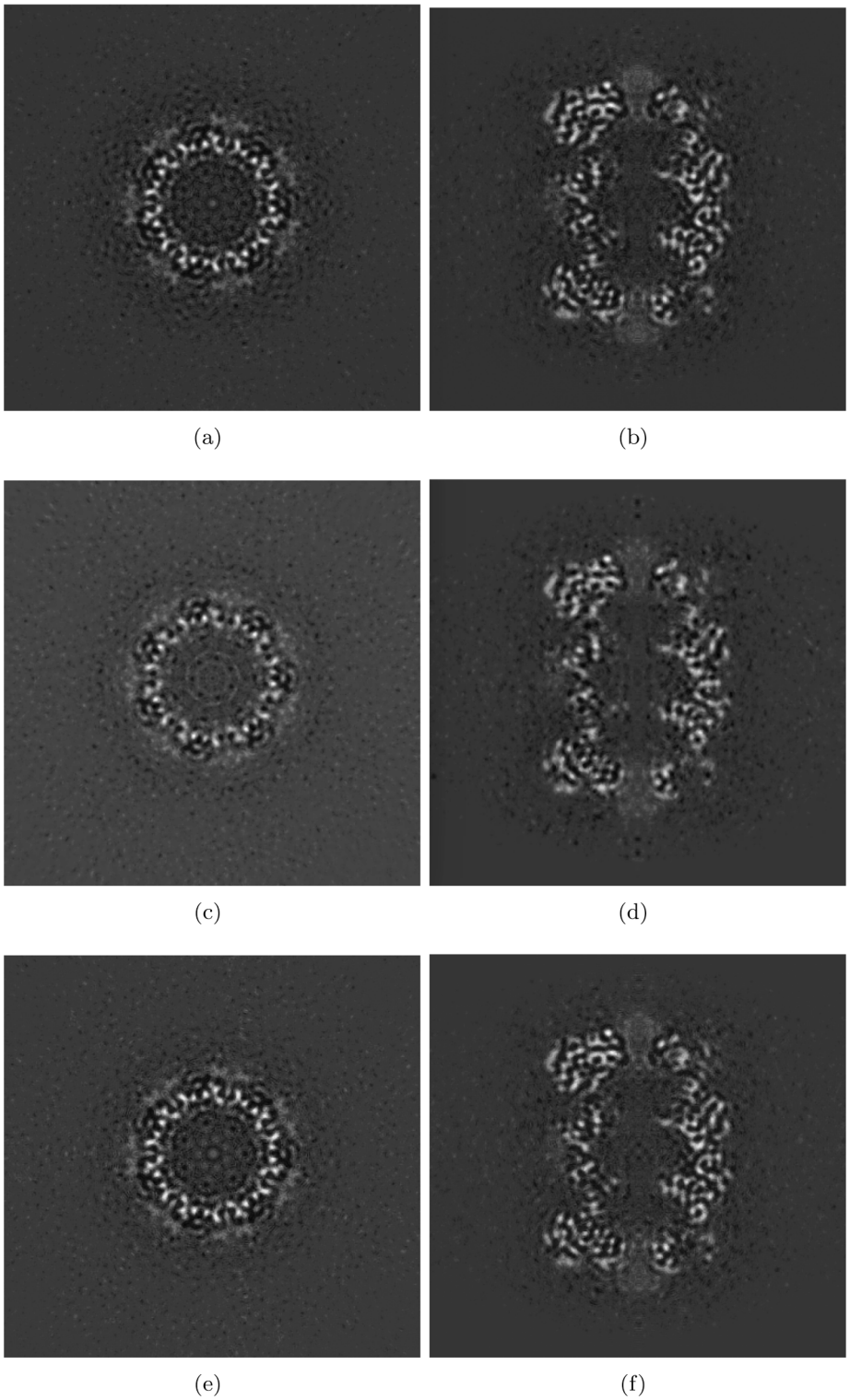


Fig. 12. Central slices of the reconstructed 3D map for *proteasome*. (a) and (b) Z-axis and Y-axis with DeepAlign (2.9 Å). (c) and (d) Z-axis and Y-axis with Xmipp Highres (3.3 Å). (e) and (f) Z-axis and Y-axis with DeepAlign consensus tool (2.7 Å).

previous examples.

During the training, the average loss obtained was 0.06 with an accuracy of 0.94. However, there were three regions in which the loss was around 0.2 and the accuracy was not better than 0.8. As only three regions presented this behavior, we decide to allow 5 regions per image. In this way, we tried to solve the slight uncertainty introduced because of those three regions with worse accuracy.

We used 7 GPUs to run DeepAlign with these data. The time required to train one region was, on average, 9 h, so to train the 42 regions we needed 54 h. The prediction time was 3 h, and the final alignment required 20 min, on average, per region, so a total of 2 h were dedicated to this step. The entire process, taking into account some additional steps, took approximately 2 days and a half. These times are higher than the ones shown in the previous examples, but here we are working with

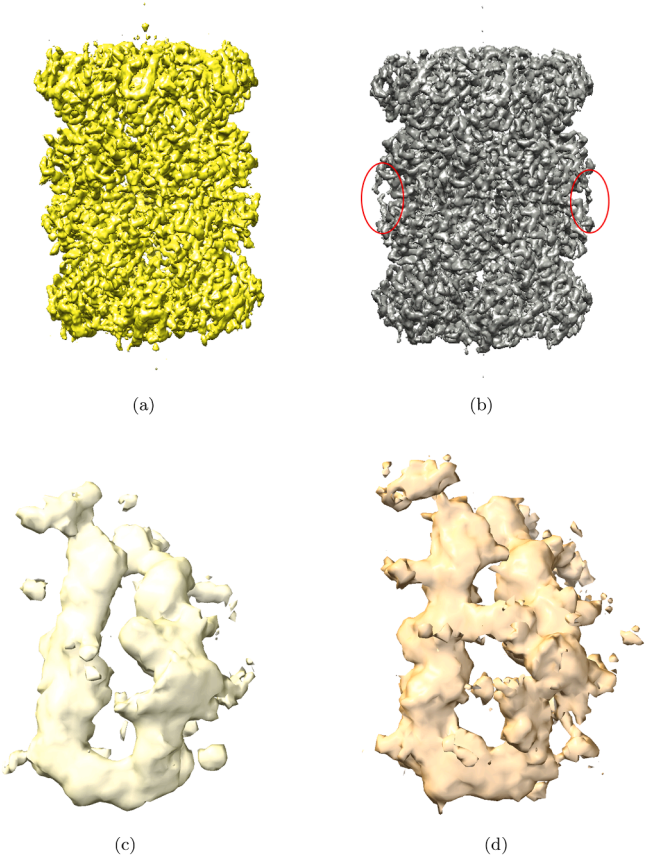


Fig. 13. 3D reconstructed maps for *proteasome*. (a) and (b) Whole 3D maps reconstructed by Xmipp Highres and DeepAlign, respectively. (c) and (d) Zoom in an area with two representative helices for Xmipp Highres and DeepAlign, respectively.

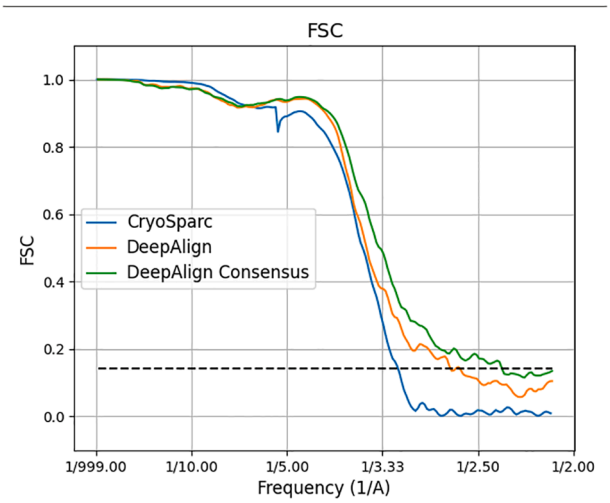


Fig. 14. FSC curves obtained for *SARS-CoV-2 Spike*. CryoSparc 3.1 Å, DeepAlign 2.7 Å, and DeepAlign consensus 2.4 Å are compared.

bigger and noisy images which required more time to train.

Here, we want to compare the best results obtained after a whole processing using CryoSparc with the possibility of using that information in DeepAlign to make one extra step of alignment and check if we are able to improve the previous solution.

The obtained results are shown in Figs. 14–17. Fig. 14 shows the FSC curves obtained for the whole processing with CryoSparc, one more step

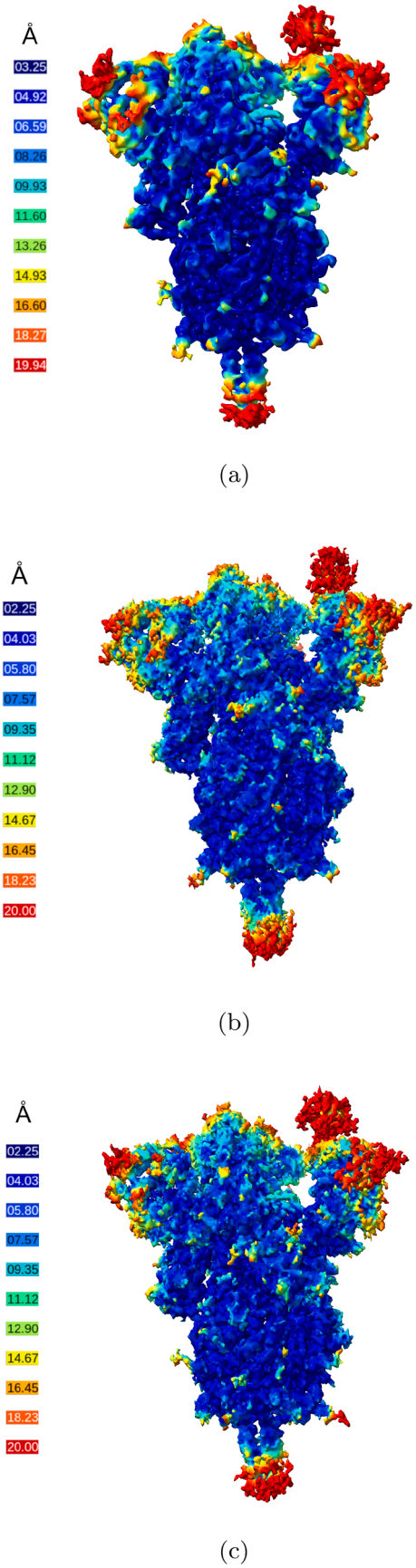


Fig. 15. Local resolution of the reconstructed 3D maps for *SARS-CoV-2 Spike*. (a) CryoSparc, (b) DeepAlign, and (c) DeepAlign consensus.

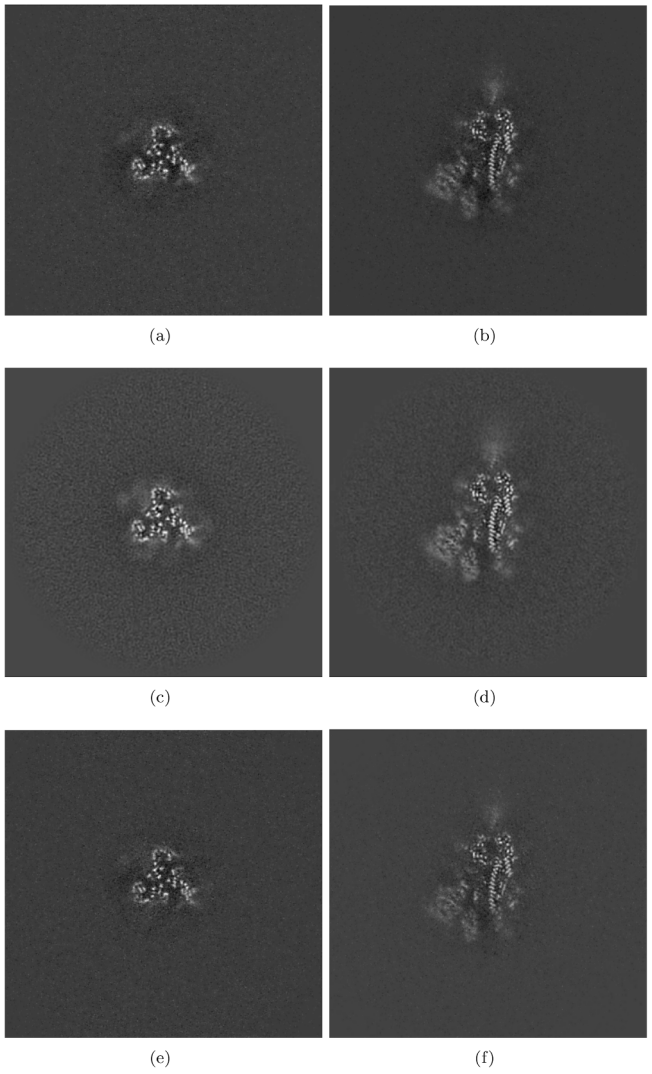


Fig. 16. Central slices of the reconstructed 3D map for *SARS-CoV-2 Spike*. (a) and (b) Z-axis and Y-axis with DeepAlign (2.7 Å). (c) and (d) Z-axis and Y-axis with CryoSparc (3.1 Å). (e) and (f) Z-axis and Y-axis with DeepAlign consensus tool (2.4 Å).

of DeepAlign, and the consensus tool considering both methods (using the subset of images in which the disagreement was less than 10°). The obtained resolution values were 3.1 Å for CryoSparc, 2.7 Å for DeepAlign, and 2.4 Å for DeepAlign consensus tool. The FSC curves were very similar, but it can be highlighted that DeepAlign was able to obtain a flatter curve in the range from 6 to 4 Å, and the fall of the curves in the higher frequencies is softer compared to the one obtained with CryoSparc. The local resolution analysis obtained with Monores is presented in Fig. 15 and it shows a very similar behaviour between all the compared methods, but the best resolution achieved with DeepAlign for some voxels was lower (2.25 Å), than that of CryoSparc (3.25 Å).

Fig. 16 and Fig. 17 show how the reconstructed map can benefit from using DeepAlign.

Fig. 16 shows some particular slices of the 3D map. We can see that the main parts of the structure are clearly represented in the three maps. However, the halo surrounded the density is reduced with DeepAlign and even more with the consensus. This halo is mainly due to particles with wrong angular assignments, as several particles showing not concordant parts of the macromolecule could contribute to the same

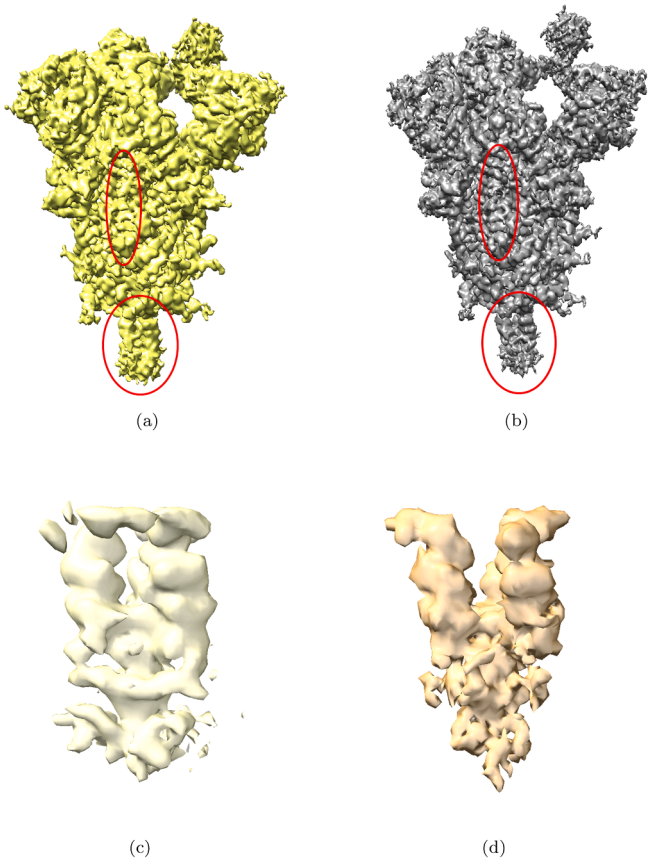


Fig. 17. 3D reconstructed maps for *SARS-CoV-2 Spike*. (a) and (b) Whole 3D maps reconstructed by CryoSparc and DeepAlign, respectively. (c) and (d) Zoom in a specific area showing several helices for CryoSparc and DeepAlign, respectively.

projection direction. This is an advantage of using DeepAlign, which was able to obtain better alignment. This is even more clear in the consensus results, as with this tool we selected only the subset of particle images where DeepAling and CryoSparc agreed in the angular assignment, which reduced the number of images from 36,558 to 17,207 (more than a 50% of reduction).

Finally, Fig. 17 shows a 3D representation of the reconstructed maps. (a) and (b) parts show the whole 3D map for CryoSparc and DeepAlign consensus, respectively. Some areas are slightly better defined and sharper in the reconstruction obtained with DeepAlign, which can be seen in the areas surrounded by a red circle. Part (c) and (d) of the figure show a zoomed area where several helices are located showing similar level of detail, DeepAlign was able to improve in the upper part of the helices but generating more noise in the lower part.

4. Conclusions

In this work, we have presented a new method to carry out the 3D alignment of particle images to obtain a 3D reconstructed map. This work is one of the first in the field using deep learning as baseline technique to obtain the alignment parameters for every image. Specifically, the whole 3D space is divided into small non-overlapping regions. In every one of them, a classifier based on CNNs is used to decide if an image comes from that region or not. Within the region, the final alignment parameters are obtained using an alignment method based on correlation. The CNNs have a light complexity, enough to be able to learn the classification problem, but keeping it as low as possible to

maintain low the total computational burden of the method. Moreover, this method is optimized to run on several GPUs, alleviating greatly the training time, which is the most consuming time step in the whole process.

The method was tested with three structures and compared with several 3D alignment approaches in the literature. The experiments have shown that this proposal is able to obtain competitive results compared to that in the state-of-the-art and generates 3D reconstructed maps with well-defined features and resolutions. In addition, the computational time to use our method is quite reasonable, as the training time is bounded and the workload can be distributed between multiple GPUs.

It is noteworthy that the deep learning basis of DeepAlign is different from the ones in other state-of-the-art approaches based on maximizing probability functions. We can expect that methods with different basis will give rise to different 3D reconstructions (different local minima in the solution space). DeepAlign, which is based on CNNs that have proven to be very robust in image processing tasks, could give us better angular assignments, as the results presented in this work seem to point out.

We have also demonstrated the usefulness of the consensus tool, which selects only the particle images that were aligned with similar parameters by several alignment procedures. Our experiments show that this tool can be very useful to further improve the reconstructed 3D maps. The consensus tool is taking advantage of using alignment parameters obtained with methods with different basis, and this can be done thanks to the development of DeepAlign.

As future work, we plan to manage 3D heterogeneity following the deep learning approach established in this work. Thus, we expect to be able to generate several 3D maps representing the different conformations present in the sample, deciding not only the alignment of the particle images but also the 3D class.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors would like to acknowledge economical support from: Spanish Ministry of Science and Innovation through Grant PID2019-104757RB-I00 (AEI/FEDER, UE), Comunidad Autónoma de Madrid through Grant S2017/BMD-3817, Instituto de Salud Carlos III through Grant PT17/0009/0010 (ISCIII-SGEFI/ ERDF), and European Union (EU) through Grants INSTRUCT - ULTRA (INFRADEV-03–2016-2017, Proposal: 731005), EOSC - Synergy (EINFRA-EOSC-5, Proposal: 857647), and iNEXT - Discovery (Proposal: 871037). The project that gave rise to these results received the support of a fellowship from 'la Caixa' Foundation (ID 100010434). The fellowship code is LCF/BQ/DI18/11660021. This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 713673.

References

Campbell, Melody G. and Veesler, David and Cheng, Anchi and Potter, Clinton S. and Carragher, Bridget, 2.8 A resolution reconstruction of the Thermoplasma acidophilum 20S proteasome using cryo-electron microscopy., eLife 4 (2015) e06380.

Chen, M., Dai, W., Sun, S.Y., He, D.J.C.Y., Schmid, M.F., Chiu, W., Ludtke, S.J., 2017. Convolutional neural networks for automated annotation of cellular cryo-electron tomograms. Nat. Methods 14, 983–985.

Chollet, F. et al., 2015. Keras, https://github.com/fchollet/keras.

de la Rosa-Trevín, J.M., Otón, J., Marabini, R., Zaldívar, A., Vargas, J., Carazo, J.M., Sorzano, C.O.S., 2013. Xmipp 3.0: an improved software suite for image processing in electron microscopy. J. Struct. Biol. 184 (2), 321–328.

de la Rosa-Trevín, J.M., Quintana, A., Del Cano, L., Zaldívar, A., Foche, I., Gutiérrez, J., Gómez-Blanco, J., Burguet-Castell, J., Cuenca-Alba, J., Abrishami, V., Vargas, J., Otón, J., Sharov, G., Vilas, J.L., Navas, J., Conesa, P., Kazemi, M., Marabini, R., Sorzano, C.O.S., Carazo, J.M., 2016. Scipion: A software framework toward integration, reproducibility and validation in 3d electron microscopy. J. Struct. Biol. 195, 93–99.

Elmlund, H., Elmlund, D., Bengio, S., 2013. Prime: probabilistic initial 3D model generation for single-particle cryo-electron microscopy. Structure 21 (8), 1299–1306.

Gupta, H., McCann, M.T., Donati, L., Unser, M., 2020. Cryogan: A new reconstruction paradigm for single-particle cryo-em via deep adversarial learning. bioRxiv 2020 (03), 2020.03.20.001016.

Henderson, R., 1992. Image contrast in high-resolution electron microscopy of biological macromolecules: TMV in ice. Ultramicroscopy 46, 1–18.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. In: Comment Published as a conference paper at the 3rd International Conference for Learning Representations. San Diego arXiv:1412.6980.

Li, R., Si, D., Zeng, T., Ji, S., He, J., 2016. Deep convolutional neural networks for detecting secondary structures in protein density maps from cryo-electron microscopy. In: 2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 41–46.

Melero, R., C.O.S. Sorzano, B. Foster, J.L. Vilas, M. Martinez, M. Marabini, E. Ramirez-Aportela, R. Sanchez-Garcia, D. Herreros, L. del Cano, P. Losana, Y. Fonsea-Reyna, P. Conesa, D. Wrapp, P. Chacon, J.S. McLellan, H.D. Tagare, J.M. Carazo, Continuous flexibility analysis of sars-cov-2 spike prefusion structures, IUCrJ (in press).

Murshudov, G.N., Skubák, P., Lebedev, A.A., Pannu, N.S., Steiner, R.A., Nicholls, R.A., Winn, M.D., Long, F., Vagin, A.A., 2011. REFMAC5 for the refinement of macromolecular crystal structures. Acta Crystallographica Section D 67 (4), 355–367.

Nogales, E., 2016. The development of cryo-EM into a mainstream structural biology technique. Nat. Methods 13 (1), 24–27.

Penczek, P., Radermacher, M., Frank, J., 1992. Three-dimensional reconstruction of single particles embedded in ice. Ultramicroscopy 40, 33–53.

Penczek, P.A., Grasucci, R.A., Frank, J., 1994. The ribosome at improved resolution: New techniques for merging and orientation refinement in 3D cryo-electron microscopy of biological particles. Ultramicroscopy 53, 251–270.

Punjani, A., Rubinstein, J.L., Fleet, D.J., Brubaker, M.A., 2017. cryoSPARC: algorithms for rapid unsupervised cryo-EM structure determination. Nat. Methods 14, 290–296.

Rolnick, D., Veit, A., Belongie, S., Shavit, N., 2018. Deep learning is robust to massive label noise.

Sanchez-Garcia, R., Segura, J., Maluenda, D., Carazo, J.M., Sorzano, C.O.S., 2018. Deep Consensus, a deep learning-based approach for particle pruning in cryo-electron microscopy. IUCrJ 5 (6), 854–865.

Scheres, S.H.W., 2012. A Bayesian view on cryo-EM structure determination. J. Mol. Biol. 415 (2), 406–418.

Scheres, S.H.W., Valle, M., Núñez, R., Sorzano, C.O.S., Marabini, R., Herman, G.T., Carazo, J.M., 2005. Maximum-likelihood multi-reference refinement for electron microscopy images. J. Mol. Biol. 348, 139–149.

Scheres, S.H.W., Gao, H., Valle, M., Herman, G.T., Eggermont, P.P.B., Frank, J., Carazo, J.M., 2007. Disentangling conformational states of macromolecules in 3d-em through likelihood optimization. Nat. Methods 4 (1), 27–29.

Sigworth, F.J., 1998. A Maximum-Likelihood approach to single-particle image refinement. J. Struct. Biol. 122, 328–339.

Sorzano, C.O.S., Vargas, J., de la Rosa-Trevín, J.M., Otón, J., Álvarez-Cabrera, A.L., Abrishami, V., Sesmero, E., Marabini, R., Carazo, J.M., 2015. A statistical approach to the initial volume problem in single particle analysis by electron microscopy. J. Struct. Biol. 189 (3), 213–219.

Sorzano, C., Vargas, J., Vilas, J., Jimenez-Moreno, A., Mota, J., Majtner, T., Maluenda, D., Martinez, M., Sanchez-Garcia, R., Segura, J., Oton, J., Melero, R., del Cano, L., Conesa, P., Gomez-Blanco, J., Rancel, Y., Marabini, R., Carazo, J., 2018. Swarm optimization as a consensus technique for Electron Microscopy Initial Volume. Appl. Anal. Optim. 2 (2), 299–313.

Sorzano, C., Vargas, J., de la Rosa-Trevín, J., Jiménez, A., Maluenda, D., Melero, R., Martínez, M., Ramírez-Aportela, E., Conesa, P., Vilas, J., Marabini, R., Carazo, J., 2018. A new algorithm for high-resolution reconstruction of single particles by electron microscopy. J. Struct. Biol. 204 (2), 329–337.

Vargas, J., Álvarez-Cabrera, A.L., Marabini, R., Carazo, J.M., Sorzano, C.O.S., 2014. Efficient initial volume determination from electron microscopy images of single particles. Bioinformatics 30, 2891–2898.

Vilas, J.L., Gómez-Blanco, J., Conesa, P., Melero, R., Miguel de la Rosa-Trevín, J., Otón, J., Cuenca, J., Marabini, R., Carazo, J.M., Vargas, J., Sorzano, C.O.S., 2018. Monores: Automatic and accurate estimation of local resolution for electron microscopy maps. Structure 26 (2), 337–344.e4.

Wagner, T., Merino, F., Stabrin, M., Moriya, T., Gatsogiannis, C., Raunser, S., 2018. Sphire-cryolo: A fast and well-centering automated particle picker for cryo-em, bioRxiv. 356584.

Wang, F., Gong, H., Liu, G., Li, M., Yan, C., Xia, T., Li, X., Zeng, J., 2016. DeepPicker: A deep learning approach for fully automated particle picking in cryo-EM. J. Struct. Biol. 195 (3), 325–336.

Wong, Wilson and Bai, Xiao-chen and Brown, Alan and Fernandez, Israel S and Hanssen, Eric and Condron, Melanie and Tan, Yan Hong and Baum, Jake and Scheres, Sjors HW, Cryo-EM structure of the Plasmodium falciparum 80S ribosome bound to the anti-protozoan drug emetine, eLife 3 (2014) e03080.

Zhang, J., Wang, Z., Chen, Y., Han, R., Liu, Z., Sun, F., Zhang, F., 2019. PIXER: an automated particle-selection method based on segmentation using a deep neural network. BMC Bioinformatics 20, 41.

Zhong, E.D., Bepler, T., Davis, J.H., Berger, B., 2019. Reconstructing continuous distributions of 3d protein structure from cryo-em images. arXiv:1909.05215.
Zhong, E.D., Bepler, T., Berger, B., Davis, J.H., 2020. CryoDRGN: Reconstruction of heterogeneous structures from cryo-electron micrographs using neural networks. bioRxiv 2020 (03), 2020.03.27.003871.
Zhu, Y., Ouyang, Q., Mao, Y., 2017. A deep convolutional neural network approach to single-particle recognition in cryo-electron microscopy. BMC Bioinformatics 18, 348.