Automatic *de novo* atomic-accuracy structure determination for cryo-EM maps using deep learning

Jiahua He & Sheng-You Huang*

School of Physics, Huazhong University of Science and Technology, Wuhan, Hubei 430074, P. R. China

Submitted to *Nature Communications* on June 17, 2020

Key Words: Cryo-electron microscopy (cryo-EM), Structure determination, Deep learning, DenseNet

*Email: huangsy@hust.edu.cn; Phone: +86-27-87543881; Fax: +86-027-87556576

Abstract

Motivation and Results: Advances in microscopy instruments and image processing algorithms have led to an increasing number of cryo-EM maps. However, building accurate models for the EM maps at 3-5 Å resolution remains challenging and time-consuming. Here, we present a fully automatic *de novo* structure determination method using a deep learning-based framework, named as DeepMM, which automatically builds atomic-accuracy all-atom models from cryo-EM maps at near-atomic resolution. In our method, the main-chain and Cα positions as well as their amino acid and secondary structure types are predicted in the EM map using Densely Connected Convolutional Networks. DeepMM was extensively validated on 40 simulated maps at 5 Å resolution and 30 experimental maps at 2.6-4.8 Å resolution as well as an EMDB-wide data set of 2931 experimental maps at 2.6-4.9 Å resolution. DeepMM built correct models for $>$60% of the cases, and it outperformed existing state-of-the-art algorithms including RosettaES, MAINMAST, and Phenix.

Availability: http://huanglab.phys.hust.edu.cn/DeepMM/

Note: This manuscript has been submitted to *Nature Communications* on June 17, 2020.

1 Introduction

Cryo-electron microscopy (cryo-EM) has now become a widely used technique for structure determination of macromolecular structures in the recent decade[1–3]. Advances in microscopy instruments and image processing algorithms have led to the rapid increase in the number of solved EM maps[1–3]. The 'resolution revolution' in cryo-EM has paved a way for the determination of high-resolution structures of previously intractable biological systems[4–10]. According to the statistics of the Electron Microscopy Data Bank (EMDB)[11], there were 2435 maps deposited in 2019, which are almost 4 times the 640 maps released in 2015.

With the rapid growth of deposited EM maps, there is an increasing gap between the maps and reconstructed/modeled 3-dimensional (3D) structures. As of April 1, 2019, there were 10560 EMDB maps, but only 4805 associated structures were deposited in the Protein Data Bank (PDB)[12]. For those maps determined at near-atomic resolution (3.0 to 5.0 Å), it is difficult to build high-resolution models with conventional software designed for X-ray crystallography. In view of the fact that near-atomic resolution maps take up the majority of current and henceforth released maps[11], tools, which can reconstruct structures *de novo* from EM maps without using known structures as templates[13], are pressingly needed. As such, some algorithms like EM-fold[14], Gorgon[15], Rosetta[16,17], Pathwalking[18–20], Phenix[21,22], and MAINMAST[23,24], have been recently presented for constructing and/or assembling structure fragments for EM maps. EM-fold tries to identify density rods in the EM map and places α-helices into the density rods using a simulated annealing Monte Carlo search algorithm[14]. Gorgon calculates the density skeleton and matches the predicted secondary structure elements (SSEs) from sequence to those in a density map[15]. RosettaES iteratively identifies structure segments from the density map and combines them using a Monte Carlo assembly method[17]. Pathwalking builds the protein Cα model from an EM map using the travelling salesman problem (TSP) solver[18]. The *phenix.map_to_model* tool in Phenix is an automatic *de novo* modeling tool in the Phenix package for building initial models of structural fragments for EM maps[22,25,26].

MAINMAST is a state-of-the-art *de novo* modeling method for near-atomic resolution cryo-EM maps developed by Terashi *et al.*[23] First, MAINMAST identifies local dense points (LDPs) in an EM map using the mean shift method. Next, the LDPs are connected into a minimum spanning

tree (MST). Then, the MST structure is refined using a Tabu search method. After that, the protein sequence is mapped onto the longest path of each refined tree using the Smith-Waterman dynamic programming algorithm[27], which results in a protein Cα model. Multiple trees are generated through the combinations of different parameters. These trees are then ranked based on a threading score, which evaluates the fitness between the sequence and the main-chain path. Full-atom structures are constructed from the Cα model using PULCHRA[28], and then refined with MDFF[29]. Recently, the algorithm of MAINMAST was also applied to segmenting EM maps with symmetry[30].

Despite the present progress in *de novo* structure building for cryo-EM maps, there are various limitations in current approaches. They can either only build structural fragments[14,15,22] or have low accuracy in terms coverage and/or sequence reproduction [17,18,23]. It remains challenging to automatically build an accurate all-atom structure from the EM maps at near-atomic resolution. Recently, machine learning has been actively applied in structure determination for EM maps, such as single particle picking[31], tomogram annotation[32], and secondary structure prediction[33]. However, applying deep learning in structure reconstruction for near-atomic resolution EM maps remains a challenging work. Si *et al* have developed a deep learning approach, C-CNN, which uses a set of cascaded convolutional neural networks to predict the protein backbone structure from high-resolution cryo-EM maps[34]. Despite its high coverage in recovering Cα atoms, C-CNN cannot build full-length protein structures for experimental maps.

Here, we have developed a fully automatic *de novo* atomic-accuracy structure reconstruction method for EM maps at near-atomic resolution through Densely Connected Convolutional Networks (DenseNets) using a deep learning-based framework, named DeepMM. Instead of tracing the protein main-chain on the raw EM density map, DeepMM first predicted the probability of main-chain atoms (N, C, and Cα) and Cα positions near each grid point using one DenseNet[35]. Then, the method traced the main-chain according to the predicted main-chain probability map. The amino acid and secondary structure types were predicted by a second DenseNet. Finally, the protein sequence was aligned to the main-chain according to the predicted Cα probabilities, amino acid types, and secondary structure types for all-atom structure building.

2 Results

2.1 Workflow of DeepMM

The workflow of DeepMM is illustrated in Figure 1**a**. Specifically, staring from a cryo-EM map and the target protein sequence, DeepMM first preprocesses the cryo-EM map, standardizes the order of axis, and interpolates grid interval to 1.0 Å. Then, DeepMM cuts the entire map into small voxels of size $11\text{Å} \times 11\text{Å} \times 11\text{Å}$. Afterwards, one DenseNet (say DenseNet A) is used to predict the main-chain and Cα probability on each of the voxels. All the predicted probability values form a 3D probability map. Next, possible main-chain paths are generated in the predicted main-chain probability map using a main-chain tracing algorithm[23]. The Cα probability values of main-chain points are interpolated from the predicted 3D Cα probability map. Afterwards, the amino acid and secondary structure types are predicted for each main-chain point through the second DenseNet (say DenseNet B). With the predicted Cα probability, amino acid type, and secondary structure type for each main-chain point, the target protein sequence is then aligned to the main-chain paths based on the Smith-Waterman dynamic programming (DP) algorithm[27]. The resulted multiple Cα models are ranked by their alignment scores. Finally, the all-atom structures are constructed from the top Cα models using the *ctrip* program in the Jackal modeling package[36,37] and refined by an energy minimization using Amber[38].

2.1.1 Network architecture

Two Densely Connected Convolutional Networks (DenseNets) are embedded into our DeepMM algorithm. Figure 1**b** illustrates the architecture of the networks. DenseNet is a feed-forward multi-layer network which uses additional paths between earlier and later layers in a dense block. DenseNets have several compelling advantages. They alleviate the vanishing-gradient problem, strengthen feature propagation, encourage feature reuse, and substantially reduce the number of parameters[35]. DeepMM also employs a hard parameter-sharing multi-task learning method, which can greatly reduces the risk of overfitting[39]. The first network (i.e. DenseNet A) is used to simultaneously predict the main-chain probability and Cα probability of a grid point. The second network (i.e. DenseNet B) is used to predict the amino acid type and secondary structure type of a main-chain local dense point (LDP). The

input for the DenseNet A are voxels of size $11\text{Å} \times 11\text{Å} \times 11\text{Å}$. The second network (DenseNet B) takes the voxels of size $10\text{Å} \times 10\text{Å} \times 10\text{Å}$ as input because main-chain points are not always on the grid after mean shift. For each voxel, the density values are normalized to the range of [0, 1] according to the maximum and minimum density values in the voxel. 3D convolutions and 3D pooling layers are used instead of their 2D counterparts used in traditional image processing because the density maps have three dimensions. Several dense blocks are used in both networks, each of which consists of eight densely connected layers. For DenseNet A, the first two dense blocks are shared by both tasks, whereas for DenseNet B, only one shared block is adopted. After the shared blocks, each task employs two task-specific blocks and gives the final prediction. The details of network architecture are provided in Supplementary Table 1.

2.1.2 Alignment of sequence to main-chain

The Smith-Waterman dynamic programming (DP) algorithm[27] is used to align the target sequence to the predicted main-chain path. The predicted Cα probability value, amino acid type, and secondary structure type are assigned to each point of the main-chain. Instead of using 20 amino acid types, amino acids are grouped into four classes according to their sizes, shapes, and distributions in EM density maps (Figure 2**d**). Secondary structures are categorized into three types of Helix, Sheet, and Coil. The match between the target sequence and main-chain path is evaluated by two scoring matrices for amino acid and secondary structure, respectively (Figure 2**b**). Namely, a target residue is more likely to be aligned to a main-chain point with the same amino acid type, the same secondary structure type, and a higher Cα probability, and vice versa. For each combination of parameters in the main-chain tracing procedure, 160 Cα models are generated. Finally, all the generated Cα models are ranked by their alignment scores. The details of the alignment procedure can be found in the **Methods** section.

2.2 Model reconstruction for simulated EM maps

We first evaluated the performance of our DeepMM algorithm on the test set of 40 simulated density maps at 5 Å resolution. DeepMM traced the main-chain of protein on the predicted main-chain

probability map rather than the raw EM density map. Thus, the generated Cα models by our DeepMM are closer to the native structures with fewer search trajectories and steps. For each of the 40 maps, DeepMM built 160 Cα models, which were ranked by their alignment scores.

Figure 3 shows a comparison of the predicted Cα models for the protein chains of different lengths by DeepMM and MAINMAST. The detailed results are provided in Supplementary Table 2. It can be seen from the figure that our DeepMM method obtained a much better performance than MAINMAST. As shown in Figure 3**a**, DeepMM built significantly more accurate Cα models, and achieved an average Cα RMSD of 0.54 Å when the top scored model was considered, compared to 1.79 Å for MAINMAST. DeepMM also generated high-quality models with less than 1.0 Å Cα RMSD for all of the 40 maps, compared with only one such model by MAINMAST. Moreover, DeepMM achieved the high-accuracy models with less than 0.5 Å RMSD for 22 of 40 maps, whereas MAINMAST failed to generate any model with $<$ 0.5 Å RMSD (Figure 3a). The program CLICK[40] was also used to evaluate the accuracy of the Cα models built by DeepMM and MAINMAST. The corresponding results are shown in Figure 3**b**. Similar to the results of Cα RMSD comparison, DeepMM generated many more high-quality models according to the CLICK RMSD criterion and achieved an average CLICK RMSD of 0.53 Å when the top model was considered, compared to 2.18 Å for MAINMAST. In addition, DeepMM also achieved a significantly higher structure overlap than MAINMAST (Figure 3**c**). Except for two top scored models with 99.75% and 99.44% structure overlap, the rest 38 top models generated by DeepMM all have a 100% structure overlap. On average, DeepMM obtained a high structure overlap of 99.98%, compare to 81.88% for MAINMAST. Figure 3 also reveals that DeepMM generated consistently high-accuracy models for all the proteins of different lengthes, whereas MAINMAST tended to perform worse with the increasing number of residues in the protein, suggesting the higher robustness of DeepMM than MAINMAST.

2.3 Model reconstruction for experimental EM maps

Our DeepMM method was further tested on the benchmark of 30 experimental density maps at 2.6-4.8 Å resolution. For each of the 30 experimental density maps, DeepMM built 4320 protein Cα models, which were then ranked by their alignment scores.

Figure 4**a** shows a comparison of the Cα RMSDs for the models built by DeepMM and MAIN-

MAST. The corresponding data are provided in Supplementary Table 3. It can be seen from the figure that DeepMM generated significantly more accurate models than MAINMAST. On average, DeepMM obtained a Cα RMSD of 10.7 Å for the top scored models, which is much better than 22.4 Å by MAINMAST. Moreover, DeepMM predicted a model of $<$ 10 Å for 18 out of 30 top scored models, of which 14 models are within 5.0 Å Cα RMSD. By contrast, only 7 and 4 models are within 10.0 Å and 5.0 Å for MAINMAST, respectively. Figure 4**b** shows a comparison of the results for the models predicted by DeepMM and RosettaES. It can be seen from the figure that DeepMM performed much better and generated many more accurate models than RosettaES. Compared to 18 models within 10 Å RMSD by DeepMM, only six models were predicted within 10.0 Å RMSD by RosettaES for the top predictions. On average, Rosetta obtained an average Cα RMSD of 27.0 Å, which is much higher than 10.7 Å for DeepMM.

Figure 4**c** shows the Cα RMSDs of the top scored models by our DeepMM method at different resolutions. It can be seen from the figure that no significant correlation is found between the map resolution and Cα RMSD for the top 1 and 10 scored models, respectively, suggesting the general applicability of our DeepMM method. Further examination of the predicted results also reveals that the model accuracy depends more on the quality than on the resolution of a map. Namely, compared to maps with relatively higher resolution but lower quality like EMD-3246A/B (2.8 Å) and EMD-5495 (3.5 Å), maps with relatively lower resolution but higher quality like EMD-2867 (4.3 Å) and EMD-3073 (4.1 Å) are more likely to be successful in reconstructing a correct model. This phenomenon can be attributed to the fact that resolution is a global estimation and resolvability is not necessarily uniform throughout the whole map[41].

The final step of the DeepMM workflow is to construct and refine all-atom structures from predicted DeepMM Cα models[36–38]. To investigate the impact of this modeling process, Figure 4**d** gives the RMSDs of DeepMM Cα models and final all-atom structures. It can be seen from the figure that there is almost no difference between the RMSDs for the DeepMM Cα models and final all-atom structures, demonstrating the accuracy of the Cα models predicted by our DeepMM.

Figure 5 gives two examples of successfully reconstructed structures by DeepMM. One example, EMD-2867, which is a nucleoprotein at 4.3 Å resolution, was successfully reconstructed by DeepMM, as shown in Figure **5a**. It can be seen from the figure that the predicted main-chain by

DeepMM overlaps well with that of the deposited structure. Accordingly, the predicted model shows an atomic-accuracy with a Cα RMSD of 3.1 Å. Figure **5b** shows the results of another example, EMD-6272, which is the bovine rotavirus VP6 at 2.6 Å resolution. Because of its high resolution, DeepMM predicted a very high accurate model with a small Cα RMSD of 1.7 Å. Correspondingly, the constructed full-atom model by DeepMM shows an excellent overlap with the deposited structure (Figure **5b**).

2.4 Evaluation of DeepMM on the EMDB-wide data set

To investigate the accuracy and general applicability of our DeepMM method, we have further evaluated the performance of DeepMM on a large test set of EMDB-wide experimental maps. This large test set consists of 2931 diverse EM maps with 2.6-4.9 Å resolutions from the EMDB that have associated structures in the PDB (See the **Methods** section). For each of the 2931 test cases, our DeepMM method was conducted to reconstruct structures using four combinations of parameters, yielding 640 models for each case. Figure 6 shows a summary of the results predicted by DeepMM. The corresponding data are provided in Supplementary Table 4. Two metrics, RMSD and TMscore, were used to evaluate the overall accuracy of predicted models. On average, DeepMM achieved a Cα RMSD of 9.8 Å for the top prediction and 8.4 Å for the top 10 predictions on this test set of 2931 maps. The corresponding average TM-scores are 0.648 and 0.694 for top 1 and top 10 predictions, suggesting the high accuracy of our DeepMM approach.

Figure **6a** shows the percentage of the predicted models at different Cα RMSD cutoffs. It can be seen from the figure that 53.6% of the top models built by DeepMM are within 10 Å Cα RMSD. For the top 10 scored predictions, 59.9% of the cases have an RMSD of less than 10 Å. The percentage of the models with different TM-score cutoffs are showed in Figure **6b**. It can be seen from the figure that 65.6% of the top models built by DeepMM have a TM-score of > 0.5. When the top 10 models were considered, the corresponding percentage increased to 73.6%. Comparing the results in Figures **6a** and **b** also reveals that the percentages for TM-score are significantly higher than those for Cα-RMSD, suggesting that the models built by DeepMM still share the same fold with native structure even if they have a large Cα RMSD.

Figure **6c** shows the percentage of correctly predicted top models (i.e. within 10 Å Cα RMSD)

at different resolutions. For EM maps at 2.5-3.0 Å resolution, DeepMM achieved an excellent performance in successfully reconstructing a correct model, and achieved a success rate of 95.3% and 96.2% for the top 1 and 10 scored models, respectively. The performance of DeepMM decreased with the decreasing map resolution. Specifically, for the EM maps with a resolution of 3.0-3.5 Å, 3.5-4.0 Å, and 4.0-4.5 Å, DeepMM obtained a success rate of 82.5%/87.3%, 53.2%/62.1%, and 22.9%/29.8% for the top 1/10 predictions, respectively. For EM maps with a resolution of 4.5 Å or worse, it is challenging for DeepMM to build correct models. On average, for the maps at 3-5 Å resolution, DeepMM gave a success rate of 50.3% and 57.0% in reconstructing a correct model within 10 Å Cα-RMSD for the top 1 and 10 predictions, respectively. Figure **6d** shows the percentage of correctly predicted top models using the criterion of TM-score > 0.5 in different resolution ranges. Similar trends in Figure **6c** can be observed in Figure **6d**. Specifically, for the maps with a resolution of 2.5-3.0 Å, 3.0-3.5 Å, 3.5-4.0 Å, 4.0-4.5 Å, and 4.5-5.0 Å, DeepMM achieved correct models with a TMscore of > 0.5 for 99.1%/99.5%, 90.7%/94.2%, 69.5%/79.6%, 36.9%/49.1%, and 15.0%/23.5% of the test cases when the top 1/10 predictions were considered, respectively. On average, for the maps at 3-5 Å resolution, DeepMM obtained a success rate of 63.0% and 71.6% in building a model with TMscore > 0.5 for the top 1 and 10 predictions, respectively.

Next, DeepMM was compared with Phenix on this test set, where the Phenix models were generated using the *phenix.map_to_model* tool[22] in the Phenix package (version 1.18.2-3874). Two metrics calculated by *phenix.chain_comparison* were used to evaluate the accuracy of a model. One is the fraction of the CA atoms in one model matching the CA atoms in another model within 3.0 Å regardless of their residue names (i.e. coverage or residue match). The other is the percentage of the sequence in the target structure reproduced by the query model (i.e. specificity of sequence match). A model with a high percentage of residue match may have a very low percentage of sequence match because of mismatching of residue names. Figures 7**a** and **b** show the percentages of protein residues and the sequence reproduced by DeepMM and Phenix at different resolutions. Figures 7**c** and **d** give the histograms of corresponding average values at different resolutions. It can be seen from the figure that DeepMM achieved a significantly better performance than Phenix in both residue match and sequence match, especially for those maps at low resolutions. For the maps at resolutions better than 3.0 Å, 94.2% of protein residues in the deposited structures were reproduced by our DeepMM method,

compared to 84.7% by Phenix. The corresponding average sequence match is 78.0% for our DeepMM approach, which is much higher than 59.7% for Phenix. For the maps at 3-5 Å resolution, the average residue match for DeepMM is 80.7%, compared with 65.0% for Phenix. The corresponding average sequence match is 38.1% for DeepMM, which is much higher than 19.2% for Phenix. Given that the prediction of sequence match is much more challenging than that of residue match, the much better performance of DeepMM than Phenix in sequence match demonstrated the atomic-accuracy of the model built by DeepMM.

It is worth mentioning that DeepMM can build fully-connected, full-length all-atom protein models, whereas Phenix is designed to build initial models of structure fragments. Figure 8 shows the protein models built by DeepMM and Phenix for one example, Chain A of 6DW1, part of a GABAA receptor at 3.1 Å resolution. The deposited structure with its associated EM density map (EMD-8923) is displayed in panel **a**. Figures 8**b** and **c** show the Phenix model and its superimposition with the deposited structure, respectively. It can be seen from the figures that the model built by Phenix consists of multiple fragments without showing any secondary structures, as expected. The predicted model by Phenix for this map had a residue match of 86.7%, but gave a very low sequence match of 9.9%. Therefore, although Phenix recovered most parts of the target protein structure from the EM density map, it assigned wrong residue names for most of the modeled fragments because its low sequence match, as shown in Figure 8**c**. In contrast, DeepMM built an excellent all-atom structure for this map, with a near-perfect residue match of 97.1% and a high sequence match of 86.8%. Therefore, the model predicted by DeepMM reproduced most of the secondary structures and had an almost identical chain trace to the deposited structure(Figure 8**d**). The corresponding amino acid names were also assigned correctly by our DeepMM approach (Figure 8**e**).

3 Discussion

With the rapid development in the cryo-EM technology, cryo-EM is becoming a routine technique for structure determination of biological macromolecules. More and more cryo-EM maps are being generated, many of which are expected to have near-atomic resolutions (3-5 Å). In addition, an advantage of cryo-EM is its ability to obtain the EM maps of multiple macromolecules in one sam-

ple and therefore can be applied to structural proteomics for structure measurement in the cellular milieu[42]. Such application will generate a large number of near-atomic resolution EM maps in one experiment. Therefore, how to fast reconstruct structures for EM maps has become an urgent and challenging issue in the cryo-EM field. However, despite some progresses have been achieved in the past decade, existing approaches often involve costly manual intervention, rely on known templates, only generate partial structure fragments, or have low atomic-accuracy in the built models. These methods will not be able to meet the needs of accurately interpreting the increasingly large number of EM maps. Therefore, *de novo* approaches that can automatically reconstruct atomic-accuracy full-atom structures from EM maps are pressingly needed. Meeting the needs, we have developed a fully automatic *de novo* structure determination method for near-atomic resolution cryo-EM maps using a deep learning-based framework, named as DeepMM. Our DeepMM approach can automatically reconstruct complete all-atom protein structures for EM maps with atomic-accuracy. DeepMM was extensively validated on diverse benchmarks and it outperformed current state-of-the-art approaches including RosettaES, MAINMAST, and Phenix.

DeepMM has also been evaluated on an EMDB-wide large test set of 2931 experimental maps at 2.6-4.9 Å resolution. Overall, DeepMM was able reconstruct correct models for over 60% of the test cases. To our knowledge, this is the first time to build full-atom protein structures for a database-wide EM maps. As a comparison, the MAINMAST, which is most similar to DeepMM, was only tested on 40 simulated maps and 30 experimental maps at 2.6-4.8 Å resolution. RosettaES was evaluated on a benchmark of only nine proteins at 3-5 Å resolution. The *phenix.map_to_model* tool in Phenix, which builds initial models of structure fragments, was tested on a large data set of 476 EMDB maps with resolution of 4.5 Å or better. Therefore, the evaluation for our DeepMM method is much more extensive than that for similar approaches. Moreover, DeepMM outperformed similar approaches on all test sets, suggesting the accuracy and general applicability of our method.

Computational efficiency is another important parameter for an automatic approach, given the increasingly large number of maps. DeepMM is fast and able to reconstruct an all-atom structure from an EM map within 1 hr on a single-GPU machine for an average-length protein chain of 300 amino acids. As a comparison, RosettaES takes 1.5 hr on a 16-core machine for a 30-residue segment[17] and MAINMAST consumes about 100 CPU hr for a protein chain of 300 amino acids[23].

In summary, we have proposed a deep learning-based automatic *de novo* method, DeepMM, to fast reconstruct full-length all-atom protein structures for near-atomic resolution EM maps. Our method was extensively evaluated on large diverse test sets of experimental maps and compared with state-of-the-art approaches. Given the high computational efficiency and all-atomic accuracy, it is anticipated that DeepMM will serve as an indispensable tool for automatic atomic-accuracy structure determination for cryo-EM maps.

4 Methods

4.1 Training the DenseNets of DeepMM

All the training parameters and procedure used for simulated EM maps are essentially the same to the parameters and procedure used for experimental EM maps unless otherwise specified.

For DenseNet A, all the grid points above a density value D_0 were used for training, where D_0 was set to 1.0 for simulated maps at 5.0 Å resolution. For experimental maps, D_0 was set to 1/2 of its recommended contour level. The labels (main-chain probability and Cα probability) of a grid point $\vec{a}$ were calculated as follows:

$$P_{\vec{a}}^{\vec{X}} = \min\{e^{-\frac{\|\vec{a}-\vec{X}\|^2}{r_0^2}}, \forall \vec{X} \in \|\vec{a}-\vec{X}\| < r_{\text{cut}}\} \tag{1}$$

where X stands for the N, C, or Cα atoms. The r_0 is the radius at which the probability drop to $1/e$. If no atom is within r_{cut} of a grid point, the corresponding probability is set to 0. A total of 512 voxels were trained in one batch and 30 epochs were trained for the whole data set. The Adam optimizer with an initial learning rate of 0.001 was used to minimize the mean absolute error (MAE). Learning rate decay was adopted, where the learning rate was reduced to 1/10 of the current value after every 10 epochs. To avoid over-fitting, the $weight_decay$ parameter of Adam optimizer was set to 1e-6 as the L2 regularization.

For DenseNet B, one point was randomly sampled within 1.0 Å for every main-chain atom in the training set. The corresponding amino acid type and second structure type marked by STRIDE[43] were assigned to each point. Twenty types of amino acids were grouped into four classes according to their sizes, shapes and distributions in their EM density maps[42], as illustrated in Figure 2**d**. Specifically,

GLY, ALA, SER, CYS, VAL, THR, ILE and PRO are grouped as Class I. LEU, ASP, ASN, GLU, GLN and MET are grouped as Class II. LYS and ARG are grouped as Class III. HIS, PHE, TYR and TRP are grouped as Class IV. Residues that have structure codes of H, G, or I by STRIDE were labelled as "Helix", those with codes of B/b or E were labelled as "Sheet", and the other residues were labelled as "Coil". All the training parameters were identical to those for DenseNet A except for using CrossEntropyLoss as loss function.

4.2 Tracing the main-chain path

The main-chain tracing algorithm in MAINMAST[23] was used to trace the main-chain path in the main-chain probability map. First, local dense points (LDPs) are identified using the mean shift algorithm. For a 3D main-chain probability map with N grid points $x_i (i = 1, \cdots, N)$ and corresponding probability values $\Phi(x_i)(i = 1, \cdots, N)$, M points $y_j^{(0)} (j = 1, \cdots, M)$ are selected as seed points with the probability value of no less than a threshold Φ_{thr}. The mean shift algorithm iteratively shifts the initial grid points $y_j^{(0)}$ towards the local highest probability by computing the weighted average of probability values as follows.

$$y_j^{(t+1)} = f(y_j^{(t)}) \tag{2}$$

$$f(y_j^{(t+1)}) = \frac{\sum_{n=1}^{N} k(y_j^{(t)} - x_n)\theta(x_n)x_n}{\sum_{n'=1}^{N} k(y_j^{(t)} - x_{n'})\theta(x_{n'})} \tag{3}$$

where $k(d)$ is the probability value of the grid point x and $\theta(x)$ is a Gaussian kernel function defined as

$$k(d) = \exp(-1.5\|\frac{d}{\sigma}\|^2) \tag{4}$$

where σ is the bandwidth and set to 1.0 in this study. If the number of LDPs in a main-chain path is less than the number of protein residues, a smaller σ value of 0.75 will be adopted. After the point position $y_j^{(t)}$ is updated, the probability value $\Theta(y_j^{(t)})$ of the point is computed as

$$\Theta(y_j^{(t)}) = \frac{1}{N}\sum_{n=1}^{N} k(y_j^{(t)} - x_n)\Phi(x_n) \tag{5}$$

The probability is further normalized to the range of [0, 1] as

$$\theta(y_j^{(t)}) = \frac{\Theta(y_j^{(t)}) - \Theta_{\min}}{\Theta_{\max} - \Theta_{\min}} \tag{6}$$

where $\Theta_{\max}$ and $\Theta_{\min}$ are the maximum and minimum probability values of all the seed points, respectively. Points with normalized probability values below a certain threshold (θ_{thr}) are discarded. Points that are within a threshold distance of 0.5 Å are clustered, and the point with the highest probability in the cluster is chosen as the representative, called LDP.

The next step is to connect LDPs into a minimum spanning tree (MST) and iteratively refine the tree structure with a Tabu search method. The edge between two LDPs within a certain distance d_{keep} is kept unchanged and only edges in the local MSTs defined by a radius of r_{local} are considered during refinement. After N_{round} steps of Tabu search, the longest path of the refined tree is traced as the main-chain path[23].

4.3 Aligning target sequence to main-chain path

The Smith-Waterman (SW) dynamic programming (DP) algorithm[27] was used to align the target sequence to the generated main-chain path. The alignment protocol is shown in Figures **2a**, **b** and **c**. The n residues $\{A_i(i = 1, ...n)\}$ in the protein are aligned to m LDPs $\{L_j(j = 1, ...m)\}$ in the main-chain path. The matching score $M(i, j)$ for a pair of A_i and L_j is computed as follows.

$$M(i,j) = w_{\text{AA}} M_{\text{AA}}(T_{\text{AA}}(A_i), T_{\text{AA}}(L_j)) + w_{\text{SS}} M_{\text{SS}}(T_{\text{SS}}(A_i), T_{\text{SS}}(L_j)) \tag{7}$$

where M_{AA} and M_{AA} are the scoring matrices for amino acid and secondary structure matching[42,44], respectively. For a residue A_i, the amino acid type is one of the four amino acid classes ($T_{\text{AA}}(A_i) = 1, 2, 3, 4$). The predicted amino acid type for an LDP L_j is also one of the four amino acid classes ($T_{\text{AA}}(L_i) = 1, 2, 3, 4$). Similarly, the secondary structure matching score is calculated using the secondary structure type predicted from the sequence ($T_{\text{SS}}(A_i) = 1, 2, 3$) by SPIDER2[45] and secondary structure type predicted on LDPs ($T_{\text{SS}}(L_i) = 1, 2, 3$). The scoring matrices M_{AA} and M_{SS} used in the alignment are shown in Figure **2b**. The w_{AA} and w_{SS} are the weights for corresponding matching scores and set to 1.0 and 0.5, respectively. With the calculated matching score $M(i, j)$, an alignment is calculated with the follow rule to form a DP matrix, F, as follows.

$$F(i,j) = \max \begin{cases} F(i-1,j) + gap \\ F(i-1,j-1) - w_{\text{C}\alpha-\text{C}\alpha}|d_{\text{std}} - d| + w_{\text{C}\alpha} P_{\text{C}\alpha}(j) + M(i,j) \\ F(i,j-1) \end{cases} \tag{8}$$

where gap is the gap penalty for unassigned residues in the protein sequence. To ensure a full-length structure reconstruction, gap is set to -10000.0 so as to forbid skipped residues. The $|d_{\mathrm{std}} - d|$ is the penalty score for Cα-Cα distance, where d_{std} is the standard Cα-Cα distance and d is the distance between LDP L_j and the last aligned LDP. The $P_{\mathrm{C}\alpha}(j)$ is the predicted Cα probability for LDP L_j. The $w_{\mathrm{C}\alpha-\mathrm{C}\alpha}$ and $w_{\mathrm{C}\alpha}$ are the weights for the corresponding scores. Here, $w_{\mathrm{C}\alpha}$ is set to 1.6, and $w_{\mathrm{C}\alpha-\mathrm{C}\alpha}$ is set to 1.0, 0.7, and 0.8 for "Helix", "Sheet", and "Coil", respectively. The final alignment scores are used to rank the generated models.

4.4 Parameter settings of DeepMM

The parameters of mean-shift, MST construction, and Tabu search are set to be the same to those in MAINMAST[23], unless otherwise specified. DeepMM also employs several parameter combinations to generate multiple Cα models for one EM map. For each combination of parameters, 10 trajectories of Tabu search are carried out, yielding 10 main-chain paths. Since DeepMM starts from the main-chain probability map, fewer parameter combinations are needed to reconstruct reliable 3D structures. For both simulated and experimental maps, the thresholds of probability (Φ_{thr}) and normalized probability (θ_{thr}) are both set to 0. For the 40 simulated maps, only one parameter combination is adopted. Specifically, the maximum number of Tabu search steps (N_{round}) is set to 100, the sphere radius of local MST (r_{local}) is set to 5.0 Å, and the constraint for the length (d_{keep}) is set to 0.5 Å. For the 30 experimental maps, we employ the following 27 combinations of parameters: the sphere radius of local MST (r_{local}=5.0, 7.5, 10.0 Å), the edge weight threshold (d_{keep}=0.5, 1.0, 1.5 Å), and the maximum number of the Tabu search steps (N_{round}=2500, 5000, 7500). For the extended EMDB-wide test set of 2931 maps, we employ the following 4 combinations of parameters: the edge weight threshold (d_{keep}=0.5, 1.0 Å) and the maximum number of the Tabu search steps (N_{round}=2500, 5000). The sphere radius of local MST (r_{local}) is set to 10 Å. For each of the generated main-chain path, 16 Cα models are generated using 8 different standard Cα-Cα distances (d_{std}=3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8 Å) on two sequence directions. Namely, 160 models (16 models for each of the 10 trajectories) are constructed for each parameter combination. The Cα models are ranked by their alignment scores. An RMSD cutoff of 5 Å is used to remove the one with lower alignment score in two similar structures. Finally, the top 10 scored protein Cα models are selected to build the all-atom structures.

4.5 Datasets used

4.5.1 Training sets

Two datasets, simulated EM map set and experimental EM map set, were used to train our DeepMM method for simulated maps and experimental maps, respectively.

For simulated EM maps, 2000 representative structures for different superfamilies in the SCOPe database[46] were taken from Emap2sec[33] as training set. Those structures were removed from the training set if they have a TM-score[47] of over 0.5 with any structure in the test set. To save the computational cost, only 100 randomly selected structures from the training set were retained. Next, we used the *e2pdb2mrc.py* program from the EMAN2 package (version 2.11)[48] to generate the simulated EM maps at 5.0 Å resolution and 1.0 Å grid interval for each structure in training and test set. The training SCOPe entries used in this study were listed in Supplementary Table 5.

For experimental EM maps, all the EM density maps at 2-5 Å resolution that have associated PDB models were downloaded from the EMDB. As of December 26, 2019, 2546 EM maps were collected. Any PDB structure and its corresponding EM map that met the following criteria were removed: (i) including nucleic acids, (ii) missing side-chain atoms, (iii) including "HETATM" residues, (iv) including "UNK" residues, (v) including more than 1 subunit (MODEL), and (vi) including less than 50 or more than 300 residues. Then, 1588 chains from the remaining 361 experimental EM maps were clustered with 50% sequence identity using CD-HIT[49], yielding a total of 1340 chains. To ensure a valid evaluation, chains were removed from training set if they have over 30% sequence identity with any chain in the test set. Each protein chain was zoned out from the whole map using a distance of 4.0 Å[23]. For good quality maps, protein chain and its associated map should have sufficient structural agreement. The cross-correlation between the experimental map and the simulated map density at the same resolution with the experimental map generated from the structure was calculated using the UCSF chimera[50]. Only the chains with a cross-correlation of over 0.65 were kept[33]. The final training set consists of 100 non-redundant protein chains. The grid intervals for experimental maps were unified to 1.0 Å using trilinear interpolation. The training EM maps and their corresponding PDB chains used in this study are listed in Supplementary Table 6.

4.5.2 Test sets

Three test sets were used to evaluate our DeepMM approach for its accuracy and general applicability, including one simulated map set and two experimental maps.

The simulated map set was taken from the test set of 40 simulated maps used by MAINMAST[23]. The maps were generated at 5.0 Å resolution with a grid spacing of 1.0 Å using the $e2pdb2mrc.py$ program in the EMAN2 package[48].

The first experimental test set is the benchmark of 30 EM maps at 2.6-4.8 Å resolution, which have been used to evaluate MAINMAST[23]. The corresponding EM maps were downloaded from the EMDB, For each EM map, a single subunit was zoned out from the whole density map at a distance cutoff of 4.0 Å.

In addition, to evaluate the accuracy and general applicability of DeepMM, we have also constructed a large test set of EMBD-wide experimental maps. The generation procedure of this set was similar to that for the experimental training set. Specifically, for each chain of the EM PDB structure at 2.5-5.0 Å resolution and no more than one subunit (MODEL) from the EMDB, a single density patch was zoned out from the whole density map at a distance cutoff of 4.0 Å. Any protein chain and its corresponding EM map patch that met the following situation were removed: (i) including nucleic acids, (ii) missing side-chain atoms, (iii) including "HETATM" residues, (iv) including "UNK" residues, (v) including less than 50 or equal or more than 300 residues, (vi) having over 30% sequence identity to any chain in the training set. The cross-correlation between the experimental map and the simulated density map at the same resolution generated from the structure should be over 0.65[33]. Each protein chain was zoned out from the whole map using a distance of 4.0 Å[23]. The finial test set consists of 2931 protein chains, which are listed in Supplementary Table 4.

Acknowledgements

The authors acknowledge professor Daisuke Kihara and his students Genki Terashi and Sai Raghavendra Maddhuri Venkata Subramaniya from Purdue University for providing their datasets. This work was supported by the National Natural Science Foundation of China (grant No. 31670724) and the startup grant of Huazhong University of Science and Technology.

Author contributions

S.H. conceived and supervised the project. J.H. and S.H. designed and performed the experiments. J.H. and S.H. wrote the manuscript.

Competing interests

The authors declare no competing interests.

Code availability

The DeepMM program is freely available for academic use through http://huanglab.phys.hust.edu.cn/DeepMM/.

Data availability

The raw data of the structure models reconstructed by our method are provided in the Supplementary Information. The data that support the findings of this study are available from the corresponding author upon request.

References

(1) Nogales E. (2016). The development of cryo-EM into a mainstream structural biology technique. *Nature methods* **13(1)**, 24-27.

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

(3) Cheng Y. (2018). Single-particle cryo-EM-How did it get here and where will it go. *Science* **361(6405)**, 876-880.

(4) Safdari, H. A., Pandey, S., Shukla, A. K., & Dutta, S. (2018). Illuminating GPCR Signaling by Cryo-EM. *Trends in cell biology* **28(8)**, 591-594.

(5) Luque, D., & Caston, J. R. (2020). Cryo-electron microscopy for the study of virus assembly. *Nature chemical biology* **16(3)**, 231-239.

(6) Renaud, J. P., Chari, A., Ciferri, C., Liu, W. T., Remigy, H. W., Stark, H., & Wiesmann, C. (2018). Cryo-EM in drug discovery: achievements, limitations and prospects. *Nature reviews Drug discovery* **17(7)**, 471-492.

(7) Li, X. et al. Electron counting and beam-induced motion correction enable near-atomic-resolution single-particle cryo-EM. *Nat. Methods* **10**, 584-590 (2013).

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

(9) Scheres, S. H. W. RELION: implementation of a bayesian approach to cryo-EM structure determination. *J. Struct. Biol.* **180**, 519-530 (2012).

(10) Adams, P. D. et al. PHENIX: a comprehensive Python-based system for macromolecular structure solution. *Acta Crystallogr. D.* **66**, 213-221 (2010).

(11) Patwardhan A. Trends in the Electron Microscopy Data Bank (EMDB). *Acta Crystallogr D.* **73**, 503-508, (2017).

(12) Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., & Bourne, P. E. (2000). The Protein Data Bank. *Nucleic acids research* **28(1)**, 235-242.

(13) Alnabati, E., & Kihara, D. (2019). Advances in Structure Modeling Methods for Cryo-Electron Microscopy Maps. *Molecules* **25(1)**, 82.

(14) Lindert, S., Staritzbichler, R., Wötzel, N., Karakaş, M., Stewart, P. L., & Meiler, J. (2009). EM-fold: De novo folding of alpha-helical proteins guided by intermediate-resolution electron microscopy density maps. *Structure* **17(7)**, 990-1003.

(15) Baker, M. L., Abeysinghe, S. S., Schuh, S., Coleman, R. A., Abrams, A., Marsh, M. P., Hryc, C. F., Ruths, T., Chiu, W., & Ju, T. (2011). Modeling protein structure at near atomic resolutions with Gorgon. *Journal of structural biology* **174(2)**, 360-373.

(16) Wang, R. Y., Kudryashev, M., Li, X., Egelman, E. H., Basler, M., Cheng, Y., Baker, D., & DiMaio, F. (2015). De novo protein structure determination from near-atomic-resolution cryo-EM maps. *Nature methods* **12(4)**, 335-338.

(17) Frenz, B., Walls, A. C., Egelman, E. H., Veesler, D., & DiMaio, F. (2017). RosettaES: a sampling strategy enabling automated interpretation of difficult cryo-EM maps. *Nature methods* **14(8)**, 797-800.

(18) Baker, M. R., Rees, I., Ludtke, S. J., Chiu, W., & Baker, M. L. (2012). Constructing and validating initial Cα models from subnanometer resolution density maps with pathwalking. *Structure* **20(3)**, 450-463.

(19) Chen, M., Baldwin, P. R., Ludtke, S. J., & Baker, M. L. (2016). De Novo modeling in cryo-EM density maps with Pathwalking. *Journal of structural biology* **196(3)**, 289-298.

(20) Chen, M., & Baker, M. L. (2018). Automation and assessment of de novo modeling with Pathwalking in near atomic resolution cryoEM density maps. *Journal of structural biology* **204(3)**, 555-563.

(21) Terwilliger, T. C., Adams, P. D., Afonine, P. V., & Sobolev, O. V. (2018). A fully automatic method yielding initial models from high-resolution cryo-electron microscopy maps. *Nature methods* **15(11)**, 905-908.

(22) Terwilliger, T. C., Adams, P. D., Afonine, P. V., & Sobolev, O. V. (2020). Cryo-EM map interpretation and protein model-building using iterative map segmentation. *Protein science* **29(1)**, 87-99.

(23) Terashi, G., & Kihara, D. (2018). De novo main-chain modeling for EM maps using MAINMAST. *Nature Communications* **9(1)**, 1618.

(24) Terashi, G., & Kihara, D. (2018). De novo main-chain modeling with MAINMAST in 2015/2016 EM Model Challenge. *Journal of structural biology* **204(2)**, 351-359.

(25) Terwilliger, T. C., Sobolev, O. V., Afonine, P. V., & Adams, P. D. (2018). Automated map sharpening by maximization of detail and connectivity. *Acta crystallographica. Section D, Structural biology* **74(Pt 6)**, 545-559.

(26) Afonine, P. V., Poon, B. K., Read, R. J., Sobolev, O. V., Terwilliger, T. C., Urzhumtsev, A., & Adams, P. D. (2018). Real-space refinement in PHENIX for cryo-EM and crystallography. *Acta crystallographica. Section D* **74(Pt 6)**, 531-544.

(27) Smith, T. F., & Waterman, M. S. (1981). Identification of common molecular subsequences. *Journal of Molecular Biology* **147(1)**, 195-197.

(28) Rotkiewicz, P., & Skolnick, J. (2008). Fast procedure for reconstruction of full-atom protein models from reduced representations. *Journal of Computational Chemistry* **29(9)**, 1460-1465

(29) Trabuco, L. G., Villa, E., Mitra, K., Frank, J., & Schulten, K. (2008). Flexible fitting of atomic structures into electron microscopy maps using molecular dynamics. *Structure* **16(5)**, 673-683.

(30) Terashi, G., Kagaya, Y., & Kihara, D. (2020). MAINMASTseg: Automated Map Segmentation Method for Cryo-EM Density Maps with Symmetry. *Journal of chemical information and modeling* DOI: 10.1021/acs.jcim.9b01110, (2020).

(31) Tegunov, D., & Cramer, P. (2019). Real-time cryo-electron microscopy data preprocessing with Warp. *Nature methods* **16(11)**, 1146-1152.

(32) Chen, M., Dai, W., Sun, S. Y., Jonasch, D., He, C. Y., Schmid, M. F., Chiu, W., & Ludtke, S. J. (2017). Convolutional neural networks for automated annotation of cellular cryo-electron tomograms. *Nature methods* **14(10)**, 983-985.

(33) Maddhuri Venkata Subramaniya, S. R., Terashi, G., & Kihara, D. (2019). Protein secondary structure detection in intermediate-resolution cryo-EM maps using deep learning. *Nature methods* **16(9)**, 911-917.

(34) Si, D., Moritz, S. A., Pfab, J., Hou, J., Cao, R., Wang, L., Wu, T., & Cheng, J. (2020). Deep Learning to Predict Protein Backbone Structure from High-Resolution Cryo-EM Density Maps. *Scientific reports* **10(1)**, 4282.

(35) Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, 4700-4708.

(36) Xiang Z, Honig B. Extending the accuracy limits of prediction for side-chain conformations. *J Mol Biol.* 2001, **311(2)**, 421-430.

(37) Petrey D, Xiang Z, Tang CL, et al. Using multiple structure alignments, fast model building, and energetic analysis in fold recognition and homology modeling. *Proteins.* 2003, **53 Suppl 6**, 430-435.

(38) D.A. Case, V. Babin, J.T. Berryman, R.M. Betz, Q. Cai, D.S. Cerutti, T.E. Cheatham, III, T.A. Darden, R.E. Duke, H. Gohlke, A.W. Goetz, S. Gusarov, N. Homeyer, P. Janowski, J. Kaus, I. Kolossvary, A. Kovalenko, T.S. Lee, S. LeGrand, T. Luchko, R. Luo, B. Madej, K.M. Merz, F. Paesani, D.R. Roe, A. Roitberg, C. Sagui, R. Salomon-Ferrer, G. Seabra, C.L. Simmerling, W. Smith, J. Swails, R.C. Walker, J. Wang, R.M. Wolf, X. Wu and P.A. Kollman (2014), AMBER 14, University of California, San Francisco.

(39) Ruder, S. (2017). An overview of multi-task learning in deep neural networks. *arXiv preprint* arXiv:1706.05098.

(40) Nguyen, M. N., Tan, K. P., & Madhusudhan, M. S. (2011). CLICK–topology-independent comparison of biomolecular 3D structures. *Nucleic acids research* **39**(Web Server issue), W24-W28.

(41) Pintilie, G., Zhang, K., Su, Z., Li, S., Schmid, M. F., & Chiu, W. (2020). Measurement of atom resolvability in cryo-EM maps with Q-scores. *Nature methods* **17(3)**, 328-334.

(42) Ho, C. M., Li, X., Lai, M., Terwilliger, T. C., Beck, J. R., Wohlschlegel, J., Goldberg, D. E., Fitzpatrick, A., & Zhou, Z. H. (2020). Bottom-up structural proteomics: cryoEM of protein complexes enriched from the cellular milieu. *Nature methods* **17(1)**, 79-85.

(43) Heinig, M., & Frishman, D. (2004). STRIDE: a web server for secondary structure assignment from known atomic coordinates of proteins. *Nucleic acids research* **32**(Web Server issue), W500-W502.

(44) Wen Z, He J, Huang SY. Topology-independent and global protein structure alignment through an FFT-based algorithm. *Bioinformatics* **36**, 478-486, (2020).

(45) Heffernan, R., Dehzangi, A., Lyons, J., Paliwal, K., Sharma, A., Wang, J., Sattar, A., Zhou, Y., & Yang, Y. (2016). Highly accurate sequence-based prediction of half-sphere exposures of amino acid residues in proteins. *Bioinformatics (Oxford, England)* **32(6)**, 843-849.

(46) Fox, N. K., Brenner, S. E., & Chandonia, J. M. (2014). SCOPe: Structural Classification of Proteins–extended, integrating SCOP and ASTRAL data and classification of new structures. *Nucleic acids research* **42**(Database issue), D304-D309.

(47) Zhang, Y., & Skolnick, J. (2005). TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic acids research* **33(7)**, 2302-2309.

(48) Tang, G., Peng, L., Baldwin, P. R., Mann, D. S., Jiang, W., Rees, I., & Ludtke, S. J. (2007). EMAN2: an extensible image processing suite for electron microscopy. *Journal of structural biology* **157(1)**, 38-46.

(49) Fu, L., Niu, B., Zhu, Z., Wu, S., & Li, W. (2012). CD-HIT: accelerated for clustering the next-generation sequencing data. *Bioinformatics (Oxford, England)* **28(23)**, 3150-3152.

(50) Pettersen, E. F., Goddard, T. D., Huang, C. C., Couch, G. S., Greenblatt, D. M., Meng, E. C., & Ferrin, T. E. (2004). UCSF Chimera–a visualization system for exploratory research and analysis. *Journal of computational chemistry* **25(13)**, 1605-1612.

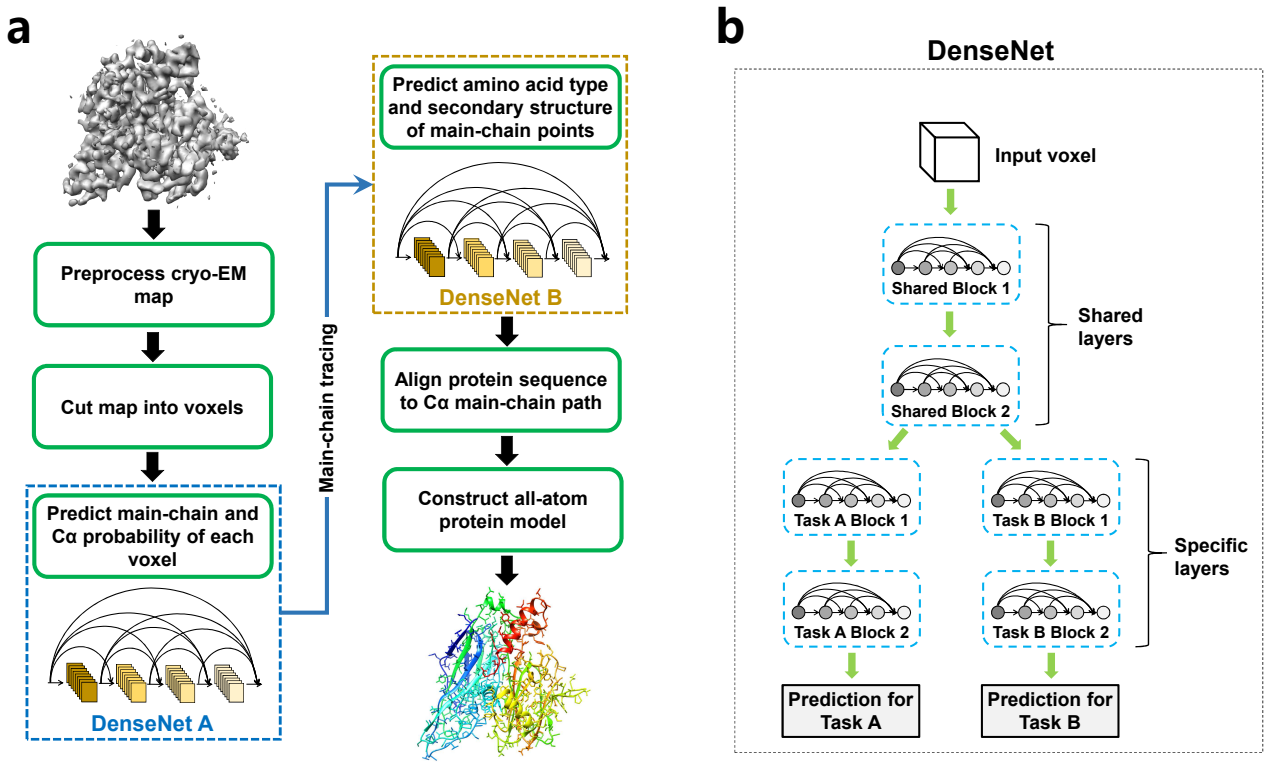

Figure 1: Workflow of our DeepMM method. a, The flowchart of DeepMM. DeepMM first predicts the main-chain and Cα probability of each density voxel using a Densely Connected Convolutional Network (DenseNet), and then traces the protein's main-chain path on the predicted main-chain probability map. Next, the amino acid and secondary structure types for each main chain point are predicted by a second DenseNet. The Cα models are generated by aligning the target sequence to the main-chain paths. Finally, the all-atom structures are constructed from the Cα models using the ctrip program and refined by an Amber energy minimization. **b,** The multi-task deep DenseNet architecture used in DeepMM. Starting from an input EM density voxel, two dense blocks are shared by both tasks in DenseNet A, while only one dense block is shared by both tasks in DenseNet B. Each prediction task employs two task-specific dense blocks and gives the final prediction.

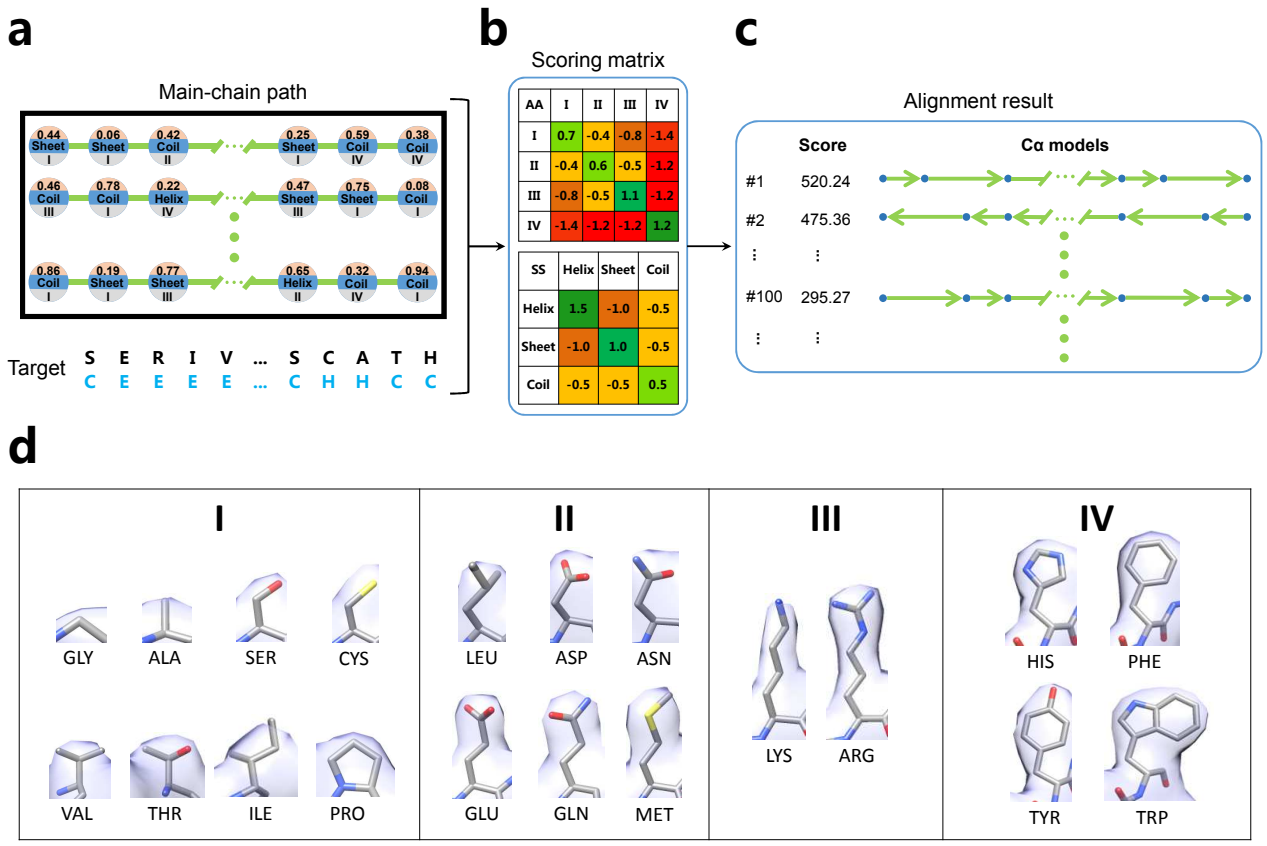

Figure 2: Alignment protocol between the target sequence and the predicted main-chain for DeepMM. a, DeepMM runs alignments of the target sequence of the EM map against each candidate main-chain path. Each sphere represents a predicted local dense point (LDP) on the main-chain path. Predicted information including the Cα probability (on the top), secondary structure (in the middle) and amino acid class (at the bottom) of LDPs is utilized during alignment. For the target sequence, its secondary structure is predicted by the SPIDER2 program, as illustrated in the sequence colored in azure under the amino acid sequence. **b,** Scoring matrices for amino acid type matching and secondary structure matching. **c,** The generated Cα models are ranked by their alignment score. **d,** Twenty amino acids are grouped into four classed according to the similarity of their side-chain EM densities.

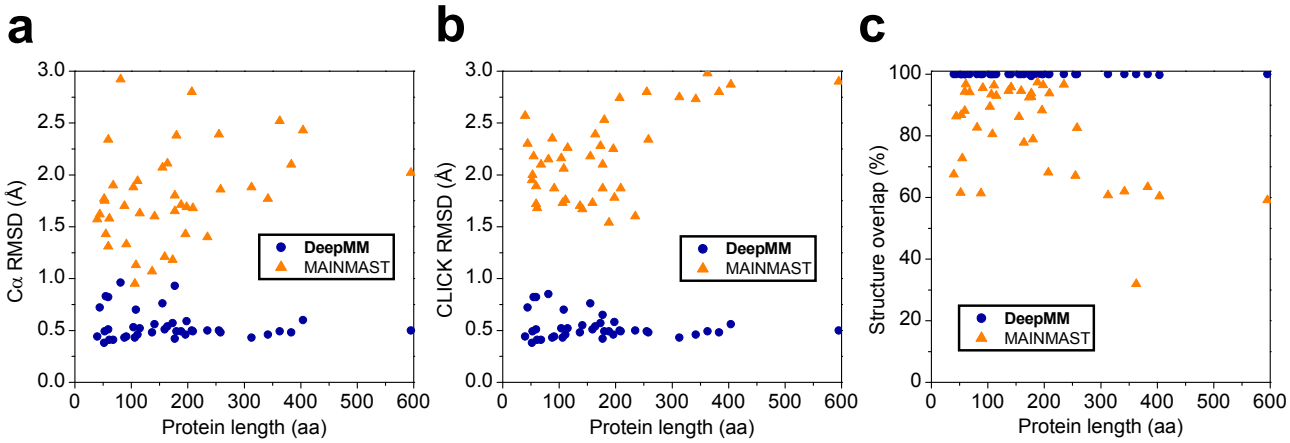

Figure 3: Comparison of the results by DeepMM and MAINMAST for the protein chains with different lengths. a, The Cα RMSDs of the top predicted models. **b,** The RMSDs of matched Cα atoms within 3.5 Å by the structure alignment tool CLICK. **c,** The structure overlap calculated by CLICK, which is defined as the fraction of matched Cα atoms.

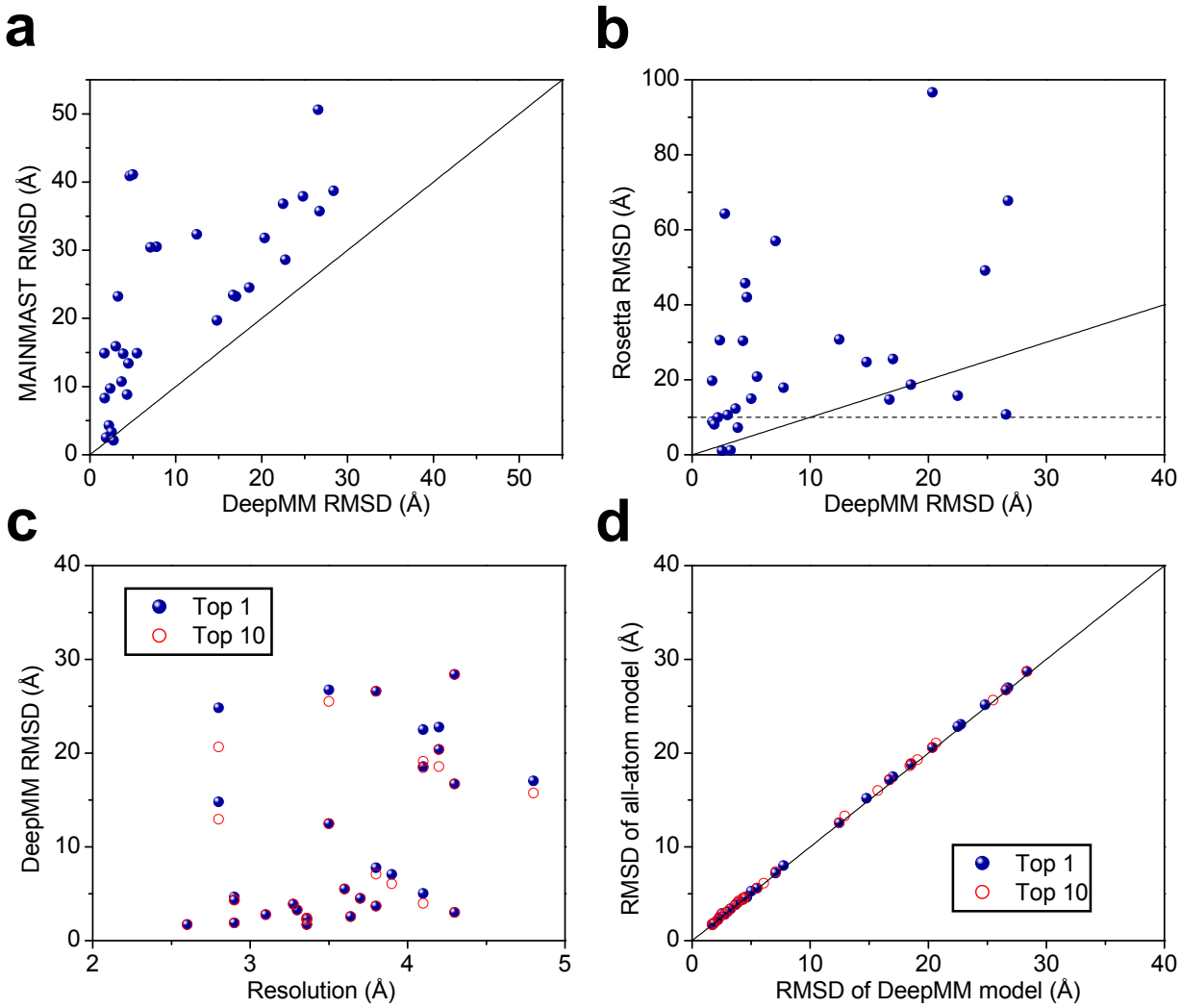

Figure 4: Modeling results of DeepMM on the test set of 30 experimental maps. The top scored models are displayed as solid spheres and the best model with least RMSD among the top 10 scored balls are displayed as empty circles. The solid line in the figure is the plot of $y = x$, and the dashed line stands for $y = 10$. **a,** Comparison of the models by DeepMM and MAINMAST in terms of Cα RMSD. **b,** Comparison of the models by DeepMM and Rosetta in terms of Cα RMSD. **c,** The Cα RMSDs at different map resolutions for the top scored models by DeepMM. **d,** Comparison of DeepMM Cα models and final all-atom structures before and after side-chain modeling.

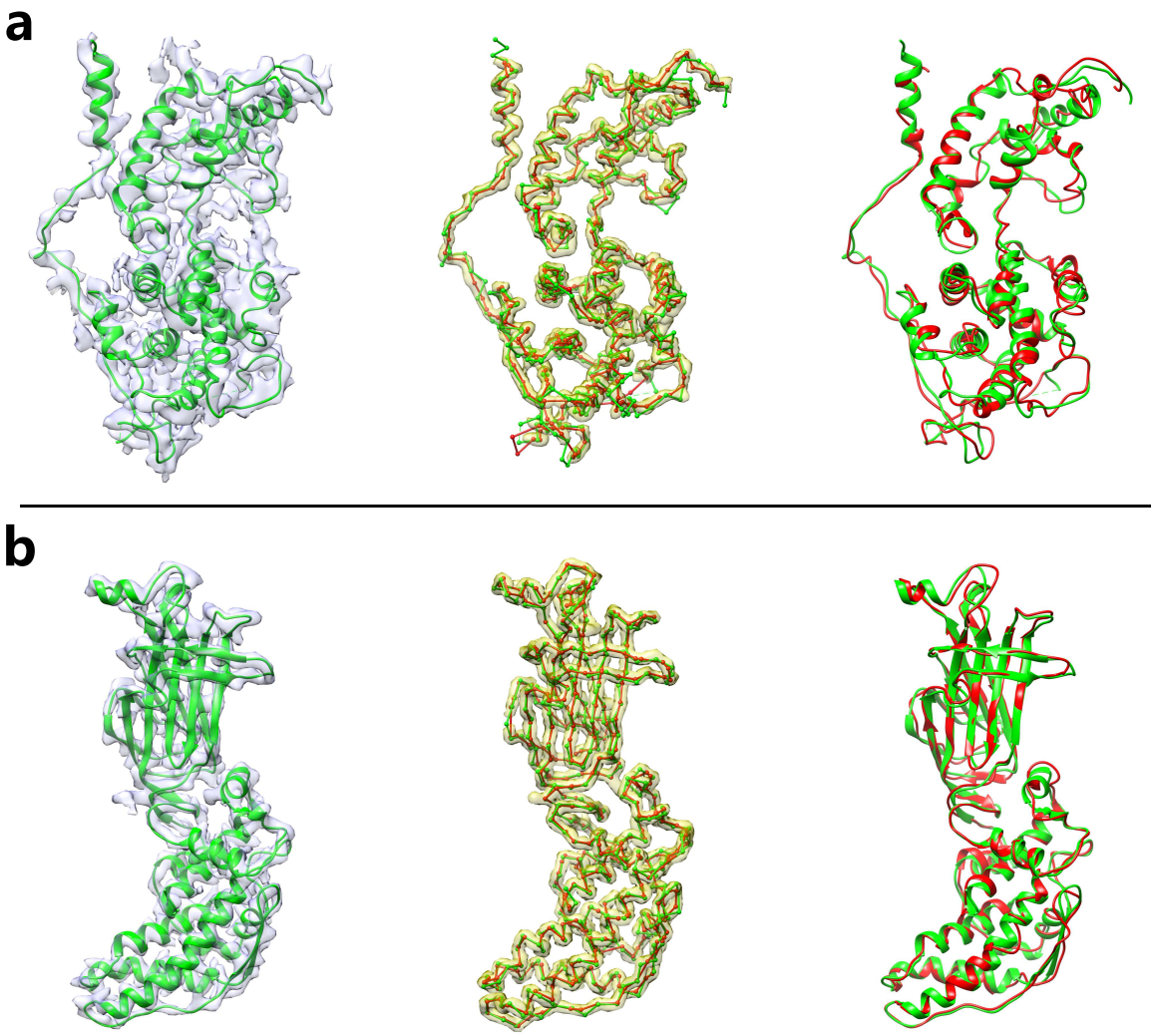

Figure 5: Examples of the models generated by DeepMM for experimental EM maps. The EM density map (transparent grey) and its associated native protein structure (green) are displayed on the left side. The Cα chains of the DeepMM model (red) and the native structure (green) are shown in ball-and-stick format on the predicted main-chain probability map (transparent yellow) in the middle. The full-atom structure generated by DeepMM (red) and the native protein structure (green) are displayed on the right side. **a,** The nucleoprotein at 4.3 Å map resolution (EMD-2867). The top ranked model by DeepMM has a Cα RMSD of 3.1 Å. **b,** The bovine rotavirus VP6 at 2.6Å map resolution (EMD-6272). The top model by DeepMM has a Cα RMSD of 1.7 Å.

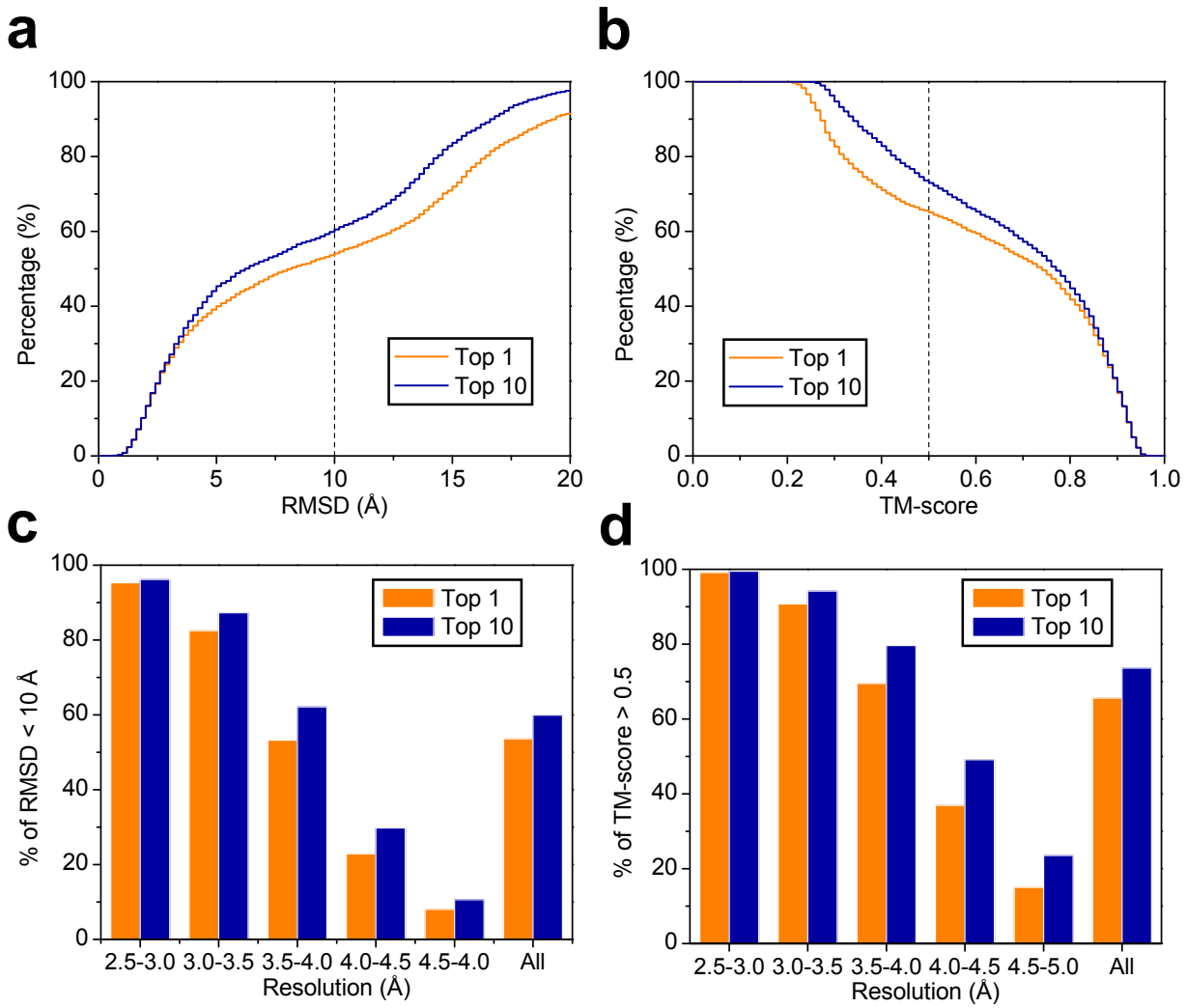

Figure 6: Test results of DeepMM on the 2931 experimental test cases. a, The percentage of the top scored models at different Cα RMSD cutoffs. **b,** The percentage of the top scored models at different TM-score cutoffs. **c,** The percentages of top scored models within 10 Å RMSD in different map resolution ranges. **d,** The percentages of the top scored models with a TM-score above 0.5 in different map resolution ranges.

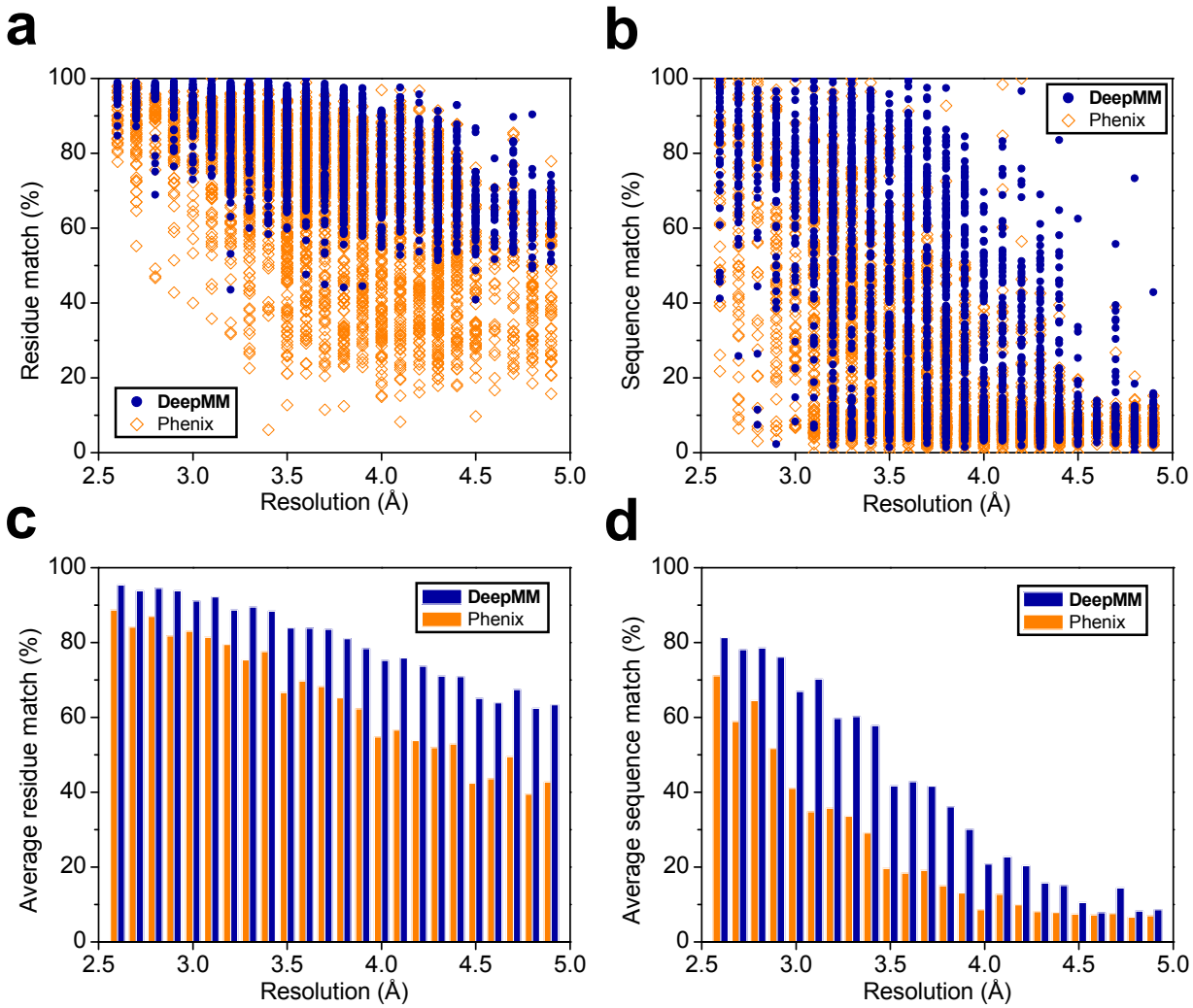

Figure 7: Comparison of the models by DeepMM and Phenix on the large test set of 2931 experimental maps at different resolutions. The results for Phenix are colored in orange, and those for DeepMM are colored in royal blue. **a,** Percentages of the protein residues in the deposited structures reproduced by DeepMM and Phenix. **b,** Percentages of the sequence of the deposited structure reproduced by DeepMM and Phenix. **c,** Average percentage of residue match by DeepMM and Phenix. **d,** Average percentage of sequence match by DeepMM and Phenix.

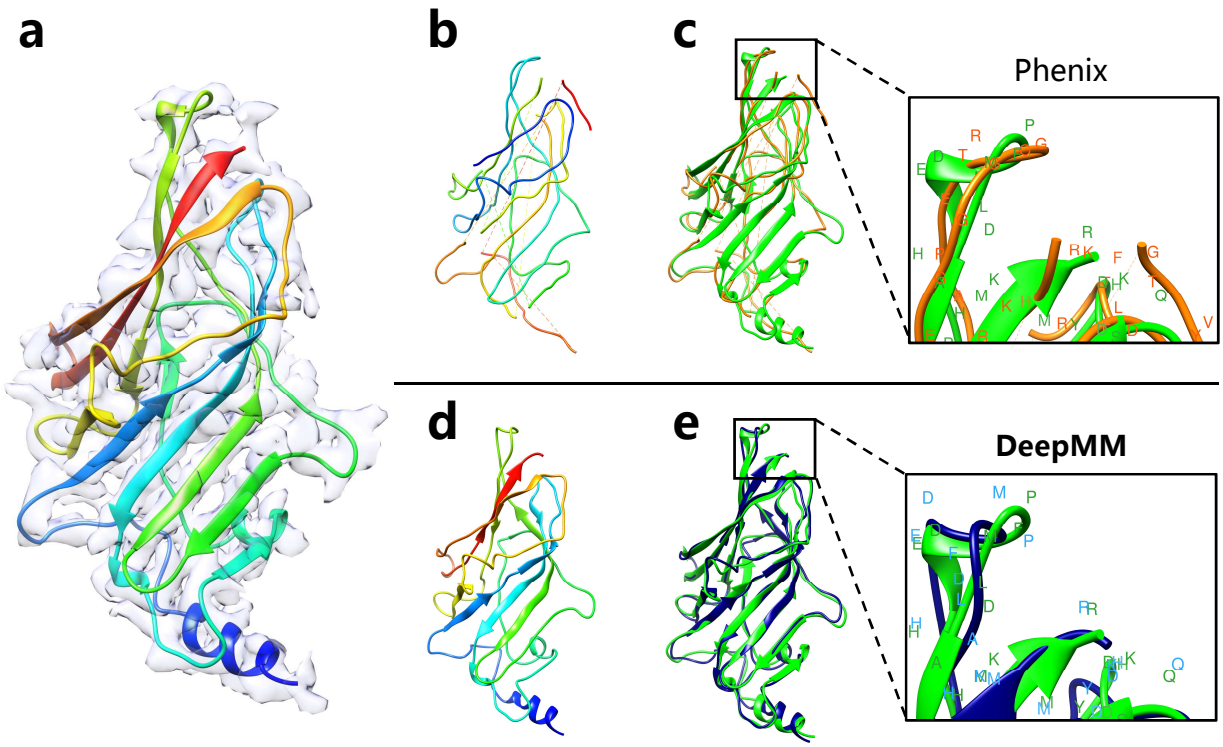

Figure 8: Protein models reconstructed by DeepMM and Phenix for the Chain A of 6DW1 and its associated EM density map at 3.1 Å resolution (EMD-8923). a, The native structure overlapped with its associated EM density map. **b,** The model predicted by Phenix, which has a residue match of 86.7% and a sequence match of 9.9%. **c,** The Phenix model (orange) overlapped with the native structure (green). The enlarged box on the right side shows that the residue names assigned by Phenix model are different from those of the native structure. **d,** The model predicted by Phenix, which has a residue match of 97.1% and a sequence match of 86.8%. **e,** The DeepMM model (royal blue) overlapped with the native structure (green). The enlarged view of the top region of the protein on the right side shows that the sequence assigned by DeepMM is close to that of the native structure.