Deep Learning Improves Macromolecule Identification in 3D Cellular Cryo-Electron Tomograms

E. Moebel[1], A. Martinez-Sanchez[2,3,4], L. Lamm[5,6], R.D. Righetto[5], W. Wietrzynski[5], S. Albert[7], D. Larivière[8], E. Fourmentin[8], S. Pfeffer[7,9], J. Ortiz[7,10], W. Baumeister[7], T. Peng[6], B.D. Engel[5,11,*], C. Kervrann[1,*]

[1] Serpico Project-Team, Centre Inria Rennes-Bretagne Atlantique and CNRS-UMR 144, Inria, CNRS, Institut Curie, PSL Research University, Campus Universitaire de Beaulieu, 35 042 Rennes Cedex, France

[2] Department of Computer Science, Faculty of Sciences, University of Oviedo, Calvo Sotelo s/n, 33071 Oviedo, Spain
[3] Health Research Institute of Asturias (ISPA), Avenida Hospital Universitario s/n, 33011 Oviedo, Spain
[4] Institute of Neuropathology, Cluster of Excellence "Multiscale Bioimaging: from Molecular Machines to Networks of Excitable Cells" (MBExC), University of Göttingen, Göttingen, Germany

[5] Helmholtz Pioneer Campus, Helmholtz Zentrum München, Ingolstädter Landstraße 1, 85764 Neuherberg, Germany
[6] Helmholtz AI, Helmholtz Zentrum München, Ingolstädter Landstraße 1, 85764 Neuherberg, Germany

[7] Max Planck Institute of Biochemistry, Am Klopferspitz 18, 82152 Martinsried, Germany

[8] Fourmentin-Guilbert Scientific Foundation, 2 Avenue du Pavé Neuf, 93 160 Noisy-le-Grand, France

[9] Zentrum für Molekulare Biologie der Universität Heidelberg, Im Neuenheimer Feld 282, 69120 Heidelberg, Germany

[10] Ernst Ruska-Centre, Wilhelm-Johnen-Straße, 52425 Jülich, Germany

[11] Department of Chemistry, Technical University of Munich, Lichtenbergstraße 4, 85748 Garching, Germany

Abstract

Cryogenic electron tomography (cryo-ET) visualizes the 3D spatial distribution of macromolecules at nanometer resolution inside native cells. However, automated identification of macromolecules inside cellular tomograms is challenged by noise and reconstruction artifacts, as well as the presence of many molecular species in the crowded volumes. Here, we present DeepFinder, a computational procedure that uses artificial neural networks to simultaneously localize multiple classes of macromolecules. Once trained, the inference stage of DeepFinder is faster than template matching and performs better than other competitive deep learning methods at identifying macromolecules of various sizes in both synthetic and experimental datasets. On cellular cryo-ET data, DeepFinder localized membrane-bound and cytosolic ribosomes (∼3.2 MDa), Rubisco (∼560 kDa soluble complex), and photosystem II (∼550 kDa membrane complex) with an accuracy comparable to expert-supervised ground truth annotations. DeepFinder is therefore a promising algorithm for the semi-automated analysis of a wide range of molecular targets in cellular tomograms.

[1]

[1] ∗ Corresponding authors: charles.kervrann@inria.fr, ben.engel@helmholtz-muenchen.de

Deep learning improves macromolecule identification in 3D cellular cryo-electron tomograms

E. Moebel, A. Martinez-Sanchez, L. Lamm, R.D. Righetto, W. Wietrzynski, S. Albert, D. Larivière, E. Fourmentin, S. Pfeffer, J. Ortiz, W. Baumeister, T. Peng, B.D. Engel, C. Kervrann

Cryogenic electron tomography (cryo-ET) visualizes the 3D spatial distribution of macromolecules at nanometer resolution inside native cells. However, automated identification of macromolecules inside cellular tomograms is challenged by noise and reconstruction artifacts, as well as the presence of many molecular species in the crowded volumes. Here, we present DeepFinder, a computational procedure that uses artificial neural networks to simultaneously localize multiple classes of macromolecules. Once trained, the inference stage of DeepFinder is faster than template matching and performs better than other competitive deep learning methods at identifying macromolecules of various sizes in both synthetic and experimental datasets. On cellular cryo-ET data, DeepFinder localized membrane-bound and cytosolic ribosomes ($\sim$3.2 MDa), Rubisco ($\sim$560 kDa soluble complex), and photosystem II ($\sim$550 kDa membrane complex) with an accuracy comparable to expert-supervised ground truth annotations. DeepFinder is therefore a promising algorithm for the semi-automated analysis of a wide range of molecular targets in cellular tomograms.

Cryogenic electron tomography (cryo-ET) can provide new insights into molecular organization and interactions by producing 3D views of the native cellular interior at sufficient resolution to identify macromolecules. Unlike fluorescence microscopy, cryo-ET lacks specific markers, so the structures of the macromolecules themselves must be used for identification. However, cryo-ET has an advantage over fluorescence microscopy in that it visualizes everything in the cell, not just the tagged molecules. With enough resolution and powerful computational tools for structural identification, it has the potential to build a complete molecular atlas of the cell.

Cryo-ET data is generated by the following steps. First, the samples are vitrified in order to preserve both the native structures and spatial distribution of macromolecules inside the cells. For many cells, a thinning step is required, which can be accomplished with focused ion beam milling [1]. Subsequently, the thin biological material is loaded into the transmission electron microscope for acquisition of a tilt-series, which serves as input to 3D reconstruction algorithms [2;3;4]. During tilt-series acquisition, the specimen is rotated around an axis perpendicular to the electron beam and imaged from multiple perspectives. The tilt-series must be acquired with low electron dose because frozen biological material is easily damaged by the electron beam. Unlike the heavy metal contrasting agents used in conventional electron microscopy, the organic molecules found in frozen cells have low contrast against the water background. Combined with the limited electron dose used for imaging, cryo-ET data have very low signal-to-noise ratios. Furthermore, due the geometry of the EM grid and increasing thickness of the sample at high tilts, tilt-series are typically restricted to ± 60 degrees. As a result, the reconstructed tomograms suffer from a wedge of missing information in Fourier space. This missing wedge artifact causes anisotropic resolution in the 3D volumes, with delocalization of densities along the Z-direction, as well as loss of information in the X–Y plane along the direction perpendicular to the tilt axis [5;6]. The missing wedge and low signal-to-noise ratio, combined with the highly crowded environment of the cell, pose important challenges to the identification of macromolecules in cellular tomograms.

One well established method for localizing macromolecules in cryo-ET data is template matching [7], where a low-resolution template depicting the macromolecule of interest is comprehensively scanned through the tomogram. High cross-correlation scores indicate potential particle positions, from which subvolumes are extracted for downstream averaging procedures. While template matching is relatively efficient for localizing large complexes such as ribosomes, it is necessary to apply a series of iterative searching, alignment, and classification steps to identify smaller complexes [8]. Additional difficulties arise when template matching is used to localize several macromolecular species that are structurally similar, or to differentiate specific states of the same macromolecular species (for example, membrane-bound vs. cytosolic ribosomes). Template matching is applied to separately localize all macromolecules of a single species (mono-class procedure). Nevertheless, dedicated classification steps [9;10] are required to differentiate true particles from false positives, as well as to subdivide these particles into structurally distinct sub-classes. Classification remains a challenging problem in cryo-ET if the number of considered classes is high. Currently, complex and time-consuming processing chains are routinely used to localize macromolecules in cellular volumes, with a single template matching or subtomogram classification round typically taking several hours of computation on specialized CPU clusters. As a result, this whole procedure is most often used to analyze only a few classes of particles in the same volume. However, cryo-electron tomograms contain many more macromolecular species embedded within the crowded cellular environment, hidden by noise and reconstruction artifacts. To address this challenging issue, powerful new pattern recognition techniques are required.

In this paper, we describe a deep learning-based framework to quickly identify macromolecules in cryo-electron tomograms. Deep learning [11] is revolutionizing various fields of data processing, including computer vision [12], image classification [13], and segmentation [14]. In bioimage analysis, convolutional neural networks (CNN) have produced spectacular results [15;16], including in super-resolution microscopy [17] and in fluorescence microscopy image denoising [18]. Briefly, a CNN is defined as an architecture composed of successive connected neuron layers. Applying the layers sequentially enables the network to progressively compute high-level features, which results in a hierarchical or multiscale representation of the data. For example, in human face recognition, the first layers typically encode basic features such as image contours/edges and textures. This allows the next layers to gradually capture more complex shapes (for example, eyes, ears), and object ensembles (for example, faces). These multiscale representations are learned from the data and can be generated faster than conventional feature extraction.

CNNs have recently been investigated for learning high-level generic features in electron microscopy. Several algorithms based on deep learning techniques have been developed for 2D particle pick-

ing in single-particle cryo-EM[19], including DeepPicker[20], AutoCryoPicker[21], crYOLO[22], Topaz[23], and Warp[24]. In cellular cryo-ET, the algorithm proposed by Chen *et al.*, was implemented for supervised segmentation of 2D slices from 3D volumes[25], but it was not designed to handle complex environments (for example, crowded cells) and requires an additional classification step to achieve satisfactory results. A promising 3D processing approach for the supervised classification of subtomograms has been proposed in Che *et al.*[26]. However, this algorithm requires that the molecular complexes of interest have already been found by another method. To overcome the limitations of the aforementioned approaches[20;25;26], we propose a 3D deep learning method to identify macromolecules within crowded native cellular environments. Our DeepFinder algorithm can handle multiple macromolecular species in one pass. We demonstrate that this improves the performance of CNNs in 3D cryo-ET, identifying small particles that template matching struggles to detect. Moreover, complex and time-consuming post-processing steps are no longer required to produce reliable results. We also show that DeepFinder is flexible and can be efficiently combined with template matching to improve localization sensitivity on crowded cellular cryo-tomograms.

Results

Overview of our 3D deep learning-based approach. We present DeepFinder, an algorithm based on 3D CNNs that, in one pass, can robustly localize macromolecules of several different species, with various sizes and shapes, within cryo-tomograms. The algorithm is built upon a multi-class network (Methods and Extended Data Fig. 1b), based on a U-Net architecture[27], as illustrated in Extended Data Fig. 1c. It consists of a training stage (Stage I) and an analysis (or inference) stage (Stage II) (see details in Methods and Fig. 1a). DeepFinder's Stage I is a supervised approach requiring expert-user inputs, whereas Stage II is nearly unsupervised, as opposed to template matching, which requires more manual input:

- In the training stage (Stage I), DeepFinder converts the input 3D coordinates of macromolecules supplied by the experts into voxel-wise annotations (Methods), avoiding the cumbersome and time-consuming manual annotation of voxels. As converting the positions of macromolecules into voxel-wise annotations is not a trivial task, we propose two approaches with different levels of computing complexity. The first approach exploits the segmentation maps built from well-delineated shapes of macromolecules estimated by a subtomogram averaging procedure[28]. In the second approach, the shapes of macromolecules are approximated by 3D spheres (Fig. 2). The sphere-based representation is appealing because information about the shapes and orientations of macromolecules is not needed, and time-consuming subtomogram averaging steps are avoided. Both approaches are made available in the DeepFinder software to enable optimization of performance and speed.
- The analysis stage is a two-step procedure. In the first step, the trained 3D CNN-based model is used to classify the tomogram voxels into different categories or sub-categories of macromolecules. In the second step, a clustering algorithm is applied to aggregate voxels into groups and determine the location of particles (gravity center of clusters) in the volume (Fig. 1b). The particles are further used for structure determination through subtomogram averaging.

DeepFinder is a free, open-source program implemented in Python with an accessible graphical user interface (GUI) (Extended Data Fig. 2). In the following sections, we demonstrate that this program successfully identifies macromolecules of varying shape and molecular weight in both artificial and cellular cryo-ET datasets. Furthermore, we show that handling several molecular species simultaneously with a multi-class strategy is especially beneficial for identifying small macromolecules.

DeepFinder identifies multiple macromolecular species at once. To test the performance of DeepFinder relative to competitive methods, we performed experiments on the SHREC'19 dataset (Dataset #1), which is composed of 10 synthetic tomograms, each containing 12 different classes of macromolecules spanning a range of sizes[29]. These macromolecules have been categorized by size into 4 groups (large, medium, small, tiny) by the organizers. The number of macromolecules per class and per tomogram varies slightly ($\sim$200 particles per class and per tomogram).

In the SHREC'19 challenge, seven different methods (including DeepFinder) were compared to evaluate the performance of deep learning techniques on challenging synthetic 3D cryo-electron tomograms. Participants had access to 9 out of 10 ground truth tomograms, whereafter the algorithms were tested on the final unseen tomogram. DeepFinder achieved the best global F_1-score (a standard metric for the precision and accuracy of detection, see evaluation in Methods) among all competing algorithms[29]. The closest competitor to DeepFinder was the DoG-CB3D method[26], which separates the localization and classification tasks. In Fig. 3a, we report the results obtained on the SHREC'19 dataset with DeepFinder, template matching, and DoG-CB3D. Unlike DoG-CB3D, DeepFinder outperformed template matching for all macromolecular species in the benchmark dataset. The template matching results were obtained with the PyTOM software[30] (Methods). Performance is correlated with macromolecular size (Fig. 3a). This is especially true for template matching, where the smallest complexes had F_1-scores close to 0. Even so, template matching is still a competitive method for detecting the largest macromolecules. On the SHREC'20 dataset[31], DeepFinder was ranked second, with essentially the same algorithm from the previous year (Extended Data Fig. 3). It is worth noting that the top algorithm, U-Net Multi-task Cascade (UMC), requires noise-free images for training[31]. This approach is therefore not applicable on real cellular datasets, although there has been substantial progress on using deep learning to denoise tomograms[23;24;32].

Furthermore, we evaluated the performance of the multi-class network versus 12 mono-class versions of our network. As shown in Fig. 3b and Extended Data Fig. 3b, the binary networks have a similar performance when compared to multi-class networks for larger macromolecules (Protein Data Bank (PDB) entries: 4b4t, 4d8q), but perform worse or fail for the smaller ones (PDB 1s3x, 3gl1, 3h84 and 3qm1). Overall, the multi-class training improves classification accuracy over binary classification. These results also suggest that smaller macromolecules and under-represented classes can benefit from features learned from classes of larger macromolecules.

Finally, it is worth noting that the sphere-based representation was able to provide good scores for all particle sizes, as shown in Fig. 3c and Extended Data Fig. 3c (except the smallest particle, PDB 1s3x in Fig. 3c). The explanation for this good performance is two-fold: first, small macromolecules tend to resemble sphere-like structures. Second, in the case of larger macromolecules, the "label noise" (that is, presence of label errors in the training data) induced by the sphere-based representation is compensated by the high quantity of labeled voxels in the segmentation map (see Rolnick *et al.*[33] for discussion). In summary, the computationally cheap sphere-based representation is recommended for faster processing, provided that the target macromolecule shape is not too topologically different from a sphere (for example, a tube-like structure).

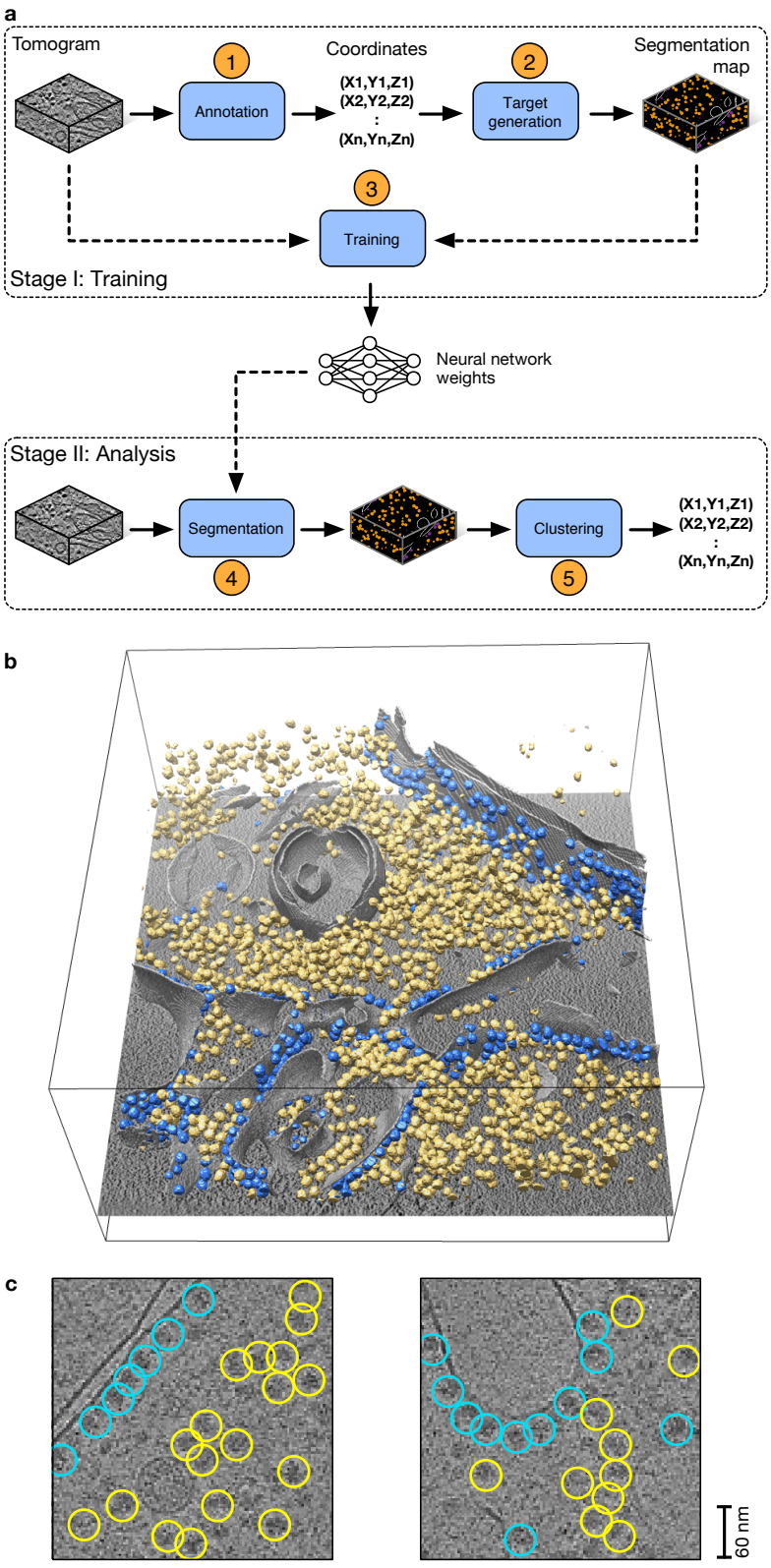


Fig. 1: Overview of DeepFinder. a, The DeepFinder workflow consists of a training stage (Stage I) and an analysis (or inference) stage (Stage II). These two stages correspond to five steps (represented by blue boxes) to locate macromolecular complexes within crowded cells. **b,** Ribosome localization with DeepFinder in a cryo-electron tomogram of a *C. reinhardtii* cell. Tomographic slice with superimposed segmented cell membrane (gray) and ribosomes classified with respect to their binding states: membrane-bound (blue) and cytosolic (yellow). **c,** Tomographic slices showing coordinates of detected ribosomes (colors correspond to **b,**). The positions and classes were determined by analyzing the segmentation map shown in **b**. This analysis used 48 tomograms for training, 1 tomogram for validation, and 8 tomograms for testing. Scale bar, 60 nm.

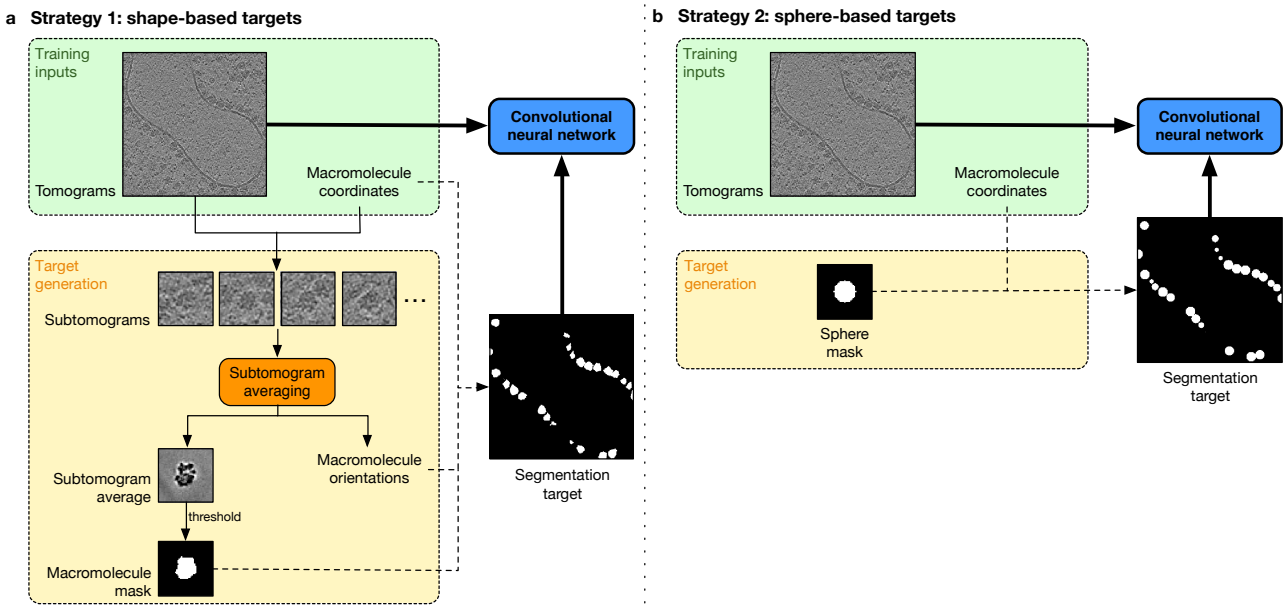


Fig. 2: Target generation strategies for training. a, In strategy 1, voxel-wise annotations are obtained from position-wise annotations using a subtomogram averaging procedure. Subtomogram averaging is a registration algorithm that produces higher resolution structures by averaging thousands of aligned subvolumes containing the molecular species. Subtomogram volumes are extracted at the annotated positions, aligned and finally averaged. The alignment procedure outputs the object orientations, while the averaging process provides a density map of the macromolecule with greatly reduced noise and missing wedge artifacts. From this density map, it is possible to create a binary mask of the macromolecule by thresholding the averaged subtomogram. The resulting mask is pasted into an empty volume at each annotated position with the estimated 3D orientation, and the resulting volume with well-delineated macromolecules is then used as a target to train the 3D CNN parameters. **b,** In strategy 2, the macromolecule masks are approximated and replaced by spherical masks. Hence, the subtomogram averaging procedure is bypassed and the target generation process is faster. However, the training targets contain more *"label noise"*.

More annotation is required to localize small macromolecules. We analyzed the influence of the training set size on the performance of our method (Extended Data Figs. 4 and 5). It is desirable to achieve good performance even if the training set is small, as manual annotation of data is time consuming and requires considerable effort in 3D imaging. As shown in Extended Data Fig. 5c (synthetic SHREC'19 dataset), the performance of DeepFinder correlates with macromolecule size. For large and medium macromolecules, the performance is remarkably stable, even when using only one training tomogram (206 macromolecules per class). For small and tiny macromolecules, the drop in performance is more notable, and larger training sets are required to get higher score values. This is not surprising since the number of labeled voxels associated with large macromolecules is high in the segmentation maps. Accordingly, less annotated macromolecules are needed.

To carefully examine the performance of DeepFinder on experimental cellular cryo-tomograms, we investigated a range of macromolecular complexes: ribosomes, Rubisco and photosystem II (PSII). These complexes correspond to different particle sizes and have different quantities of available expert annotations. Ribosomes (large) and Rubisco complexes (small) are relatively well localized with template matching, yielding abundant annotations for training and allowing a direct comparison of DeepFinder with the template matching results. Detecting PSII is a more challenging task, given that in addition to being a small membrane complex, the annotations were manually assigned and thus limited in quantity. We used the shape-based approach for these tests, as it offers reduced "label noise" and performs slightly better on average (F_1 score) than the sphere-based approach for detecting macromolecules with molecular weights greater than 200 kDa (Fig. 3c and Extended Data Fig. 3c). The calculation of F_1-scores usually relies on the availability of ground truth positions, which are provided by synthetic datasets (SHREC'19 and SHREC'20), but are generally not available with experimental data. Instead of ground truth, the F_1 scores reported below were calculated with respect to the annotations provided by the experts, which used a combination of template matching, visual curation, subtomogram averaging and classification. An F_1-score of 1 indicates that the performance of DeepFinder is identical to the approach supervised by the experts.

DeepFinder imitates expert annotations of large macromolecules. First, we explored the ability of our method to localize two ribosomal states (membrane-bound and cytosolic ribosomes), and we compared the results to those obtained with template matching (Methods). In this study, we used a cryo-ET dataset (Dataset #2, Methods) composed of 57 tomograms of *Chlamydomonas reinhardtii* cells and annotations of membrane-bound 80S ribosomes (∼3.2 MDa) performed by experts[34]. In our experiments, we considered four classes to better detect the target macromolecules: membrane-bound ribosome (*mb-ribo*), cytosolic ribosome (*ct-ribo*), membrane, and background. The expert annotations contained positions of 8,795 *mb-ribos*. The training datasets corresponding to the *ct-ribo* and membrane classes were obtained by using semi-automated computational tools, without careful expert supervision (Methods). The classes were highly imbalanced since the background class represents 99.5% of training voxels within the tomogram. As this scenario is common in cryo-ET, the training procedure (especially the generation of batches) needs to be adapted by integrating techniques such as re-sampling (Methods). Figure 1b-c illustrates an example of DeepFinder's four-class segmentation and provides insight into the spatial distribution of two ribosomal states. DeepFinder detected *mb-ribos* that were not annotated by the experts.

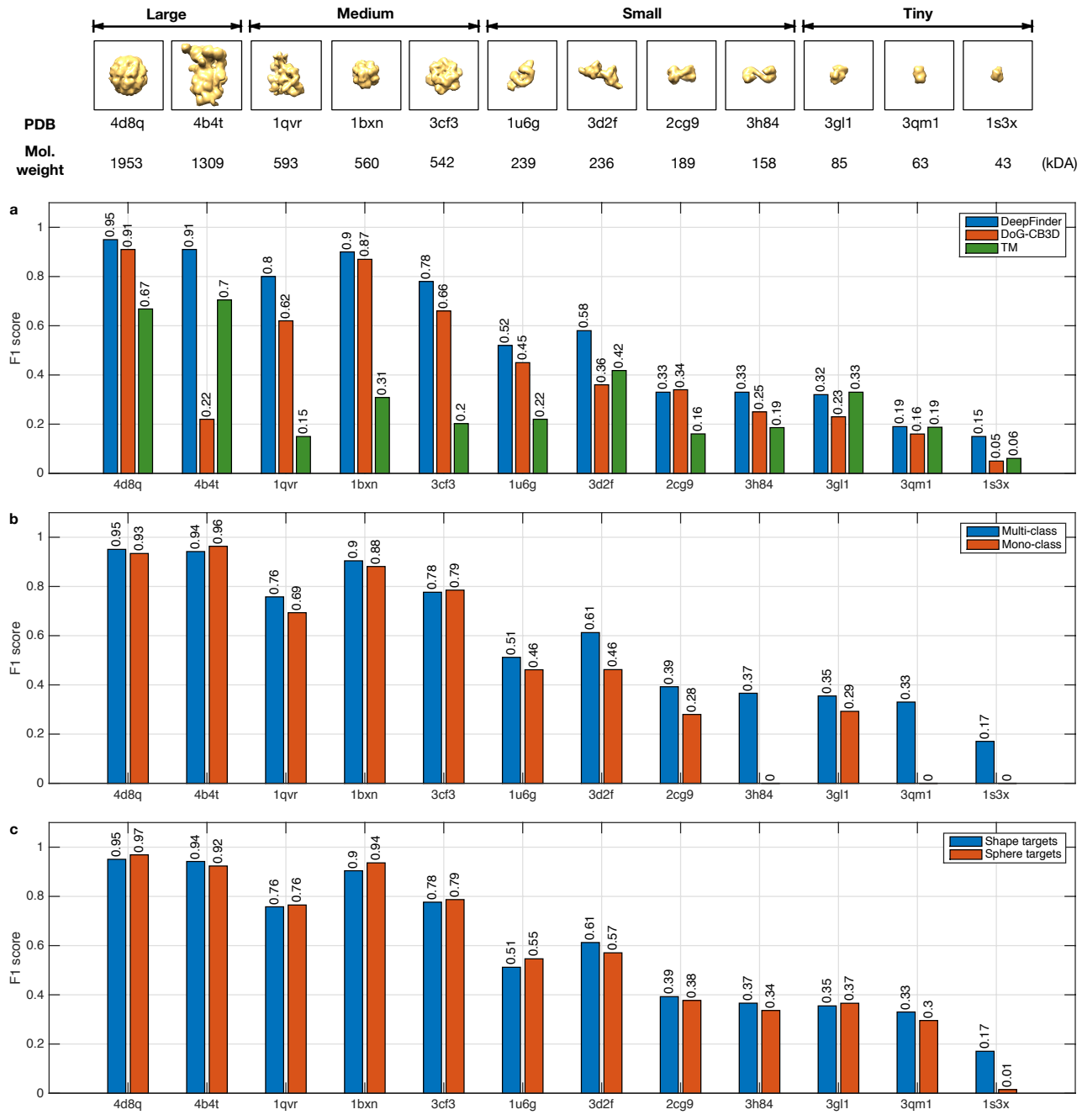


Fig. 3: Analysis of algorithm performance on the synthetic dataset (SHREC'19 challenge). a, Performance (F_1-score) of DeepFinder, DoG-CB3D and template matching algorithms and ability of algorithms to discriminate between 12 species of macromolecules. The most challenging macromolecules in SHREC'19 (ref. [29]) and SHREC'20 (ref. [31]) are in the tiny and small macromolecule classes (1s3x (human Hsp70 ATPase domain, ∼43 kDa), 3qm1 (*Lactobacillus johnsoni* Cinnsmoyl esterase LJ0536 S106A mutant in complex Ethylferulate Form II, ∼63 kDa), and 3gl1 (*Saccharomyces cerevisiae* ATPase domain of Ssb1 chaperone, ∼85 kDa) and the small macromolecules (2cg9 (Hsp90-Sba1 closed chaperone complex, ∼189 kDa), 3h84 (GET3, ∼158 kDa), 1u6g (Cand1-Cul1-Roc1 complex, ∼239 kDa), 3d2f (complex of Sse1p and Hsp70, ∼236 kDa)). The medium and large macromolecules are 1bxn (Rubisco from *Alcaligenes eutrophus*, ∼560 kDa), 1qvr (ClpB, ∼297 kDa), 3cf3 (P97/vcp in complex with ADP, ∼542 kDa), 4d8q (eukaryotic chaperonin TRiC/CCT, ∼1.95 MDa), 4b4t (26S proteasome, ∼1.31 MDa). The highest (best) possible value of an F_1-score is 1.0 and the lowest (worst) possible value is 0. The scores of template matching were provided by the SHREC'19 challenge organizers (Utrecht University, Department of Information and Computing Sciences and Department of Chemistry). **b,** Performance of DeepFinder implemented as a multi-class network architecture and as an architecture made of 12 binary networks. These two architectures differ only by the number of output neurons. **c,** Influence of the training target generation method ("shapes" versus "spheres"). In the case of "shapes", the exact shapes of the macromolecules have been used to annotate the tomograms. In the case of "spheres", the shape and the orientation of macromolecules are not needed to generate the training targets. This analysis used 8 tomograms for training, 1 tomogram for validation, and 1 tomogram for testing.

Several of these particles were false positives, but after a detailed inspection (Supplementary Note 1), we confirmed that many others are ribosomes that were missed or discarded during the annotation process.

Analysis of score-maps. In Fig. 4, it is clear that the score-map from template matching (Fig. 4d) is much noisier than the score-maps generated by DeepFinder (Fig. 4b-c). Template matching produced high cross-correlation scores at ribosome locations but also at false-positive locations containing other high contrast structures (for example, cell membrane in Fig. 4d). Consequently, template matching is mainly used to crudely discard the voxels with no important structural information. In the next step, the experts must apply post-processing techniques to separate the desired macromolecules from the false positives. Compared to template matching, DeepFinder produces less noisy score-maps, with meaningful score values in well-localized blobs.

Furthermore, DeepFinder allows accurate simultaneous localization of two ribosomal states in cellular cryo-ET data, as illustrated in Fig. 1b. The *mb-ribos* (in blue) are primarily located close to cell membranes, whereas the *ct-ribos* (in yellow) are mainly located in the cytosol.

Analysis of structural resolution. DeepFinder achieves 3D structural resolution comparable to the expert annotations, as determined through subtomogram averaging. Using the test set tomograms, we computed subtomogram averages for each ribosomal state (*ct-ribos* and *mb-ribos*) (Fig. 4f) (Supplementary Note 1) by applying the same registration procedure (rotational matching[35]) to the subvolumes from DeepFinder and the expert annotations. The resulting density maps of *mb-ribos* are composed of two regions: the ribosome density and the membrane density. As expected, the membrane density computed from the set of *mb-ribo* particles found by DeepFinder is well defined, while there is no membrane density in the average computed from the set of *ct-ribo* particles.

We assessed the resolution by calculating the Fourier shell correlation (FSC) for the subtomogram averages of each ribosomal state (Fig. 4e). For both ribosomal states, the resolution was comparable between the expert annotations and DeepFinder (23 Å and 24 Å, respectively). That said, the goal of this experiment was not to produce the highest resolution average (which could require >10,000 particles), but rather to test how well DeepFinder can distinguish subpopulations, even when particle number is limited.

Additionally, we examined the nature of the particles that were detected by DeepFinder but not annotated by the experts, i.e. $S_{DF} \setminus S_E$, where S_E and S_{DF} denote the sets obtained by experts and DeepFinder, respectively. We carefully analyzed the difference between sets S_{DF} and S_E and considered the two following hypotheses:

H_1 : The set S_{DF} contains too many false positives.

H_2 : The set contains particles with a low imaging quality (due to noise and blur), suggesting that our method found supplementary true *mb-ribo* particles that were missed or discarded during the supervised annotation.

We aligned and averaged *mb-ribo* particles in the set $S_{DF} \setminus S_E$ and observed that the corresponding density map depicts a ribosome structure (Supplementary Note 1). Thus, hypothesis H_1 is unlikely. As detailed in Supplementary Note 1, DeepFinder finds additional membrane-bound ribosomes, missed or discarded by experts (hypothesis H_2). Therefore, it appears that the number of actual *mb-ribos* is higher than expected: in our test set, we detected $+20.5\%$ *mb-ribos* when compared to the S_E set. This result confirms the benefit of combining several analysis methods in cryo-EM[36]. The consensus between the expert processing chain and DeepFinder decreases the overall false negative rate and generates a larger set $S_E \cup S_{DF}$ of membrane-bound ribosomes. The extended set of particles is probably less homogeneous, as it contains supplementary particles which are noisier. Adding this set of particles to the average may blur the result and therefore degrade the structural resolution. Nevertheless, these particles not found by template matching should not be discarded *a priori*, as they provide a more complete picture of the cellular environment. In summary, DeepFinder, as well as human experts (who used template matching followed by CPCA classification with visual inspection), does miss some true positives. Our results illustrate the collaborative strength of the two imperfect particle picking procedures. Even though we demonstrated on the SHREC'19 and SHREC'20 datasets that DeepFinder has better precision, recall and F_1-score than template matching (see also the *Recall*, *Precision* and F_1-scores curves of DeepFinder (F_1-score of 0.86) and template matching (F_1-score of 0.50) in Extended Data Fig. 6 (Dataset #2)), the global performance can be boosted further with consensus analysis among competing methods[36].

DeepFinder finds small macromolecules in cellular tomograms. On cellular cryo-electron tomograms, the F_1-score is not significantly worse if we use an *mb-ribo* training dataset consisting of one-fifth (1,408 *mb-ribos*) of the complete annotated dataset (8,795 *mb-ribos*) (Fig. 5c). This suggests that DeepFinder does not require a large quantity of annotations for localizing large macromolecules, but more annotations are necessary to localize small macromolecular species. We demonstrate this principle below on smaller macromolecules, including both soluble and membrane proteins.

To test how well DeepFinder detects small particles in cellular tomograms, we conducted experiments on two challenging datasets depicting pyrenoids (Dataset #3) (Fig. 5a-c) and thylakoid membranes (Dataset #4) (Fig. 5e-h) within *C. reinhardtii* cells[38;40]. In Dataset #3, DeepFinder was able to identify Rubisco holoenzymes with remarkable performance (Fig. 5a-b). The achieved F_1-score (0.83) is similar to what was obtained for *mb-ribos* (0.86) (Dataset #2), even though the molecular mass of Rubisco (∼560 kDa) is much lower than that of the 80S ribosome (∼3.2 MDa). Subtomogram averaging performed with the top 30,000 DeepFinder hits yielded a map resolved to 15 Å. Template matching of the top 30,000 hits following CPCA classification post-processing produced a nearly identical map using the same alignment and averaging pipeline but required more computing time and extensive classification, as described previously[9]. DeepFinder inference is faster and yields a "clean" set of particles without any additional post-processing. The two averages are also practically identical to the previously published map at 16.5 Å (EMD-3694). This result suggests that DeepFinder is able to identify small macromolecular species with similar performance to that seen for larger complexes, provided that the training set is large enough.

In our experiments on Dataset #4, DeepFinder was able to detect photosystem II (PSII) dimers embedded within native thylakoid membranes (Fig. 5e-h). We trained DeepFinder on three tomograms using four particle classes that were manually annotated[38]: PSII dimers (∼550 kDa), cytochrome b6f dimers (∼220 kDa), unknown densities, and background. A very important distinction is that, while the Rubisco annotations were generated by supervised template matching and classification, the PSII ground truth was assigned completely manually through "membranogram" visual inspection[38] (Fig. 5g) and served further as a reference to compute the F_1 scores. In Supplementary Table 2, we report the F_1-scores for template matching and DeepFinder (when using the mono-class or multi-class strategies) to detect PSII in the unseen test tomogram; the multi-class approach produced better results than the mono-class approach and template matching. PSII complexes were detected quite accurately in some

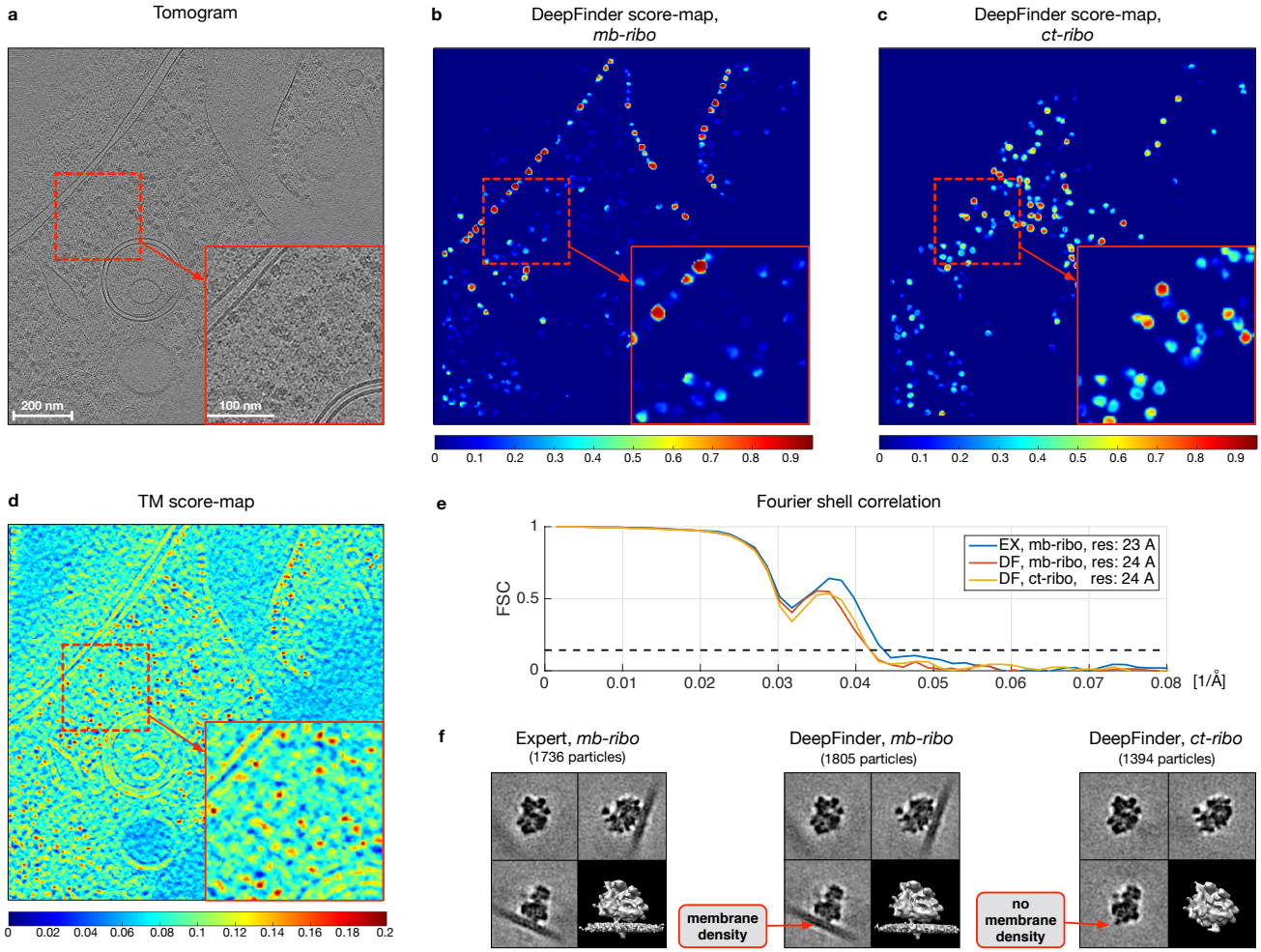


Fig. 4: Comparison of score-maps obtained with template matching and DeepFinder, and analysis of structural resolution through subtomogram averaging (Dataset #2). **a,** Experimental cryo-electron tomogram depicting a *C. reinhardtii* cell (Dataset #2, Methods). **b-c,** Score-maps of *mb-ribos* and *ct-ribo* particles with DeepFinder. **d,** Score-map of ribosomes with template matching (TM). **e,** FSC curves for each subtomogram average with estimated resolutions. The resolution corresponding to expert annotations of *mb-ribo* particles (23 Å) is comparable to the resolution obtained with *mb-ribo* particles localized with DeepFinder (24 Å). **f,** Subtomogram averages obtained from expert annotations (left) and from particles localized with DeepFinder (middle: *mb-ribos*, right: *ct-ribos*). The averages were low-pass filtered to 63 Å at the beginning of each iteration of fast rotational matching (FRM) alignment and were all generated with the same alignment procedures and parameter settings. This analysis used 48 tomograms for training, 1 tomogram for validation, and 8 tomograms for testing and averaging.

membranes (as illustrated in Fig. 5f-g), which was confirmed by the high F_1-score (0.737 for membrane 4 in Supplementary Table 2), whereas in other membranes they were not. Overall, the mono-class approach with only PSII performed worse than the multi-class setting, in particular with respect to Recall (0.505 versus 0.638 in Supplementary Table 2). Template matching yielded lower values (F_1 scores, Recall, Precision, Supplementary Table 2) than those obtained with the mono-class and multi-class DeepFinder strategies. In contrast to the precision of DeepFinder, template matching produced high cross-correlation scores along most of the thylakoid membranes and struggled to distinguish PSII complexes from the surrounding membrane density (Fig. 5f-g). The DeepFinder results were limited by the small training dataset size (298 PSII particles from 18 membranes) as well as the variable quality of the experimental data. Membranes where PSII complexes were poorly resolved had the most missed picks and lowest F_1-scores. Nevertheless, the results illustrated in Fig. 5e-h are promising, and we believe that adding more particles to the training sets (as exemplified in the Dataset #3 experiment) will further improve the capability of DeepFinder to reliably detect PSII and other membrane proteins.

Collectively, these experiments on soluble and membrane complexes of different sizes consistently indicate that the multi-class approach is a flexible and robust method to identify target macromolecules. Moreover, spurious components such as gold beads or ice surface contamination can be typically included in the "background" negative class during the training task in order to reduce false positives, as illustrated in Extended Data Fig. 7.

Discussion

Currently, only a handful of different molecular species have been analyzed in cellular cryo-tomograms. These volumes contain rich information on the whole proteome, but many macromolecules are hardly discernible from noise and artifacts. To realize the potential of visual proteomics [41] and study the interaction between several macromolecular species [42], better structure recognition algorithms

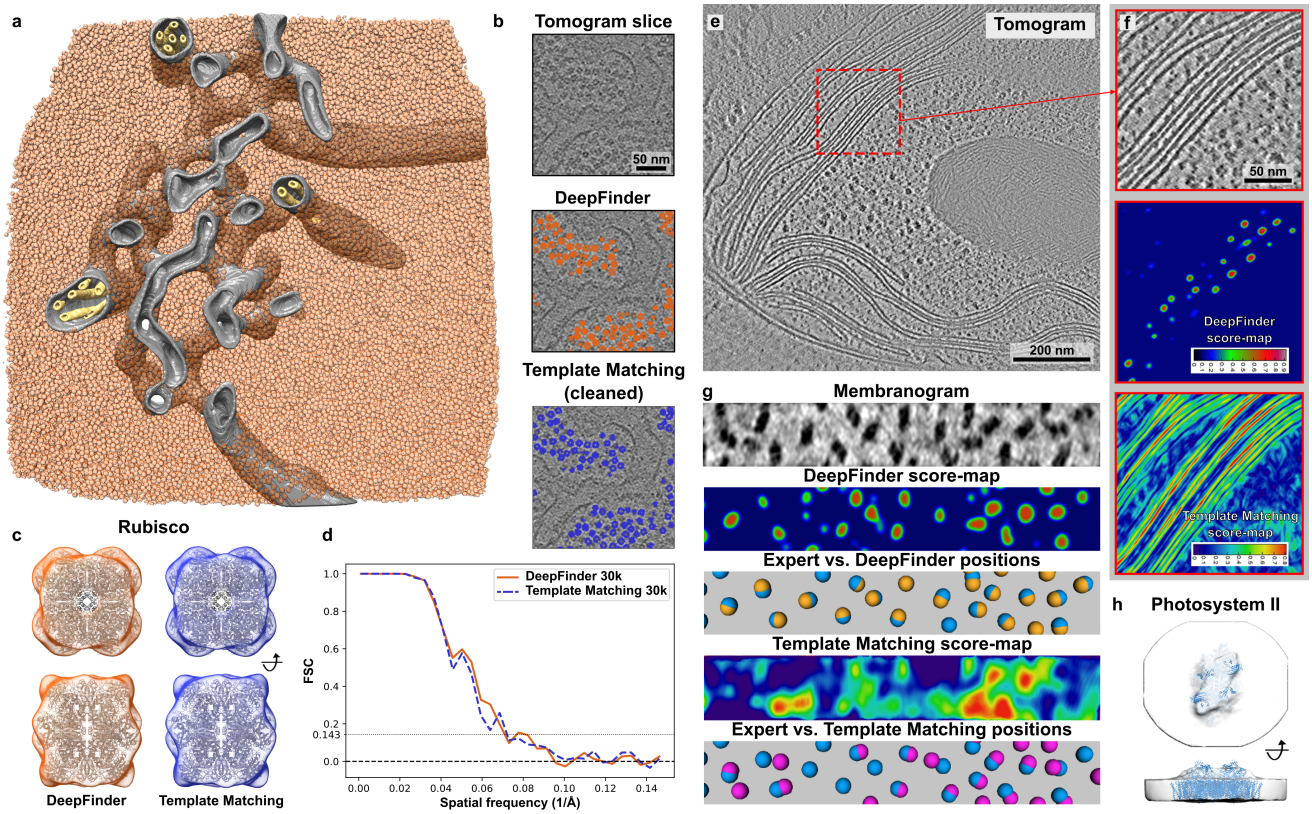


Fig. 5: DeepFinder localizes small macromolecules in cellular tomograms. a, Segmentation of a tomogram of the native Chlamydomonas pyrenoid (Dataset #3). Rubisco holoenzymes segmented by DeepFinder are displayed in transparent orange. For visualization purposes, hand-segmented pyrenoid tubule membranes (gray and yellow) have been added. **b,** Tomogram slice with corresponding DeepFinder segmentation and template matching (cleaned during post-processing with CPCA classification[9]). **c,** Rubisco averages (top and side views) obtained with PyTOM software[30] from the top 30,000 DeepFinder and template matching particles, with a Rubisco molecular structure (PDB 7JN4, ref.[37]) fitted into the volumes. **d,** Fourier shell correlation (FSC) curves from the averages obtained with 30,000 particles each. Both methods yield a final resolution of 15 Å (FSC $>$ 0.143) (Methods). This analysis used 4 tomograms for training, and 1 tomograms for testing and averaging. **e,** Slice through a tomogram of the chloroplast within an intact Chlamydomonas cell (Dataset #4). Scale bar, 200 nm. **f,** Zoomed-in view of the tomogram slice, with the associated score-maps produced by DeepFinder and template matching. Scale bar, 50 nm. **g,** The membranogram is a visualization approach where densities from the tomogram are projected onto the surface of a segmented thylakoid membrane, resulting in a topological view of the membrane surface[38]. Below are membranograms of the associated DeepFinder and template matching score-maps and a visualization comparing the positions of particles found by the expert (blue) versus DeepFinder (yellow) and template matching (pink). **h,** Subtomogram average (white) of PSII complexes, calculated from 246 particles identified by DeepFinder on a single test tomogram, fit with a molecular structure of PSII (PDB 6KAD, ref.[39], blue). This analysis used 3 tomograms for training, 1 subtomogram for validation and 1 tomogram for testing and averaging.

are required. Deep learning offers a powerful framework to address this challenge. Therefore, we have developed DeepFinder to efficiently identify molecular complexes with variable shapes and molecular weights. We showed quantitatively that this deep-learning based method can generate molecular averages with resolutions that are similar to averages obtained by expert curation, as demonstrated for ribosomes in Fig. 4 and Rubisco complexes in Fig. 5. Moreover, once trained, DeepFinder is relatively fast compared to the common template matching and subtomogram classification pipeline (Table 1). Larger datasets can be processed in one day with DeepFinder, while several macromolecular species are simultaneously identified.

When evaluating performance on experimental data, it is important to consider that annotations are a subset of the unknown ground truth. This is why no algorithm should produce an F_1-score of 1 with respect to the expert annotations. Indeed, although most of the expert annotations can be considered as true positives, there will almost certainly be a percentage of false positives and false negatives in the labeled dataset. An F_1-score of 1 during training would indicate that the network is rather overfitting to the expert's annotations, instead of really learning the structures of particles. Therefore, for a properly working detection algorithm, an F_1-score of 1 is not to be expected. In particular, a low precision value does not necessarily mean that the picked particles are false positives.

To cope with the missing wedge artifact (that is, delocalization along Z-direction), the usual template matching procedure uses an isotropic template, along with missing wedge-constrained cross-correlation as a criterion. In contrast, DeepFinder is trained to predict a segmentation mask depicting particle shapes that are not affected by the missing wedge (Fig. 2). In fact, DeepFinder applies a series of linear (convolution) and nonlinear operators (max-pooling, ReLU activation). All these cascaded operations are performed in real space and progressively modify the results in Fourier space, thereby reducing the influence of the missing wedge. A way to interpret the cascade mechanism is to consider the simple case of generating a binary

		Computing time	Hardware
Stage I: Training	Target generation, sphere-based	3 sec per tomogram	2.9 GHz Intel Core i7
	Target generation, shape-based	35 min (low resolution subtomo. averaging) + 3 sec per tomogram	32-core CPU cluster
	Training (until convergence)	15 hours	Tesla K80 GPU
Stage II: Analysis	Step 1: segmentation	20 min per tomogram	Tesla K80 GPU
	Step 2: clustering	5 min per tomogram	2.9G Hz Intel Core i7

Table 1: Computing times obtained for processing Dataset #2, composed of 57 tomograms of size $928 \times 928 \times 464$ voxels.

mask from an image or volume. The thresholding applied for binarization is a nonlinear operation. The resulting binary mask will have its corresponding Fourier representation completely filled with information, even if the input data has a missing wedge. Meanwhile, applying the segmentation mask to the input data amounts to multiplying them in real space and is equivalent to applying a convolution operation in Fourier space, hence partially filling the missing wedge. This principle has been used to fill the missing wedge in cryo-ET and electron crystallography[43;44;45]. DeepFinder generates an output like this, except it can have multiple discrete objects in the segmentation. Finally, even if some features are missing or distorted due to the missing wedge, particles can still be recognized based on the remaining intact features of the structures.

It is worth noting that annotating the training set requires some time. However, when processing multiple tomograms (as is normally the case in cryo-ET), the benefits from the much faster inference by DeepFinder (Table 1) outweigh the annotation time. Moreover, one important goal is to achieve generalizability to all kinds of tomograms, which already works to some extent, as we demonstrate by applying our network trained on the Chlamydomonas (algae) data to previously unseen mouse cell tomograms (Extended Data Fig. 8). The generalizability will only get better as new training data is added, decreasing the amount of necessary annotation for new analysis and making DeepFinder even more efficient. In contrast, the long computational time of template matching remains fixed. As it is not possible to precisely quantify the time of the annotation task, which depends on human performance and expertise, we focused on algorithm run-time with given annotations (Table 1 and Methods ("Description of template matching")), as is common in machine learning.

In conclusion, DeepFinder is an efficient new algorithm for cryo-ET image analysis, which we believe will contribute to developing visual proteomics in the coming years. While unsupervised learning holds great promise for the future, the current price to be paid for better results is to resort to supervised machine learning algorithms guided by time-consuming expert annotations. Structural databases including the Protein Data Bank (PDB) and Electron Microscopy Data Bank (EMDB) can be leveraged to help expand the range of molecular species that can be identified in cellular tomograms. Future challenges include developing unsupervised or self-supervised deep learning methods in order to decrease the amount of required training data.

References

[1] Schaffer, M. *et al.* Optimized cryo-focused ion beam sample preparation aimed at in situ structural studies of membrane proteins. *J. Struct. Biol.* **197**, 73–82 (2017).

[2] Frank, J. Approaches to large-scale structures. *Curr. Opin. Struct. Biol.* **5**, 194–201 (1995).

[3] McEwen, B., Renken, C., Marko, M. & Mannella, C. Principles and practice in electron tomography. *Methods Cell Biol.* **89**, 129–168 (2008).

[4] McIntosh, R., Nicastro, D. & Mastronarde, D. New views of cells in 3D: an introduction to electron tomography. *Trends Cell Biol.* **15**, 43–51 (2005).

[5] Nicastro, D., Frangakis, A., Typke, D. & Baumeister, W. Cryo-electron tomography of neurospora mitochondria. *J. Struct. Biol.* **129**, 48–56 (2000).

[6] Guesdon, A., Blestel, S., Kervrann, C. & Chrétien, D. Single versus dual-axis cryo-electron tomography of microtubules assembled in vitro: limits and perspectives. *J. Struct. Biol.* **181**, 169–78 (2013).

[7] Best, C., Nickell, S. & Baumeister, W. Localization of protein complexes by pattern recognition. *Methods Cell Biol.* **2007**, 615–638 (2007).

[8] Albert, S. *et al.* Direct visualization of degradation microcompartments at the ER membrane. *Proc. Natl. Acad. Sci.* **117**, 1069–1080 (2020).

[9] Förster, F., Pruggnaller, S., Seybert, A. & Frangakis, A. S. Classification of cryo-electron sub-tomograms using constrained correlation. *J. Struct. Biol.* **161**, 276–286 (2008).

[10] Martinez-Sanchez, A. *et al.* Template-free detection and classification of membrane-bound complexes in cryo-electron tomograms. *Nat. Methods* **17**, 209–216 (2020).

[11] LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).

[12] LeCun, Y., Kavukcuoglu, K. & Farabet, C. Convolutional networks and applications in vision. In *Proc. IEEE Int. Symp. on Circuits and Systems*, 253–256 (2010).

[13] Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet classification with deep convolutional neural networks. In *Proc. Neural Inf. Processing Systems (NIPS)*, 1–9 (2012).

[14] Long, J., Shelhamer, E. & Darrell, T. Fully convolutional networks for semantic segmentation. In *Proc. Conf. Comput. Vis. Pattern Recognition*, 3431–3440 (2014).

[15] Falk, T. *et al.* U-net – deep learning for cell counting, detection, and morphometry. *Nat. Methods* **16**, 67–70 (2019).

[16] Belthangady, C. & Royer, L. Applications, promises, and pitfalls of deep learning for fluorescence image reconstruction. *Nat. Methods* **16**, 1215–1225 (2019).

[17] Ouyang, W., Aristov, A., Lelek, M., Hao, X. & Zimmer, C. Deep learning massively accelerates super-resolution localization microscopy. *Nat. Biotechnology* **36**, 460–468 (2018).

[18] Weigert, M. *et al.* Content-aware image restoration: pushing the limits of fluorescence microscopy. *Nat. Methods* **12**, 1090–1097 (2018).

[19] Wu, X., Zeng, X., Zhu, Z., Gao, X. & Xu, M. Template-based and template-free approaches in cellular cryo-electron tomography structural pattern mining. *Comp. Biology* **Chapter 11**, 1146–1152 (2019).

[20] Wang, F. *et al.* DeepPicker : A deep learning approach for fully automated particle picking in cryo-EM. *J. Struct. Biol.* **195**, 325–336 (2016).

[21] Al-Azzawi, A., Ouadou, A., Tanner, J. J. & Cheng, J. AutoCryoPicker: an unsupervised learning approach for fully automated single particle picking in cryo-EM images. *BMC Bioinformatics* **20**, 326 (2019).

[22] Wagner, T. *et al.* SPHIRE-crYOLO is a fast and accurate fully automated particle picker for cryo-EM. *Communications Biology* **2**, 218 (2019).

[23] Bepler, T. *et al.* Positive-unlabeled convolutional neural networks for particle picking in cryo-electron micrographs. *Nat. Methods* **16**, 1153–1160 (2019).

[24] Tegunov, D. & Cramer, P. Real-time cryo-electron microscopy data pre-

processing with Warp. *Nat. Methods* **16**, 1146–1152 (2019).

[25] Chen, M. *et al.* Convolutional neural networks for automated annotation of cellular cryo-electron tomograms. *Nat. Methods* **14**, 983–985 (2017).

[26] Che, C. *et al.* Improved deep learning based macromolecules structure classification from electron cryo tomograms. *Mach. Vis. Appl.* **29**, 1227–1236 (2018).

[27] Ronneberger, O., Fischer, P. & Brox, T. U-Net: Convolutional networks for biomedical image segmentation. In *Proc. Med. Image Comput. Comput. Assist. Interv. (MICCAI)*, vol. 9351, 234–241 (2015).

[28] Förster, F. & Hegerl, R. Structure determination *in situ* by averaging of tomograms. *Cell. Electron Microsc.* **79**, 741–767 (2007).

[29] Gubins, I. *et al.* SHREC'19 Track : Classification in cryo-electron tomograms. In *Eurographics Workshop on 3D Object Retrieval*, SHREC – 3D Shape Retrieval Contest (2019), 1–6 (2019). URL `https://www2.projects.science.uu.nl/shrec/cryo-et/2019/`.

[30] Hrabe, T. *et al.* PyTOM: A python-based toolbox for localization of macromolecules in cryo-electron tomograms and subtomogram analysis. *J. Struct. Biol.* **178**, 177–188 (2012).

[31] Gubins, I. *et al.* SHREC 2020: Classification in cryo-electron tomograms. *Computers & Graphics* **91**, 279–289 (2020).

[32] Moebel, E. & Kervrann, C. A Monte Carlo framework for missing wedge restoration and noise removal in cryo-electron tomography. *J. Struct. Biol.; X* **4**, 100013 (2020).

[33] Rolnick, D., Veit, A., Belongie, S. & Shavit, N. Deep learning is robust to massive label noise. *arXiv Prepr.* (2017). `arXiv:1705.10694v2`.

[34] Pfeffer, S. *et al.* Dissecting the molecular organization of the translocon-associated protein complex. *Nat. Communications* **8**, 14516 (2017).

[35] Chen, Y., Pfeffer, S., Hrabe, T., Schuller, J. M. & Förster, F. Fast and accurate reference-free alignment of subtomograms. *J. Struct. Biol.* **182**, 235–245 (2013).

[36] Sanchez-Garcia, R., Segura, J., Maluenda, D., Carazo, J. & Sorzano, C. Deep consensus, a deep learning-based approach for particle pruning in cryo-electron microscopy. *IUCrJ.* **5**, 854–865 (2018).

[37] He, S. *et al.* The structural basis of Rubisco phase separation in the pyrenoid. *Nat Plants* **6**, 1480–1490 (2020).

[38] Wietrzynski, W. *et al.* Charting the native architecture of chlamydomonas thylakoid membranes with single-molecule precision. *eLife* **9**, e53740 (2020).

[39] Sheng, X. *et al.* Structural insight into light harvesting for photosystem II in green algae. *Nat Plants* **5**, 1320–1330 (2019).

[40] Freeman-Rosenzweig, E. *et al.* The eukaryotic co_2-concentrating organelle is liquid-like and exhibits dynamic reorganization. *Cell* **171**, 148–162 (2017).

[41] Förster, F., Han, B. G. & Beck, M. Visual proteomics. *Methods Enzymol.* **483**, 215–243 (2010).

[42] Vendeville, A., Larivière, D. & Fourmentin, E. An inventory of the bacterial macromolecular components and their spatial organization. *FEMS Microbiology Reviews* **35**, 395–414 (2011).

[43] Gipson, B. R. *et al.* Automatic recovery of missing amplitudes and phases in tilt-limited electron crystallography of two-dimensional crystals. *Phys. Rev. E* **84**, 011916 (2011).

[44] Deng, Y. *et al.* ICON: 3D reconstruction with "missing-information" restoration in biological electron tomography. *J. Struct. Biol.* **195**, 100–112 (2016).

[45] Biyani, N. *et al.* Image processing techniques for high-resolution structure determination from badly ordered 2D crystals. *J. Struct. Biol.* **203**, 120–134 (2018).

Methods

Description of DeepFinder algorithm – Stage I (Training). In the training stage (Fig. 1a), DeepFinder parameters are learned from pairs of tomograms and their corresponding voxel-wise annotations. The underlying 3D CNN requires that each voxel is annotated as a member of a given macromolecular species or as background. While voxel-wise class labels are naturally available for synthetic data, this is often not the case for experimental data. In our case, the experts only supplied the 3D coordinates of macromolecules of interest without labeling voxels. In practice, voxel-wise annotation is seldom performed manually in cryo-ET for two reasons: *i)* it is time consuming to individually label each voxel belonging to a 3D macromolecule; *ii)* noise and artifacts in the data make it difficult to accurately distinguish macromolecule borders.

To get voxel-wise annotations, DeepFinder implements two strategies starting from 3D spatial coordinates of the particles:

- *Shape-based annotation:* In this first strategy, the targets are generated by using the structural shape and orientation of each macromolecule. This can be achieved via standard subtomogram averaging procedures [28] (Fig. 2). While this shape-based strategy provides accurate targets, it also induces some "label noise". Indeed, the average shape (as obtained with subtomogram averaging) does not capture all structural variability of a given individual macromolecule.
- *Sphere-based annotation:* In the alternative strategy, 3D spheres are positioned at each annotated location in the tomogram. The sphere radius should be similar to the size of the target macromolecule. As the experts generally know the size of the target macromolecule (for example, $\sim$15 nm radius for ribosomes to $\sim$5 nm radius for small particles like Rubisco), this parameter can be readily input into DeepFinder. This strategy is easier to implement than the shape-based annotation and is appealing in terms of computing time. The main limitation is that spheres may introduce substantially more "label noise" than the shape-based approach. The resulting errors in the annotations hinder the training procedure, and the segmentation network may have suboptimal performance. Nonetheless, it has been shown that CNNs have a natural robustness to moderate levels of "label noise" [33], which is also confirmed in our experiments with DeepFinder.

In our experiments, all results have been obtained by using shape-based annotations, except for the comparison between sphere-based and shape-based annotations on synthetic data (Fig. 3c).

Training datasets and batch generation. Due to memory limitations, it may not be feasible to load the whole tomogram set with the corresponding targets. Instead, for each training iteration, we sample a group of smaller 3D patches to constitute a "batch". The patch size should be large enough to capture sufficient context information. For a macromolecule with a 10 voxel radius, we choose a patch size of $56 \times 56 \times 56\times$ voxels. Our implementation is such that only the current batch is loaded into memory. This helps to handle large datasets with limited computational resources, allowing DeepFinder to be applied on lower-end consumer GPUs with less RAM.

Another problem is the high imbalance between class labels. Because of the small size of the macromolecules compared to the tomogram size, more than 99% of its voxels are annotated as "background". This causes the trained network to be skewed towards the over-represented class. Therefore, we "guide" the sampling procedure by selecting patches only at annotated positions such that each patch contains at least one macromolecule. An additional benefit of sampling only at annotated positions is that the amount of false negatives is reduced (here the negative class is the background). It is indeed common that annotations are not exhaustive. Therefore false "background" labels remain at missed macromolecule positions, contributing to increased "label noise". The proposed patch sampling procedure does not discard all false negatives (for example, a false negative could be neighboring a true positive), but the number should be relatively small at the end.

An additional problem is the imbalance between competing macromolecule classes: for instance, 20 macromolecules of class #1 versus 100 macromolecules of class #2. To reduce this imbalanced class effect, we apply a bootstrapping algorithm (that is, resampling), so that the distribution of the positive classes in a batch will be uniform. This stochastic resampling procedure is effective for sampling the under-represented classes more frequently than the over-represented classes.

It is also common in deep learning to use data augmentation to artificially increase the size of training sets. In our approach, we implement "data augmentation" to each training example by randomly applying a 180° rotation with respect to the microscope tilt-axis. We assume that the input volumes were oriented properly beforehand. Nevertheless, we do not use "mirror" operations or geometric deformations because the macromolecule structure (including its chirality) is the main clue for detection. Also, we do not use random rotations because of the well-determined orientation of missing wedge artifacts, which is preserved when applying 180° rotations with respect to tilt-axis. Finally, we apply random shifts to improve invariance to translations.

Optimization. In our experiments, DeepFinder has been computationally trained with the ADAM algorithm, chosen for its good convergence rate [46], by setting the learning rate to 0.0001, the exponential decay rate to 0.9 for the first moment estimate, and to 0.999 for the second moment estimate. The batch size was set to 25 and the patch size to $56 \times 56 \times 56$ voxels. These values (batch size and patch size) were used to process datasets #1 and #2 and can be revised according to the available GPU capacity (see Table in "Code availability and implementation details"). We chose to set the number of mini-batches to 25 in order to maximally fit into the available GPU in our case. No regularization (for example, L_2 regularizer or "drop out") was needed for processing the datasets. We implemented both the categorical cross-entropy loss (as implemented in Keras) and Dice [47] loss functions. On the synthetic SHREC'19 dataset, Dice loss was able to better localize the smallest macromolecules. However, both loss functions yielded similar results on real cryo-ET data (localization of *mb-ribo* and *ct-ribo* particles).

Description of DeepFinder algorithm – Stage II (Analysis). We describe below the two steps of the DeepFinder analysis stage (Fig. 1a).

Analysis stage (Stage II): Step #1 – Multi-class voxel-wise classification. The objective is to provide a segmentation map for which each voxel is assigned to a class label (representing a macromolecular species). The architecture of DeepFinder is based on U-Net [27], an "encoder-decoder" type of network designed for segmenting images in an end-to-end manner. U-Net is an extension of the fully convolutional network and achieves multi-resolution feature representation and produces high-resolution label maps. The DeepFinder architecture consists of a down-sampling path (that is, encoder) needed to generate global information and an up-sampling path (that is, decoder) used to generate high-resolution outputs [27;48] , that is, local information (Extended Data Fig. 1c). Down-sampling is performed with max-pooling layers (factor 2) and up-sampling with up-convolutions (sometimes called "backward convolution"), which is essentially a trained and nonlinear up-sampling operation. Combining global and local information is performed by concatenating features at different spatial resolutions. The fully convolutional nature of DeepFinder allows one to use various input patch shapes, with the constraint that the patch size must be a multiple of four, because of the two down-sampling stages. Large tomograms can be processed by using an overlap-patch strategy. Unlike Milletari *et al.* [48], DeepFinder is not that "deep", since we found that using more than two down-sampling stages does not improve the segmentation results. Also, we used only $3 \times 3 \times 3$ filter sizes, as described by Simonyan *et al.* [49]. The rationale behind this choice is that two consecutive $3 \times 3 \times 3$ filters mimic a larger $5 \times 5 \times 5$ filter but with fewer parameters. Training is then faster, easier and requires less memory. An important concept in neural architectures is the receptive field of deepest neuron layers. This determines the size of the spatial context to be used to make decisions. Considering a large spatial context is essential to handle macromolecule classes involving interactions with the environment, for instance interactions with cellular membranes. It is established that adding convolutional layers after down-sampling operations is appropriate to enlarge the spatial context [48]. Accordingly, we added two supplementary convolutional layers in the lowest stage of our architecture. In the end, the receptive field size of our network is $48 \times 48 \times 48$ voxels. To complete the description, we use rectified linear units (ReLU) [13] as activation functions for every layer except the last one, which uses a *soft-max* function. While ReLU is a popular choice to tackle non-linearities in the network, the *soft-max* function is necessary in order to interpret the network outputs as pseudo-probabilities for each class. In summary, step #1 is capable of robustly classifying the voxels of the cryo-ET tomogram into N classes with a high accuracy.

Analysis stage (Stage II): Step #2 – Clustering for macromolecule localization. Given the multi-class voxel-wise classification map and classification errors, the objective of step #2 is to estimate the spatial coordinates of each particle of a given macromolecular species. The voxel labels should be spatially clustered into distinct 3D connected components, each cluster corresponding to a unique particle. Because of noise, non-stationarities in the background, and artifacts in the tomogram, step #1 generates isolated labels or very small groups of voxels, as well as groups that contain different labels. Post-processing is then necessary

to aggregate neighboring voxels into 3D connected components and to assign a unique label to each component (or cluster). Clusters that are significantly smaller than the size of target particles are considered as false positives and are discarded. The centroids of meaningful clusters are computed to determine the location of the particles. As the centroid is computed by uniformly averaging the coordinates of cluster voxels, we are able to calculate positions with sub-voxel precision. As several voxel labels can be spatially grouped in a given cluster, the most frequent label is assigned to the detected particle. To address this task, we used the "mean-shift" clustering algorithm [50]. The main advantage of "mean-shift" is that it is controlled by only one parameter, commonly called the bandwidth, which is directly related to the average size of macromolecules. The popular K-means algorithm was not considered further since the number of clusters must be provided as an input parameter by the user.

Evaluation of computing times. Computing times for Stage I and Stage II are reported in Table 1. The inference stage (Stage II) is especially fast when compared to template matching, allowing the analysis of dozens of tomograms per day. As a first example, for processing a tomogram from Dataset #4 (size $928 \times 928 \times 464$ voxels), template matching took 2:08 hours, using a 96-core CPU cluster (see details below in "Description of template matching"), while the DeepFinder inference stage took only 18 min using a Nvidia Quadro RTX 5000 GPU. As a second example, if we consider 10 tomograms of Dataset #2 (same size as above), the computing time is 260 hours (10×26 hours) with template matching, using a 32-core CPU cluster (see details below in "Description of template matching"), while it is 23:20 hours (20 hours (training) + 10×20 min (inference)) with DeepFinder, using a Tesla K80 GPU (Table 1).

These estimations do not take into account the time needed to annotate the data (for DeepFinder) nor the time to generate the template and subsequent subtomogram classification (for template matching). Moreover, as the DeepFinder training expands and becomes more generalized, less training is required to analyze new data, further increasing its speed advantage over template matching. Note that the shape-based target generation takes more computing time than the sphere-based strategy, because it relies on a subtomogram alignment routine (with no classification procedure) performed on downsampled (binned) tomograms (for Dataset #2 we used subtomograms of size $47 \times 47 \times 47$ voxels).

Description of synthetic data (Dataset #1). The synthetic dataset was generated by the cryo-electron microscopy group of Utrecht University for the SHREC'19 challenge [29]. The dataset was created as follows [29]: *i)* tomogram density maps were generated using Protein Data Bank (PDB) structures (identifiers: 1bxn, 1qvr, 1s3x, 1u6g, 2cg9, 3cf3, 3d2f, 3gl1, 3h84, 3qm1, 4b4t and 4d8q); *ii)* a series of projection images was generated with a $\pm 57^\circ$ tilt-range and a 3° tilt-step; *iii)* projection images were degraded by adding noise and contrast transfer function such that the signal-to-noise ratio was 0.02; *iv)* the tomogram was reconstructed using weighted back-projection. The resulting tomograms have a size of $512 \times 512 \times 512$ voxels and a voxel size of 1 nm.

In our experiments, we split the Dataset #1 into training, validation and test sets. The training set was composed of 8 tomograms annotated with a total of 19,956 macromolecules. The validation set was composed of one tomogram annotated with a total of 2,490 macromolecules. The test set was composed of one tomogram annotated with a total of 2,540 macromolecules. The training was performed until convergence, which took 10,000 iterations and 33 hours of computation on a Nvidia Tesla K80 GPU.

Description of experimental cryo-ET data. For Datasets #2, #3 and #4, the tomograms were binned to a voxel size of 13.68 Å with tomogram dimensions of $928 \times 928 \times 464$ voxels. Tilt range was $\pm 60^\circ$ with an increment of 2°.

Experimental 80S ribosome cryo-ET data (Dataset #2). This dataset is composed of 57 tomograms of *C. reinhardtii* cells, and was used for membrane-bound 80S ribosome (*mb-ribo*) annotation. To produce the annotations, the experts used template matching and then filtered out the false positives by applying the CPCA subtomogram classification algorithm [9] and careful visual inspection (see illustration in Extended Data Fig. 1a).

As starting point, Dataset #2 was annotated by localizing 8,795 *mb-ribo* particles under expert supervision. Then, we introduced two additional classes without expert supervision: the cytosolic ribosomes (*ct-ribos*) class and the *membrane* class. Annotations for these two additional classes were obtained by applying semi-automatized computational tools to the tomograms. First, we selected members of the *membrane* class by employing the TomoSegMemTV algorithm, which is dedicated to cell membrane segmentation [51]. Next, we obtained training *ct-ribo* particles by applying template matching and selecting candidates located further than 273.6 Å (the ribosome diameter) from the segmented membranes. The motivation behind adding new classes to the available annotations was twofold: first, we wanted to demonstrate the ability of DeepFinder to localize and identify multiple molecular species on real data with one pass; second, we noticed that the multi-class approach tends to improve the discriminating power of CNN, therefore better finding *mb-ribo* particles in experimental tomograms. When trained with only the *mb-ribo* examples, DeepFinder additionally finds unwanted cytosolic ribosomes. By considering the *ct-ribo* class in the training, we encourage the network to better discriminate both ribosome subclasses (corresponding to binding states). Note that the annotations of membranes and cytosolic ribosomes were obtained without the supervision of an expert. Therefore, more errors are expected for these two classes when compared to the *mb-ribo* examples, which were reliably annotated by experts. In the end, Dataset #2 was annotated with four classes: three positive classes (*mb-ribo, ct-ribo,* and "membrane"), as well as a negative class ("background"). The "background" voxels are those that do not belong to any of the three positive classes.

Dataset #2 was arbitrarily split into training, validation and test sets. The training set was composed of 48 tomograms annotated with 6,834 *mb-ribos* and 6,687 *ct-ribo* particles. The validation set was one tomogram annotated with 222 *mb-ribos* and 254 *ct-ribo* particles. The test set was composed of eight tomograms annotated with 1,736 *mb-ribos* and 2,594 *ct-ribo* particles. Training was carried out using the categorical cross-entropy loss function (similar results were obtained with the Dice loss function). The training was performed for 6,000 iterations and took 20 hours of computation on a Nvidia Tesla K80 GPU (see plots of loss evolution during training in Extended Data Fig. 5a).

Experimental pyrenoid cryo-ET data (Dataset #3). The dataset is composed of five tomograms of the *C. reinhardtii* pyrenoid, which have been annotated with a total of 176,229 Rubisco holoenzymes. The expert annotations were obtained with template matching (excluding hits outside the pyrenoid matrix using a manually segmented mask), followed by subtomogram classification using CPCA clustering [9]. For more details about the dataset and the annotations, see ref. [40].

A validation set was not used because DeepFinder was run with the same network architecture previously fixed. The training set was composed of four tomograms annotated with 129,662 Rubisco complexes. The test set was composed of one tomogram annotated with 46,567 Rubisco complexes. The training was performed for 20,000 iterations and took 35 hours on a Tesla K80 GPU. The loss curve converged after 10,000 iterations, suggesting that only half the time (17.5 hours) would have been sufficient to obtain the presented results.

Experimental thylakoid cryo-ET data (Dataset #4). The dataset is composed of four tomograms of *C. reinhardtii* thylakoid membranes, annotated with a total of 637 photosystem II (PSII) complexes. The expert annotations were obtained by manual particle picking using the "membranogram" approach [38]. For more details about the dataset and the annotations, see ref. [38].

Dataset #4 was split into training, validation and test sets. The training set was composed of three tomograms annotated with 298 PSII complexes. The validation set was a region from one of the training tomograms annotated with 143 PSII complexes. The distance between training and validation particles was large enough to avoid overlap. The test set was composed of one tomogram annotated with 196 PSII complexes. The training was performed for 5,000 iterations and took 3.5 hours of computation on a Nvidia Quadro RTX 5000 GPU.

Ribosome subtomogram averaging. From the test tomograms of Dataset #2, we extracted 1,805 membrane-bound ribosomes and 1,394 cytosolic ribosomes. These numbers were obtained by choosing the cluster size that maximizes the F_1-score (Extended Data Fig. 6). Binary masks were used to focus on regions of interest. Volumes of size $184 \times 184 \times 184$ at a voxel size of 3.42 Å were extracted and reconstructed from the unbinned tilt series using the IMOD program subtomosetup [52]. For subtomogram averaging, five iterations of fast rotational matching (FRM) were carried out in PyTOM [30] using EMD-4145 as an initial reference. The reference was low-pass filtered to 63 Å at the beginning of each iteration. The membrane density of EMD-4145 was manually erased, so that the alignment was focused on the ribosome structure. The same process was applied to obtain the expert subtomogram average in Fig. 4f.

PSII subtomogram averaging. From our test tomogram of Dataset #4, we extracted the 329 picked positions that were close to segmented membranes. For each of these positions, we computed the corresponding normal vector to the membrane. We incorporated this normal vector as a prior on the particles' orientation angles in the first subtomogram averaging step in RELION[53], which we also used for all subsequent averaging tasks. The approach is based on previous averaging protocols[10;54]. Here, we averaged subvolumes of size $90 \times 90 \times 90$ and enforced C10 symmetry[10] in order to only align the membranes in the subvolumes and postpone the refinement of the structures to later steps. Next, having aligned the membranes for all the picked positions, we performed the first refinement step, enforcing C2 symmetry, which is PSII's symmetry group. To clean the PSII picks, we performed subtomogram classification to separate the subvolumes into five classes, which were then further classified into five more classes. This procedure was repeated three times in order to avoid discarding true picks. In the next step, the remaining picks were refined again using C2 symmetry. A final 5-fold classification gave us five classes showing different membrane-bound densities. We merged the classes that were similar to PSII and performed a final refinement step for the remaining 246 positions, yielding the final subtomogram average.

Rubisco subtomogram averaging.

DeepFinder particles: A list containing the coordinates of the top 30,000 hits obtained by DeepFinder on the previously published tomogram of the *C. reinhardtii* pyrenoid matrix[40] (EMD-12749) was used to extract boxes for subtomogram averaging. Volumes comprising $64 \times 64 \times 64$ voxels at a voxel size of 3.42 Å were extracted and reconstructed from the unbinned tilt series (EMPIAR-10694) using the IMOD program subtomosetup[52]. The tilt series was corrected for the contrast transfer function (CTF) by phase flipping in IMOD prior to subtomogram extraction. The previously published subtomogram average of Rubisco (EMD-3694) was low-pass filtered to 40 Å and used as initial reference for the iterative subtomogram aligning and averaging in PyTOM[30]. Using this reference, one iteration of global search was carried out in real space with an angular step of 18° using the PyTOM script `align.py`. Subsequently, four iterations of local real space alignment were carried out with the PyTOM script `GLocalJob.py`, starting with an angular step of 9°. Particles were randomly assigned into two half-sets which were independently refined ("gold standard"). Adaptive alignment resolution and angular sampling were determined automatically based on the Fourier shell correlation[55] (FSC) curve between the two half-maps at each iteration. During refinement, C4 symmetry was imposed. Additional D1 symmetry was imposed to the final half-maps, amounting to the known D4 symmetry of Rubisco. Finally, the half-maps were postprocessed in RELION[53] using a soft mask, yielding a final resolution of 15 Å (FSC > 0.143, ref.[56]) after correcting for artificial correlations induced by the mask[57].

Template matching particles: We extracted subtomograms from the same list of coordinates of the top 30,000 template matching hits that had been used for the previously published subtomogram average of Rubisco (EMD-3694)[40]. Subtomogram extraction, averaging and postprocessing were carried out exactly as described above for the DeepFinder particles. This set of particles yielded a reconstruction at 15 Å (FSC > 0.143).

Description of template matching. The template matching algorithm implemented in the PyTOM[30] toolbox was used to analyze all datasets:

- *Dataset #1:* The template matching results on the synthetic SHREC'19 dataset were provided by courtesy of Utrecht University (Department of Information and Computing Sciences, Department of Chemistry) and were obtained as follows: first, template matching was applied for each class. As the particles have been simulated, we know the true number of candidates for each tomogram: 2,500 particles/12 classes = 211. Then, the top 211 candidates for each class were sequentially selected and extracted, from the largest macromolecules to the smallest ones, in a way that all these candidates do not overlap with the already extracted candidates.
- *Dataset #2:* The template matching was preformed in previously published work[8;34], and the annotations were subsequently provided for our study. For the template, the authors used a low-resolution subtomogram average generated from the dataset, which in turn was obtained de novo using manually selected ribosomes and FRM alignment.
- *Dataset #3:* The template matching was performed in previously published work[40], and the annotations were subsequently provided for our study. For the template, the authors used a Rubisco crystal structure, low-pass filtered to 33 Å. Particles were extracted exhaustively from the masked pyrenoid matrix volume and then cleaned by extensive hierarchical classification. See ref.[40] for more details.
- *Dataset #4:* We used the previously obtained PSII average[38] as a template, low-pass filtered to 30 Å, for an angular search with step of 20°.

The computing times of template matching for PSII and ribosomes (using PyTOM) are given the following table:

	PSII	**Ribosomes**
Template size	$16 \times 16 \times 16$	$40 \times 40 \times 40$
Tomogram size	$928 \times 928 \times 464$	$928 \times 928 \times 464$
Angular step	19.95°	12.85°
Number of orientations	1,944	7,112
CPU cores	96	32
Runtime	**2:08 hours**	**26 hours**

Evaluation. We used the F_1-score to assess localization performance. The F_1-score is the harmonic mean of *Precision* and *Recall* which depend on the number of true positives (TP). To estimate the number of TP, for each test tomogram, bounding boxes were placed at each true location. A detected particle was considered to be a TP if the estimated centroid was located within the bounding box.

Code availability and implementation details

The code can be downloaded for free from our GitLab website (`https://gitlab.inria.fr/serpico/deep-finder`) along with accompanying documentation (`https://deepfinder.readthedocs.io/en/latest/`). DeepFinder is embedded into the new release of Scipion[58] (`https://github.com/scipion-em/scipion-em-deepfinder`), an open-source image processing framework for cryo-electron microscopy (`http://scipion.i2pc.es/`).

Each step of DeepFinder shown in Fig. 1a can be executed with scripts using the API (examples are provided) or with a graphical user interface. These steps may also be embedded in other workflows, for example, if the user needs only the segmentation step. To implement DeepFinder, we used Keras (`http://keras.io`), an open-source toolbox written in Python and using the TensorFlow framework.

All training procedures were achieved using a Nvidia Tesla K80 GPU, running Cuda 8 and cuDNN 6. Below, we display the memory consumption of DeepFinder for different training parameters.

	Batch size: 15	**Batch size:** 25
Patch size: $40 \times 40 \times 40$ voxels	2.56 GB	3.76 GB
Patch size: $56 \times 56 \times 56$ voxels	6.16 GB	9.86 GB

We used Chimera[59] and ChimeraX[60] software for 3D visualization purposes.

Data availability

The synthetic dataset (Dataset #1) is available on the website of the SHREC'19 challenge (`http://www2.projects.science.uu.nl/shrec/cryo-et/2019/`). A tomogram from the experimental dataset of *C. reinhardtii* cells (Dataset #2) (ref.[34;61]) can be found in the Electron Microscopy Data Bank (EMDB) under accession number EMD-3967 (ref.[62]). The test tomogram of the Chlamydomonas pyrenoid used for subtomogram averaging (Dataset #3) (ref.[40]) can be downloaded from the EMDB under accession number EMD-12749, and the raw tilt-series data for this tomogram is available at the Electron Microscopy Public Image Archive (EMPIAR) under accession number EMPIAR-10694. All four tomograms used to train and test the detection of PSII in Chlamydomonas thylakoids (Dataset #4) (ref.[38]) can be downloaded from the EMDB under accession numbers EMD-10780, EMD-10781, EMD-10782, and EMD-10783.

References

[46] Kingma, D. P. & Ba, J. L. ADAM: a method for stochastic optimization. *arXiv Prepr.* (2014). arXiv:1412.6980v9.

[47] Salehi, S. S. M., Erdogmus, D. & Gholipour, A. Tversky loss function for image segmentation using 3D fully convolutional deep networks. In *Proc. MICCAI workshop on Machine Learning in Medical Imaging (MLMI)*, 379-387 (2017).

[48] Milletari, F., Navab, N. & Ahmadi, S.-a. V-Net : fully convolutional neural networks for volumetric medical image segmentation. In *Proc. IEEE Int. Conf. 3D Vision (3DV)*, 565-571 (2016).

[49] Simonyan, K. & Zisserman, A. Very deep convolutional networks for large-scale image recognition. In *Proc. Int. Conf. Learn. Representation*, 1–14 (2015).

[50] Comaniciu, D., Meer, P. & Member, S. Mean Shift : a robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**, 603–619 (2002).

[51] Martinez-Sanchez, A., Garcia, I., Asano, S., Lucic, V. & Fernandez, J.-j. Robust membrane detection based on tensor voting for electron tomography. *J. Struct. Biol.* **186**, 49–61 (2014).

[52] Kremer, J. R., Mastronarde, D. N. & McIntosh, J. R. Computer visualization of three-dimensional image data using IMOD. *J. Struct. Biol.* **116**, 71–76 (1996).

[53] Zivanov, J. *et al.* New tools for automated high-resolution cryo-EM structure determination in RELION-3. *Elife* **7**, e42166 (2018).

[54] Bharat, T. B. & Scheres, S. Resolving macromolecular structures from electron cryo-tomography data using subtomogram averaging in RELION. *Nat. Protoc* **11**, 2054–2065 (2016).

[55] Harauz, G. & van Heel, M. Exact filters for general geometry three dimensional reconstruction. *Optik (Stuttg)* **78**, 146–156 (1996).

[56] Rosenthal, P. B. & Henderson, R. Optimal determination of particle orientation, absolute hand, and contrast loss in single-particle electron cryomicroscopy. *J. Mol. Biol.* **333**, 721–745 (2003).

[57] Chen, S. *et al.* High-resolution noise substitution to measure overfitting and validate resolution in 3D structure determination by single particle electron cryomicroscopy. *Ultramicroscopy* **135**, 24–35 (2013).

[58] de la Rosa-Trevín, J. *et al.* Scipion: a software framework toward integration, reproducibility and validation in 3D electron microscopy. *Journal of Structural Biology* **195**, 93–99 (2016).

[59] Pettersen, E. F. *et al.* UCSF Chimera - a visualization system for exploratory research and analysis. *J. Comput. Chem.* **25**, 1605–1612 (2004).

[60] Goddard, T. *et al.* UCSF ChimeraX: Meeting modern challenges in visualization and analysis. *Protein Sci.* **27**, 14–25 (2018).

[61] Albert, S. *et al.* Proteasomes tether to two distinct sites at the nuclear pore complex. *Proc. Natl. Acad. Sci.* **114**, 201716305 (2017).

[62] Henderson, R. Avoiding the pitfalls of single particle cryo-electron microscopy : Einstein from noise. *Proc. Natl. Acad. Sci.* **110**, 18037–18041 (2013).

Acknowledgements

This work was jointly supported by the Fourmentin-Guilbert Foundation and Région Bretagne (Brittany Council). Calculations were performed on the Inria Rennes computing grid facilities partly funded by France-BioImaging infrastructure (French National Research Agency - ANR-10-INBS-04-07, "Investments for the future") and at the Max Planck Institute for Biochemistry computing cluster, Martinsried, Germany. L. Lamm, R.D. Righetto, W. Wietrzynski, T. Peng, and B.D. Engel were supported by DFG grant EN 1194/1-1 as part of FOR 2092, The Munich School for Data Science (MUDS), and Helmholtz Association. A. Martinez-Sanchez was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2067/1- 390729940.

We thank F. Förster and M. Killinger for fruitful discussions about cryo-ET data analysis and deep learning applied to large 3D volumes analysis, respectively.

We thank the organizers of the SHREC'19 and SHREC'20 challenges for helpful assistance and for providing the template matching results: I. Gubins and R.C. Veltkamp (Utrecht University, Department of Information and Computing Sciences), G. van der Schot and F. Förster (Utrecht University, Department of Chemistry).

Finally, we thank S. Prima for careful reading of the paper and valuable suggestions and comments.

Author contributions

E.M. designed and implemented the presented DeepFinder method and carried out the biocomputing experiments. C.K. supervised the project and was in charge of overall direction and planning. E.F., D.L., and C.K. devised the project and the main conceptual ideas, with assistance from A.M.-S. B.D.E. and W.B. facilitated access to datasets. B.D.E., S.A., W.W., and S.P. provided the *C. reinhardtii* datasets and annotations (Datasets #2, #3 and #4). A.M.-S., J. Ortiz and B.D. Engel conceived experiments on real datasets. L.L., R.D.R., W.W., and T.P. performed experiments on datasets depicting thylakoid membranes and pyrenoid matrices within vitreously-frozen *C. reinhardtii* cells. E.M., B.D.E. and C.K. co-wrote the paper. All authors provided critical feedback and helped shape the research, analysis and paper.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to C. Kervrann and B.D. Engel.

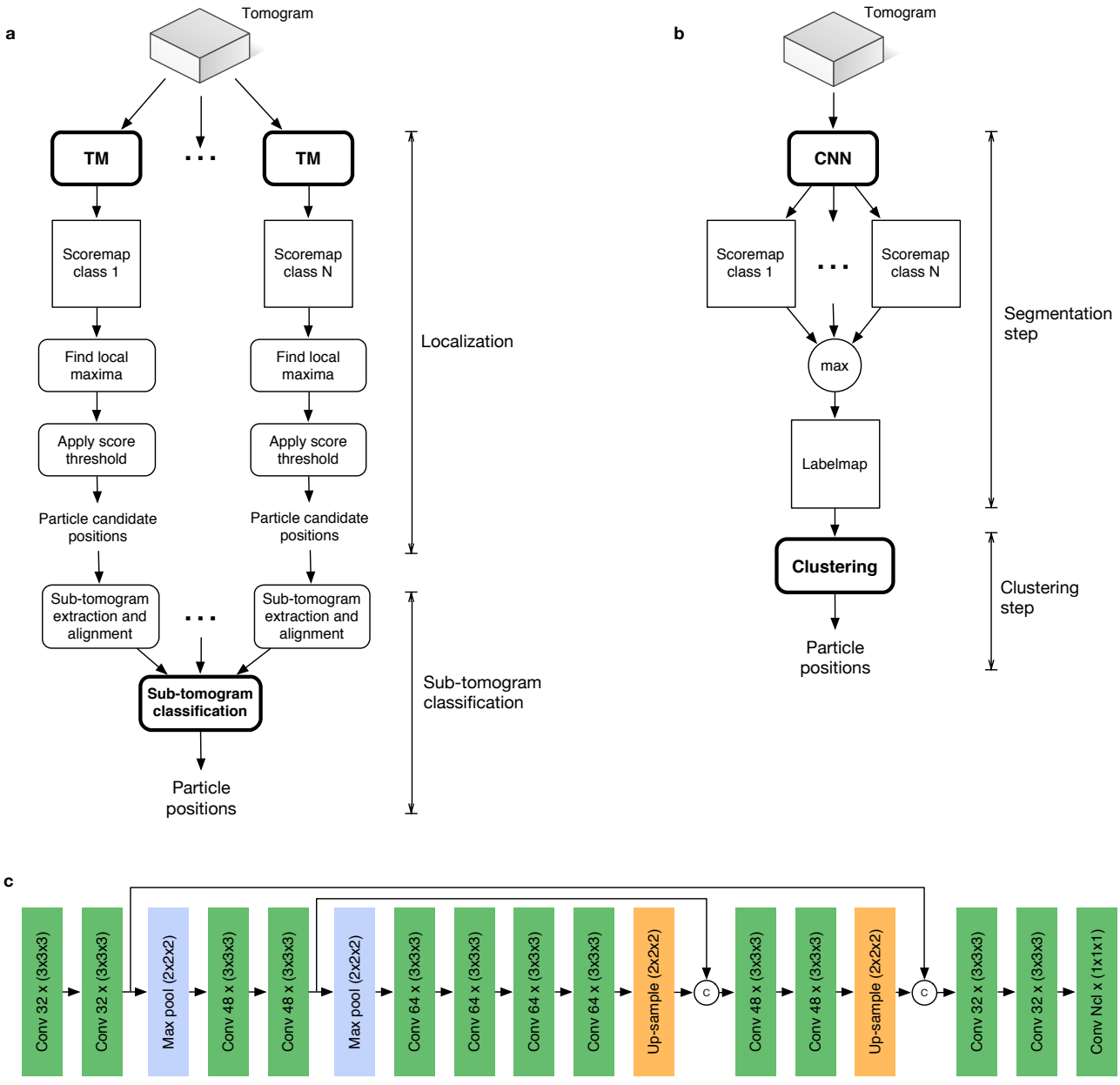
Supplementary information

Deep learning improves macromolecule identification in 3D cellular cryo-electron tomograms

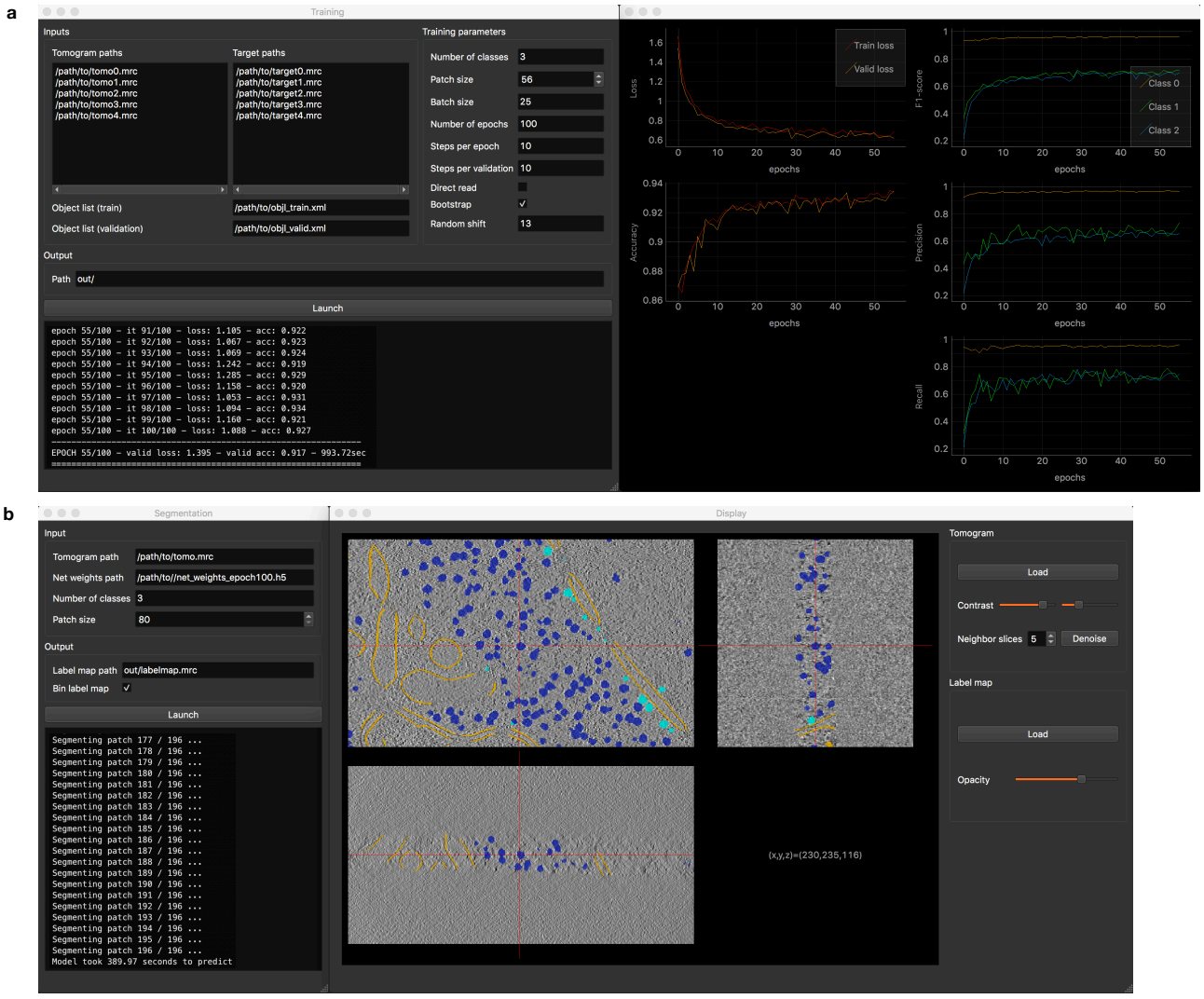
E. Moebel, A. Martinez-Sanchez, L. Lamm, R.D. Righetto, W. Wietrzynski, S. Albert, D. Larivière, E. Fourmentin, S. Pfeffer, J. Ortiz, W. Baumeister, T. Peng, B.D. Engel, C. Kervrann

Membranes		M1	M2	M3	M4	M5	M6	M7	M8	M9	**Global**
Mono-class PSII	F_1-score	0.557	0.465	0.533	0.625	0.789	0.516	0.571	0.588	0.286	**0.566**
	Precision	0.710	0.476	0.800	0.625	0.789	0.727	0.556	0.500	0.400	**0.643**
	Recall	0.458	0.455	0.400	0.625	0.789	0.400	0.588	0.714	0.222	**0.505**
Multi-class PSII	F_1-score	0.632	0.582	0.714	0.737	0.696	0.250	0.562	0.182	0.571	**0.619**
	Precision	0.638	0.485	0.625	0.636	0.593	0.750	0.600	0.250	0.800	**0.601**
	Recall	0.625	0.727	0.833	0.875	0.842	0.150	0.529	0.143	0.444	**0.638**
Template Matching PSII	F_1-score	0.400	0.200	0.355	0.258	0.350	0.279	0.318	0.000	0.286	**0.313**
	Precision	0.556	0.375	0.344	0.571	0.333	0.261	0.259	0.000	0.400	**0.353**
	Recall	0.312	0.136	0.367	0.167	0.368	0.300	0.412	0.000	0.471	**0.281**

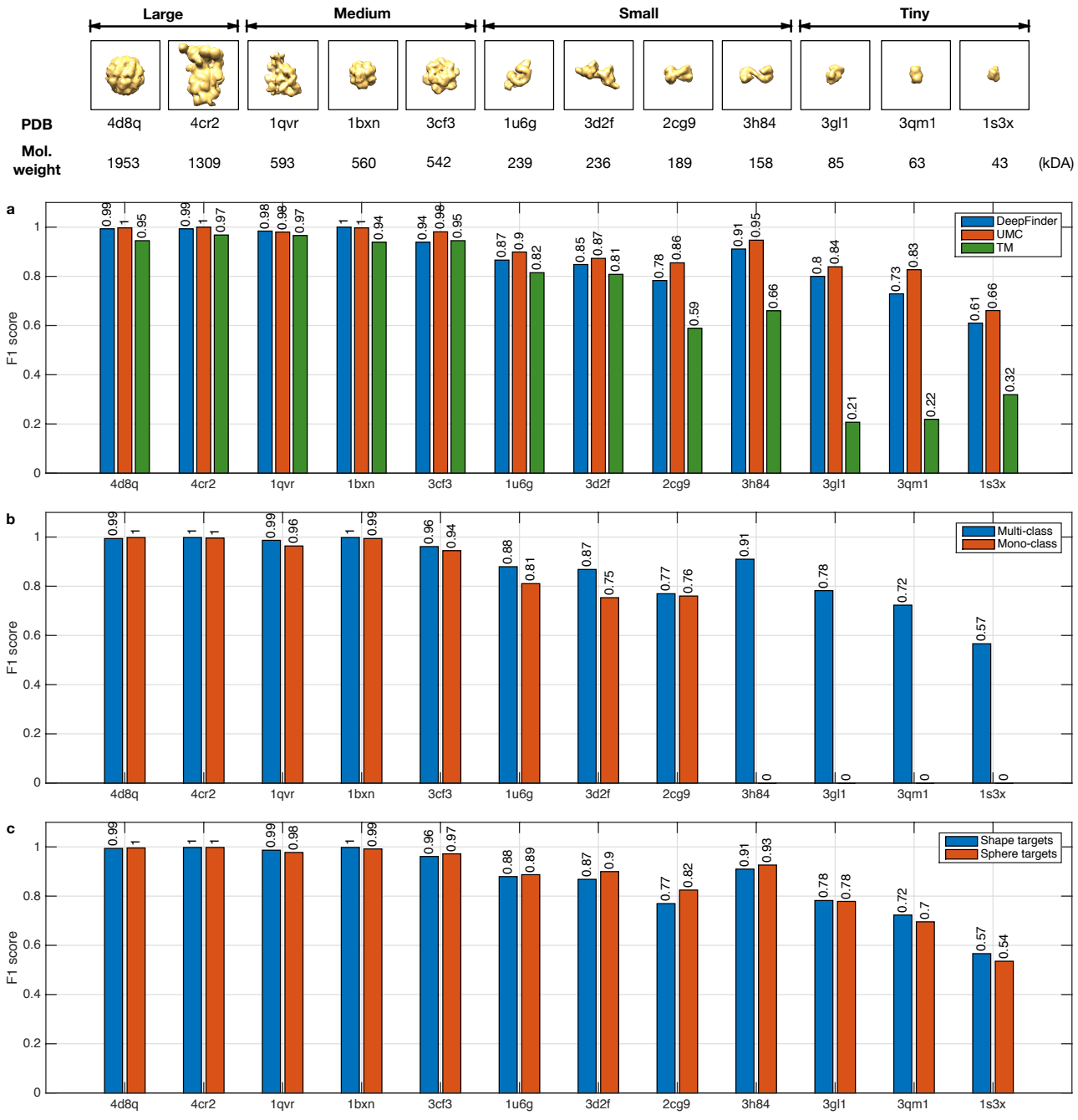
Supplementary Table 2: **Comparison of F_1-scores for the detection of PSII complexes embedded within native thylakoid membranes (Dataset #4)**. For the test tomogram, we ran the DeepFinder mono-class (470 particles) and multi-class (508 particles) strategies, as well as PyTOM template matching (508 particles). For an even comparison, we thresholded the template matching hits to match the number of picks from the DeepFinder multi-class approach.The scores were measured after masking the picks to different membranes (M1, M2...) of the test tomogram. These membranes vary in resolution and in the number of PSII complexes they host.



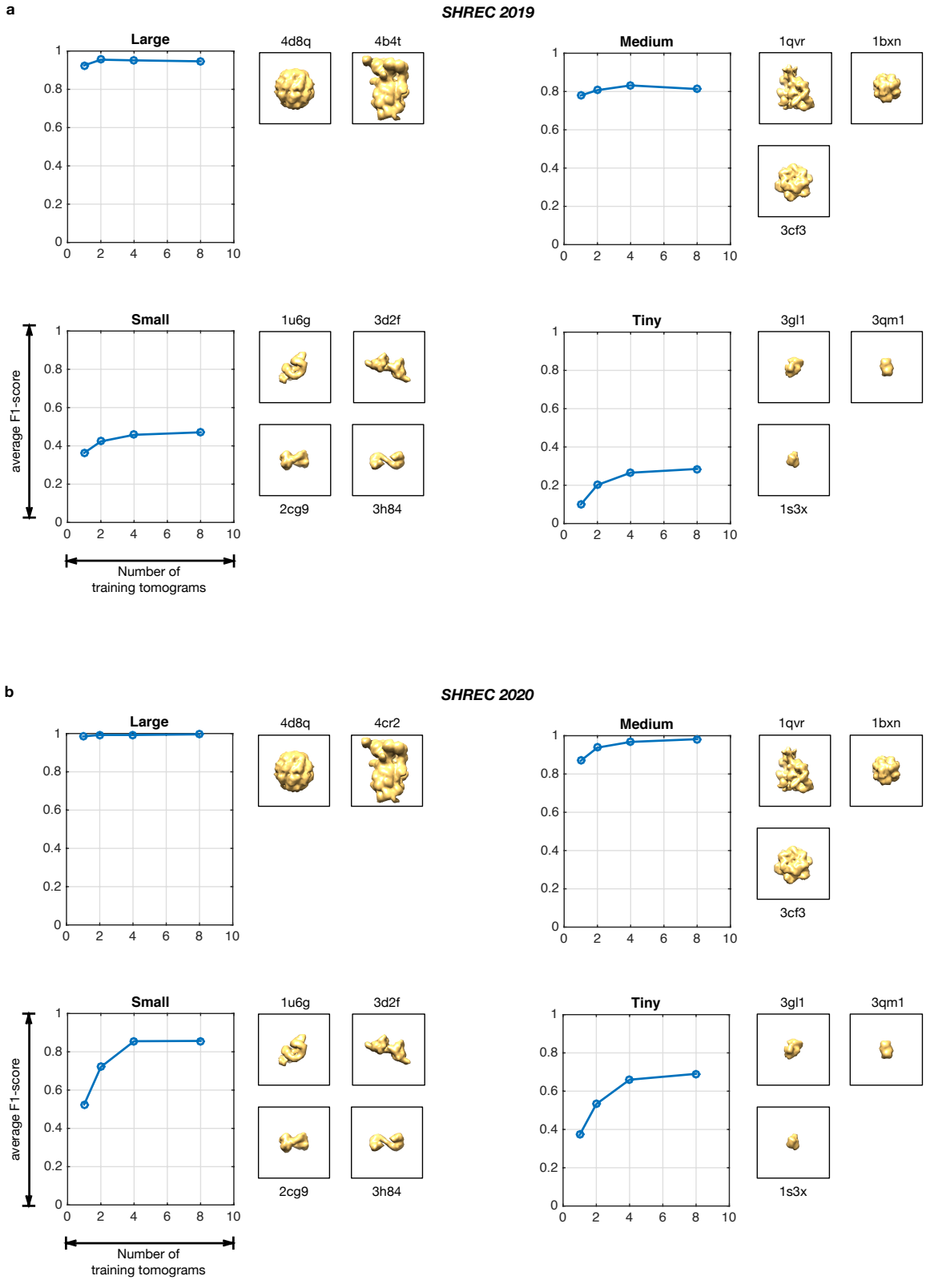
Extended Data Fig. 1: Two workflows for macromolecule localization in cryo-ET. a, Conventional processing pipeline based on template matching. **b,** DeepFinder (analysis stage): a multi-class approach able to localize particles of several different macromolecular species in one pass. **a,** and **b,** highlight why DeepFinder is more agile than template matching when several macromolecule classes need to be localized. **c,** CNN architecture used in DeepFinder and based on U-Net[27]. The architecture adopts the encoder-decoder paradigm, which produces an output volume with the same size as the input volume. Each green box represents a convolutional layer. The number of filters n and the filter size s is labeled as $n \times (s \times s \times s)$. All convolutional layers are followed by a ReLU activation function, except the last layer, which uses a soft-max function. The up-sampling is achieved with up-convolutions (also called "backward-convolution"). Combining feature maps from different scales is performed by concatenation along the channel dimension. In the end, the total number of architecture parameters is approximately 903k. More precisely, this number depends slightly on N_{cl}, the number of classes: $902,928 + N_{cl} \times 33$.



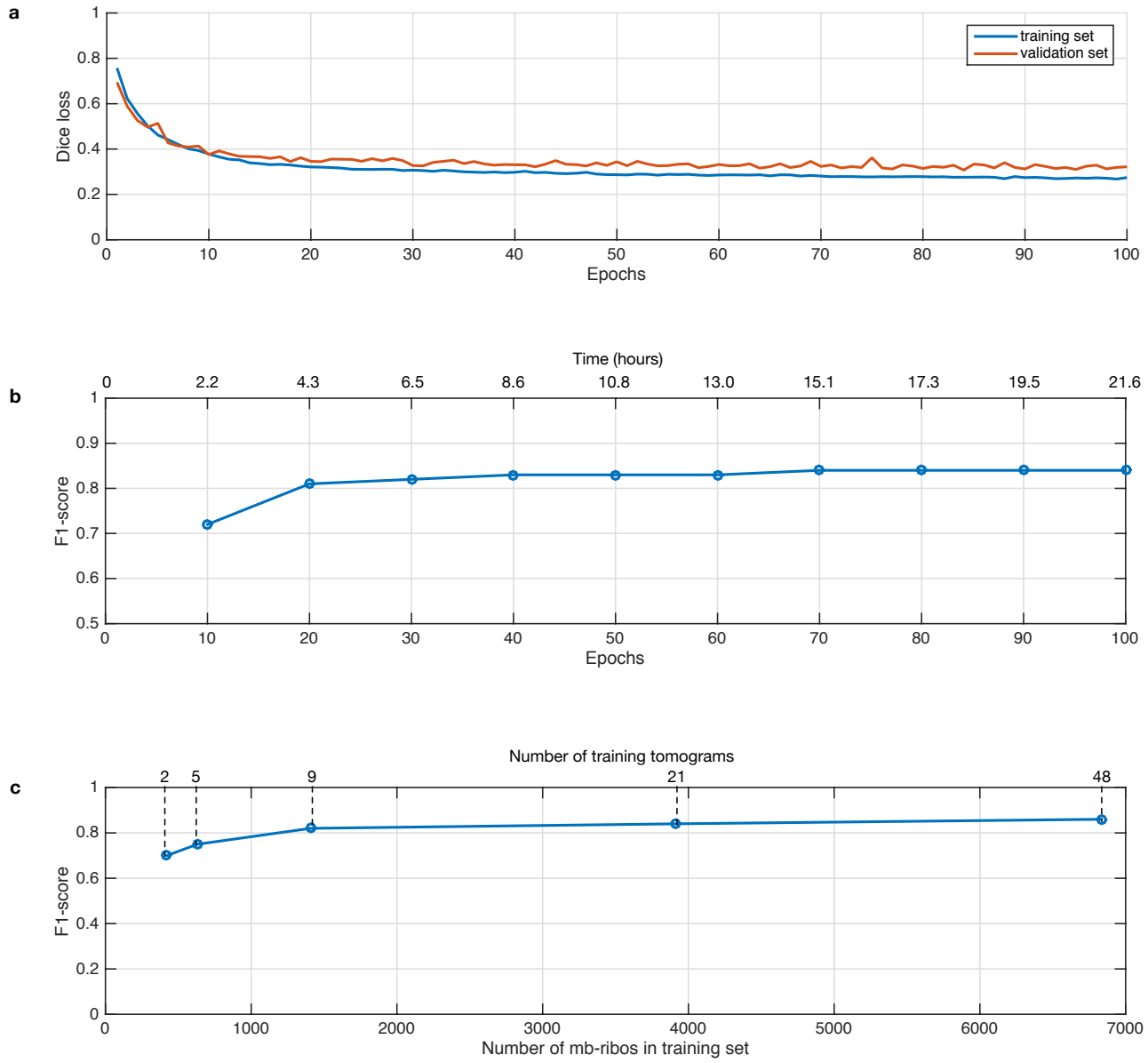
Extended Data Fig. 2: DeepFinder graphical user interface. a, Training interface composed of a first window for parametrizing the procedure and a second window for displaying the training metrics in real-time. **b,** Segmentation interface which also opens a data visualization tool. This tool allows the user to explore the tomogram with superimposed segmentations. In addition, DeepFinder also incorporates interfaces for tomogram annotation, target generation and clustering (see the documentation at `https://gitlab.inria.fr/serpico/deep-finder` for more information).



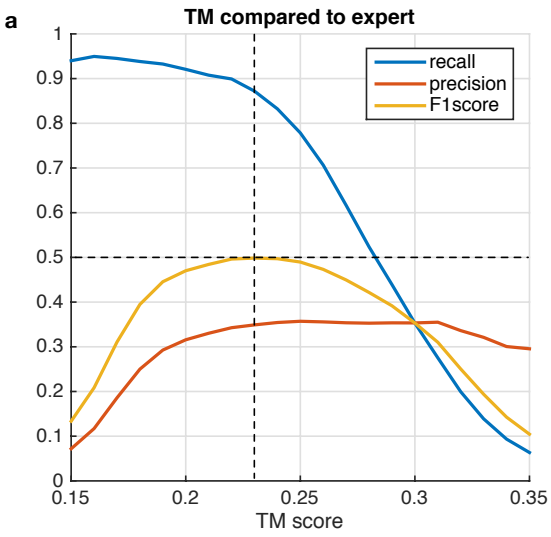
Extended Data Fig. 3: Analysis of algorithm performance on the synthetic dataset (SHREC'20 challenge). a, Performance (F_1-score) of DeepFinder, UMC and template matching algorithms and ability of algorithms to discriminate between 12 classes/subclasses of macromolecules. The highest (best) possible value of an F_1-score is 1.0 and the lowest (worst) possible value is 0. The scores of template matching were provided by the SHREC'20 challenge organizers (Utrecht University, Department of Information and Computing Sciences and Department of Chemistry). **b,** Performance of DeepFinder implemented as a multi-class network architecture and as an architecture made of 12 binary networks. These two architectures differ only by the number of output neurons. **c,** Influence of the training target generation method ("shapes" versus "spheres"). In the case of "shapes", the exact shapes of the macromolecules have been used to annotate the tomograms. In the case of "spheres", the shape and the orientation of macromolecules are not needed to generate the training targets. This analysis used 8 tomograms for training, 1 tomogram for validation, and 1 tomogram for testing.

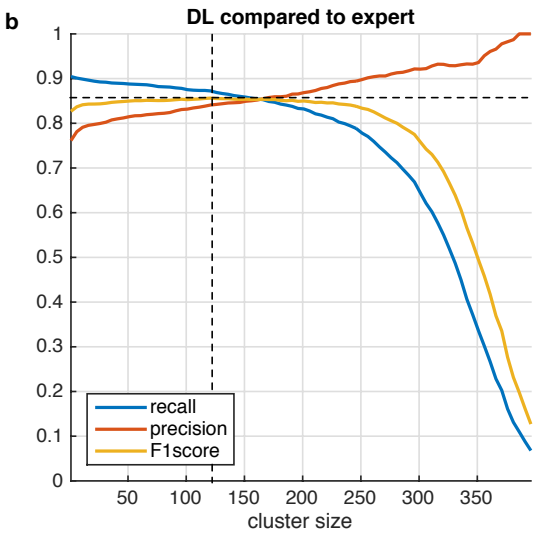


Extended Data Fig. 4: Evolution of F_1-scores with respect to sizes of the training sets (number of tomograms) on the synthetic SHREC dataset (12 classes). Scores are displayed for both the SHREC 2019 **a,** and 2020 **b,** editions. This figure gives an estimation of the amount of annotated data needed to identify macromolecules. This amount depends on the size of the target macromolecule: smaller targets require more annotations. Each tomogram contains in average 208 macromolecules per class. The macromolecules have been categorized into 4 groups (large, medium, small and tiny). This analysis used 8 tomograms for training, 1 tomogram for validation, and 1 tomogram for testing.

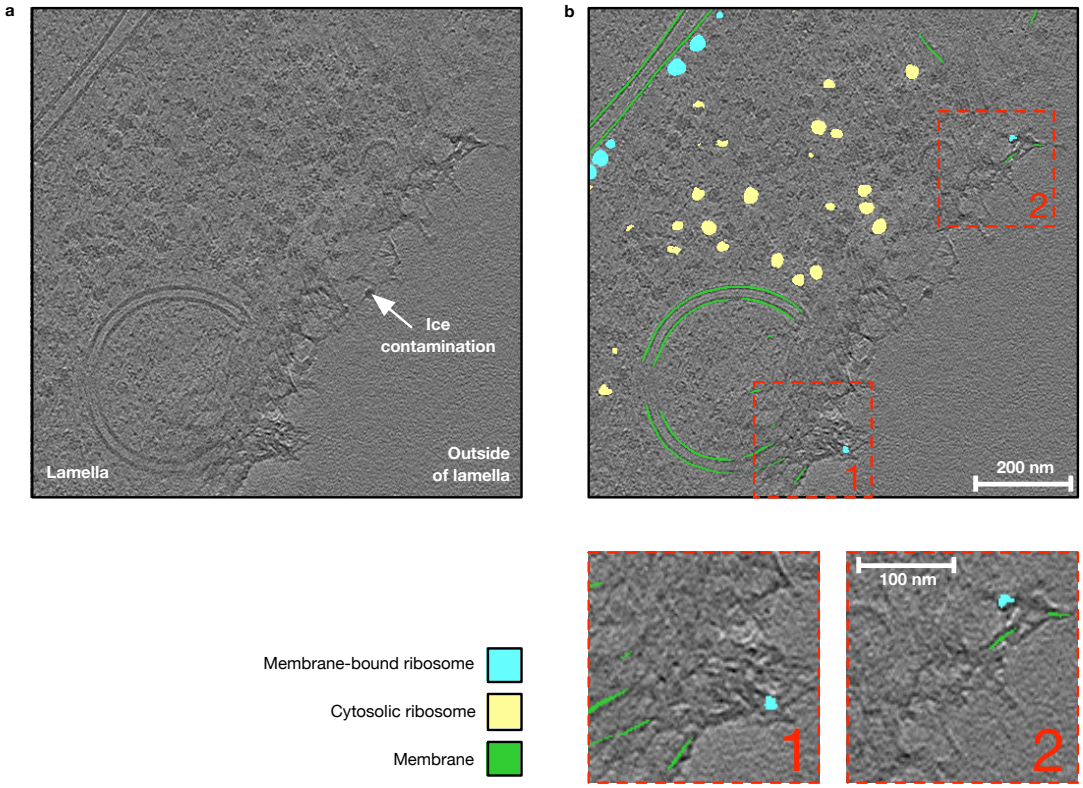


Extended Data Fig. 5: Evolution of F_1-score with respect to training iterations and training set size on real cryo-ET Dataset #2, *Chlamydomonas reinhardtii* (3 classes). a, The loss, which quantifies the segmentation quality, is computed for the training set, as well as for the validation set. Comparing both curves allows assessment of the generalization capabilities of DeepFinder. The curves for both sets should ideally overlap, otherwise it indicates overfitting (the network memorizes trained samples instead of learning discriminating features). One epoch equals 100 training iterations. **b,** The F_1-score, which quantifies the localization performance, computed on the test set. The F_1-score is obtained by comparing the membrane-bound ribosomes found by DeepFinder to expert annotations. The time axis has been obtained using a Tesla K80 GPU. The curve indicates that competitive particle picking results are obtained after 20 epochs, or 4.3 hours with the required GPU. This analysis used 21 tomograms for training, 1 tomogram for validation, and 8 tomograms for testing.**c,** In a similar fashion to Extended Data Fig. 4, this curve provides an estimate of the quantity of training data required to achieve a competitive result. It appears that this quantity is 1,400 ribosomes (9 tomograms), which is a typical size for a cryo-ET dataset. On first glance, this estimate seems to contradict the estimates in Extended Data Fig. 4: the numbers do not coincide (the curve labeled "Large" estimates that quantity at 208 particles). Note that SHREC'19 is a synthetic dataset, composed of 12 classes. Here, we are dealing with a real cellular dataset consisting of 3 classes (membrane, *mb-ribo*, and *ct-ribo*). It appears that having a larger number of classes enables the use of smaller training sets. On the other hand, the case of real data is more difficult, notably because of the presence of "label noise" (errors due to the annotation pipeline) and other sources of signal corruption such as the missing wedge, the contrast transfer function and the low signal-to-noise ratio (in part caused by increased molecular crowding inside cells). This analysis 1 tomogram for validation, and 8 tomograms for testing.

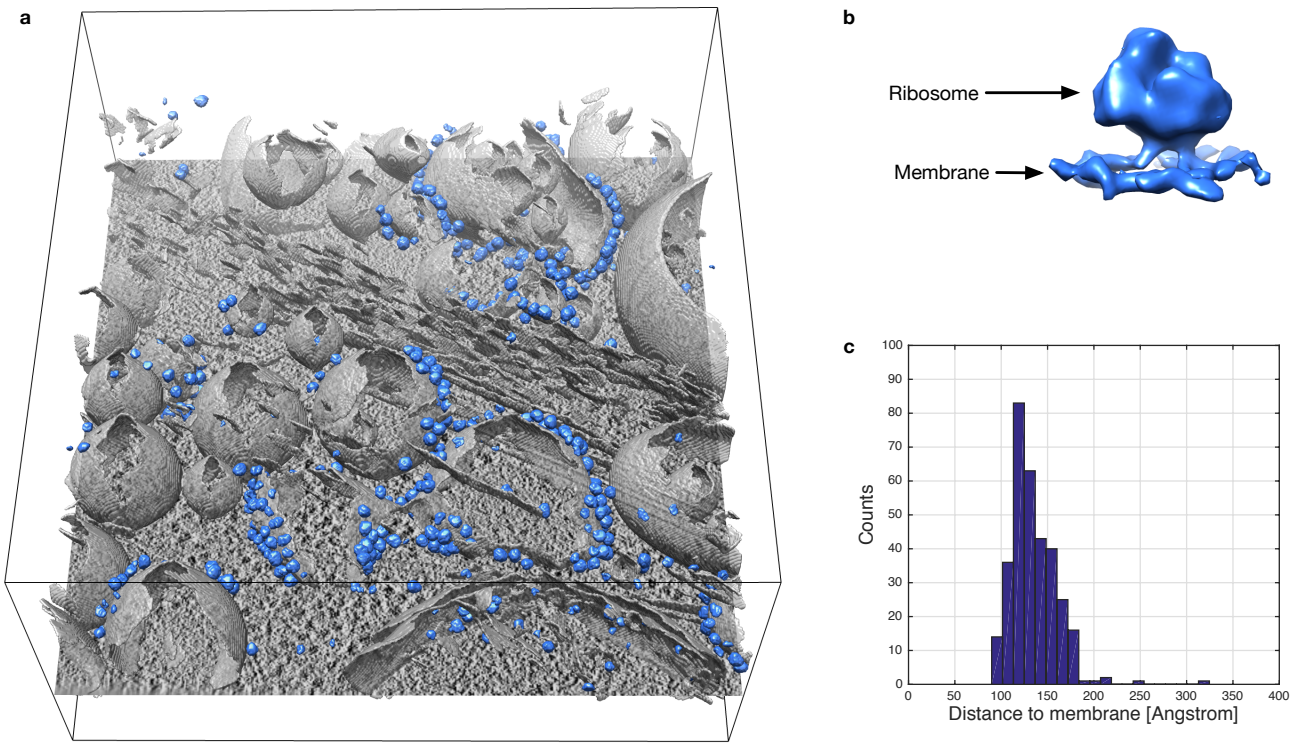




Extended Data Fig. 6: Quantitative analysis of overlap with expert annotations on cellular cryo-ET data (Dataset #2, *mb-ribos*). We varied the thresholds of template matching (**a**) and DeepFinder (**b**) to compute the *Recall* (ratio between the number of true positives and the number of particles in the ground truth), *Precision* (ratio between the number of true positives and the number of detected particles) and F_1-score (2 $\times$ (Recall $\times$ Precision) / (Recall + Precision)) curves. The threshold parameter for template matching is the constrained correlation coefficient, and for DeepFinder it is the cluster size, which corresponds to the macromolecule volume (in voxels). We obtained a maximum F_1-score of 0.86 for DeepFinder and a maximum F_1-score of 0.50 for template matching (with no post-classification step, see Extended Data Fig. 1a). Template matching and DeepFinder both have good *Recall* values, but template matching has a lower *Precision* than DeepFinder. This suggests that template matching can be recommended to select many candidates, but a time-consuming post-classification is required to improve *Precision*. DeepFinder has much higher *Precision* values, which confirms the results from the synthetic dataset (SHREC'19 challenge). This analysis used 48 tomograms for training, 1 tomogram for validation, and 8 tomograms for testing.



Extended Data Fig. 7: **DeepFinder handles ice contamination on the lamella surface. a,** Tomogram slice depicting the border of a FIB-milled lamella. The lamella contains a *Chlamydomonas reinhardtii* cell, with a lamella surface suffering from ice contamination. **b,** Tomogram slice with superimposed DeepFinder segmentation. Most of the ice contamination artifacts have been correctly classified as "background". Nonetheless, some missclassifications exist, as can be observed in the zoomed-in boxes (in dashed red). In boxes 1 and 2, DeepFinder confuses some artifacts with membranes, and some features are wrongly classified as membrane-bound ribosomes. Such missclassifications can be filtered out, either by masking the boundaries of the lamella, or by rejecting segmented objects that are too small (using the "cluster size" attribute given by the clustering step of the DeepFinder analysis stage). This analysis used 48 tomograms for training, 1 tomogram for validation, and 8 tomograms for testing.

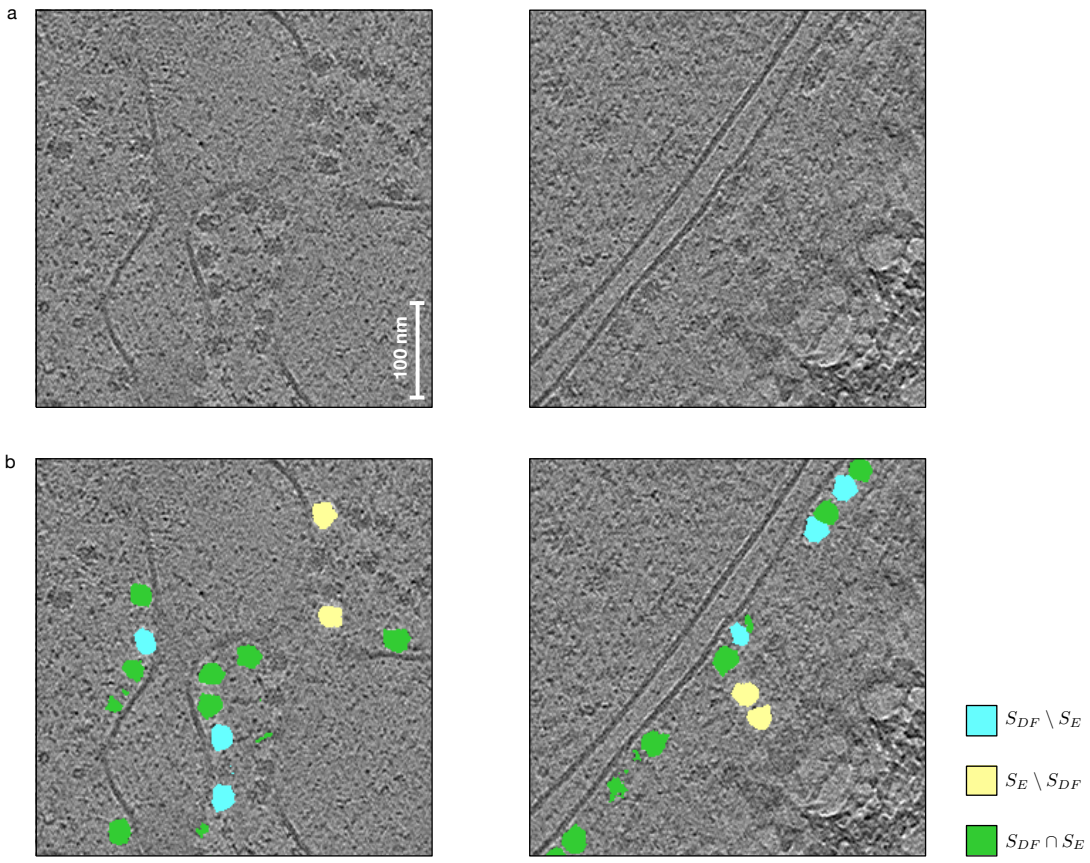


Extended Data Fig. 8: **The generalization potential of DeepFinder on P19 cells**. DeepFinder was trained on the Chlamydomonas (algae) dataset and then applied on a tomogram of mouse P19 cells (EMD-10439). Although the ribosome has a different structure for the two species, for a given voxel size (13.68 Å) the structures are similar enough for DeepFinder to identify and localize *mb-ribo* particles in a P19 cell. **a,** Tomographic slice with both the superimposed segmented cell membrane (gray) and *mb-ribo* particles (blue). **b,** Average density from 300 *mb-ribo* particles. **c,** Histogram of *mb-ribo* particle distance from the nearest cell membrane. In this histogram, the maximum mode is located at 136.8 Å, which corresponds to the ribosome radius. This analysis used 48 tomograms for training, 1 tomogram for validation, and 1 tomogram for testing.

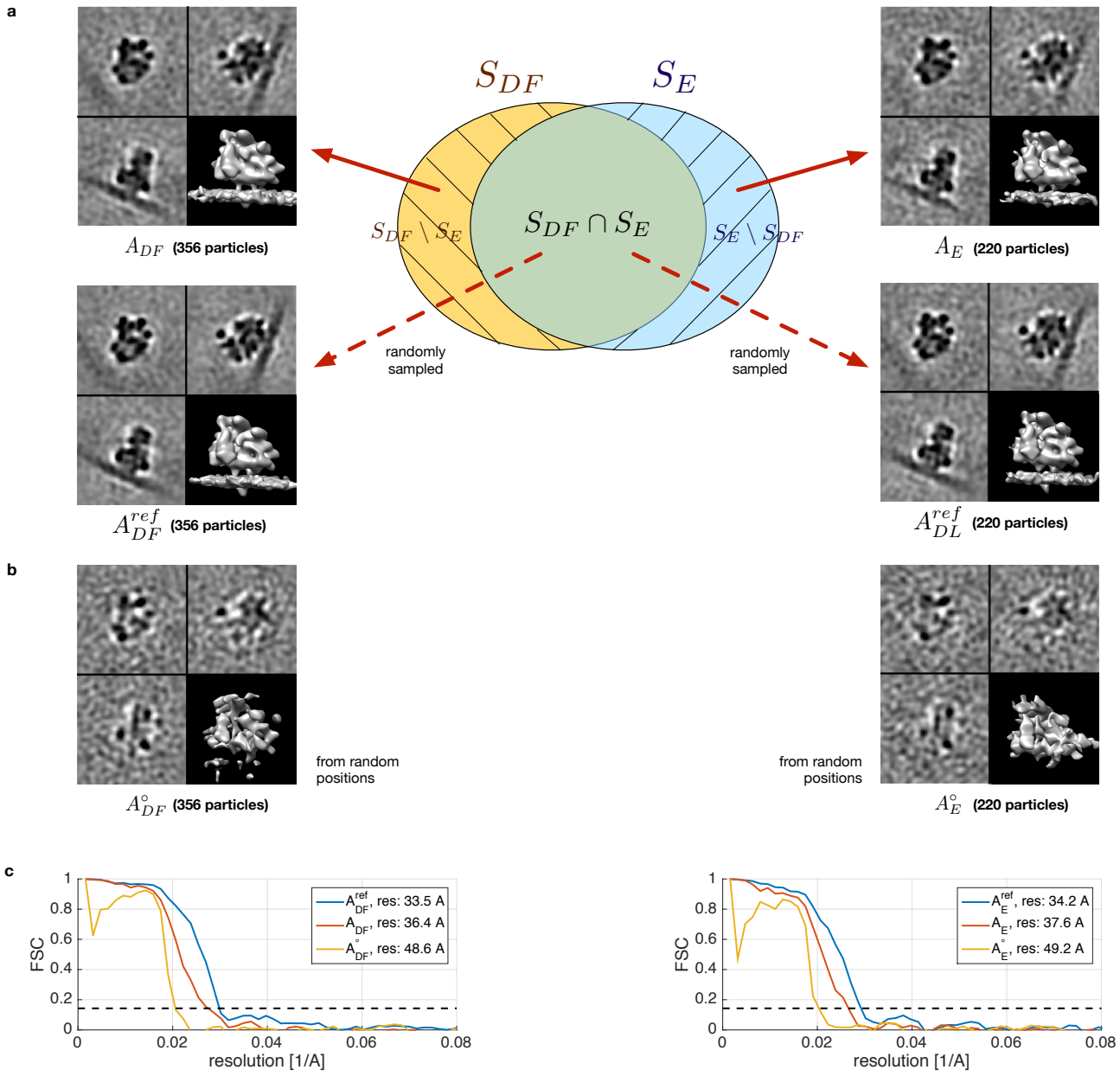
Supplementary Note 1

Analysis of consensus response

In this note, we examine the complementarity between the two sets of *mb-ribo* macromolecules found by the experts and DeepFinder. In the following analysis, we denote the sets obtained by experts and DeepFinder as S_E and S_{DF}, respectively. While the overlap $S_E \cap S_{DF}$ between both sets was substantial (1,516 particles), there was also a significant number of particles belonging to $S_E \setminus S_{DF}$ (220 particles), that is, the particles annotated by the expert but not found by DeepFinder, and to $S_{DF} \setminus S_E$ (356 particles), that is, particles found by DeepFinder but missed by the expert. We can benefit from the two complementary sets of particle positions to improve the overall validation rates. The union $S_E \cup S_{DF}$ of the two sets increases the list of potential *mb-ribo* macromolecules, for which a confidence level can be assigned to each set member depending on whether it belongs to $S_E \cap S_{DF}$, $S_{DF} \setminus S_E$ or $S_{DF} \setminus S_E$. The particles belonging to $S_E \cap S_{DF}$, that is, found by both methods, are very likely to be true positives. Meanwhile the particles belonging to $S_E \setminus S_{DF}$ and $S_{DF} \setminus S_E$ can be labeled as "suspicious" and require more investigation. These two non-union sets are relatively small, enabling assignment of the bulk of the high-confidence particles so the expert can focus on validating the remaining low-confidence particles. In this manner, it is possible to uncover inaccuracies in the expert annotations and refine the true-positive particle class, which can further improve the training performance of DeepFinder.



Analysis of localization consensus between DeepFinder and experts. **a,** Two tomogram slice ROIs depicting *Chlamydomonas reinhardtii* cells. **b,** Membrane-bound ribosomes mapped into the ROIs. The ribosomes found by DeepFinder but missed by the experts ($S_{DF} \setminus S_E$) are blue. The ribosomes found by the experts but missed by DeepFinder ($S_E \setminus S_{DF}$) are yellow. The ribosomes found by both DeepFinder and the experts ($S_{DF} \cap S_E$) are green. As expected, members of $S_{DF} \cap S_E$ constitute the majority of identified ribosomes. Members of $S_{DF} \setminus S_E$ tend to be found at locations where the membrane has less contrast (**b**, left) or where neighboring ribosomes are close (**b**, right). Members of $S_E \setminus S_{DF}$, which were obtained with the expert pipeline (template matching and CPCA clustering), may also be located at positions where membrane contrast is low (**b**, left). Nevertheless, it appears that this pipeline has a tendency of confusing membrane-bound and cytosolic ribosomes. The proximity of ice-contamination (**b**, right) also seems to be a factor responsible for missclassifications. This analysis used 48 tomograms for training, 1 tomogram for validation, and 8 tomograms for testing.



Analysis of consensus decisions and overlap sets (Dataset #2). a, The central Venn diagram represents the overlap between the *mb-ribo* sets S_{DF} (found by DeepFinder) and S_E (annotated by expert). Thus, $S_E \cap S_{DF}$ is the subset of *mb-ribo* particles found by both DeepFinder and the experts, $S_{DF} \setminus S_E$ is the subset of *mb-ribos* found by DeepFinder only (and missed by the experts), and $S_E \setminus S_{DF}$ is the subset of *mb-ribo* particles found by the experts only (and missed by DeepFinder). The origin of red arrows pointing to the subtomogram averages $\mathbf{A}_{DF}$, $\mathbf{A}^{ref}_{DF}$, $\mathbf{A}_E$, $\mathbf{A}^{ref}_E$ indicate the particle subsets used to compute the averages. A ribosome density is clearly visible in $\mathbf{A}_{DF}$, therefore one can safely assume that the FP rate in $S_{DF} \setminus S_E$ is low. **b,** The subtomogram averages $\mathbf{A}^{\circ}_{DF}$ and $\mathbf{A}^{\circ}_E$ have been computed using subtomograms sampled from random positions. These averages serve to estimate a lower bound for the FSC curve. The correlation values equal or below this bound are considered "noise" values, and are caused by alignment bias[62]. **c,** FSC curves for the above subtomogram averages. The averages $\mathbf{A}^{ref}_{DF}$ and $\mathbf{A}^{ref}_E$ have both led to a higher resolution than $\mathbf{A}_{DF}$ and $\mathbf{A}_E$, implying that the *mb-ribo* particles in the set $S_{DF} \setminus S_E$ and in the set $S_E \setminus S_{DF}$ are more heterogenous than the *mb-ribo* particles in the set $S_{DF} \cap S_E$. Also, $\mathbf{A}_{DF}$ and $\mathbf{A}_E$ have led to a higher resolution than $\mathbf{A}^{\circ}_{DF}$ and $\mathbf{A}^{\circ}_E$, meaning that the impact of alignment bias is not significant. This analysis used 48 tomograms for training, 1 tomogram for validation, and 8 tomograms for testing.