DEEP LEARNING FOR MODELING PROTEIN ATOMIC STRUCTURES FROM CRYO-EM DENSITY MAPS

A Dissertation

presented to

the Faculty of the Graduate School

at the University of Missouri-Columbia

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

---

by

NABIN GIRI

Dr. Jianlin Cheng, Dissertation Supervisor

MAY 2025

The undersigned, appointed by the Dean of the Graduate School, have examined the dissertation entitled:

DEEP LEARNING FOR MODELING PROTEIN ATOMIC
STRUCTURES FROM CRYO-EM DENSITY MAPS

presented by Nabin Giri,
a candidate for the degree of Doctor of Philosophy, and hereby certify that, in their opinion, it is worthy of acceptance.

Dr. Jianlin Cheng

Dr. Dong Xu

Dr. Yi Shang

Dr. Clarissa Durie

DEDICATION

I dedicate this dissertation to my parents and my two sisters.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor, Dr. Jianlin Cheng, for his guidance and support throughout my graduate studies. I thank my committee members, Dr. Dong Xu, Dr. Yi Shang, and Dr. Clarissa Durie, for their valuable insights and encouragement. I would also like to extend my appreciation to Dr. Liguo Wang for his invaluable support and guidance.

I am grateful to the alumni of the Bioinformatics and Machine Learning (BML) Lab - Dr. Farhan Quadir, Dr. Xiao Chen, Dr. Raj Roy, Dr. Jian Liu, and Dr. Elham Soltanikazemi - for their mentorship and advice. My appreciation goes to my colleagues at BML: Alex Morehead, Ashwin Dhakal, Sajid Mahmud, Frimpong Boadu, Yanli Wang, Rajan Gyawali, Joel Selvaraj, Akshata Hegde, Pawan Neupane, Shreya Basnet, and Tom Nguyen. Special thanks to Rajan for his interest in continuing this research at BML after my graduation.

Finally, I thank Pravash Kumar and Rasmita Koirala for their friendship and kindness.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

DEEP LEARNING FOR MODELING PROTEIN ATOMIC STRUCTURES FROM CRYO-EM DENSITY MAPS

Nabin Giri

Dr. Jianlin Cheng, Dissertation Supervisor

ABSTRACT

Proteins are the dynamic molecular machines that perform essential biological processes important for life. While cryo-electron microscopy (cryo-EM) has revolutionized our ability to visualize large macromolecular complexes at near-atomic resolution, accurately interpreting these density maps to build atomic structures remains challenging. This dissertation presents computational methods that address critical challenges in atomic structure modeling from cryo-EM density maps.

The primary contribution of this work is the development of *Cryo2Struct*, a method for atomic structure modeling using a *de novo* approach that combines 3D transformers with Hidden Markov Models to build atomic structures directly from cryo-EM density maps. This method is further extended in *Cryo2Struct2*, which incorporates evolutionary information from protein language models and utilizes cryo-EM-based predicted atomic structures as templates for AlphaFold3 to refine and correct protein structures.

To support these method development, this dissertation introduces *Cryo2StructData*, a large labeled cryo-EM dataset for artificial intelligence (AI)-based structure modeling, now publicly available to the scientific community. Additionally, to overcome the inherent noise in cryo-EM data, this work develops and releases datasets specifically designed for AI-based density map enhancement, for improving map interpretability which helps in both manual and automatic atomic structure modeling tasks. This work also contributes to the development of *DeepProLigand*, a framework for studying protein-ligand interactions using cryo-EM density maps data.

The computational methods developed in this dissertation demonstrate substantial improvements in both accuracy and completeness of atomic structure modeling from experimental cryo-EM density map. These contributions advance our ability to determine protein structures from cryo-EM density map, ultimately increasing our understanding of protein functions.

All tools and datasets developed through this research are publicly available to promote further scientific advances in this rapidly evolving field.

Chapter 1

INTRODUCTION

Proteins are the fundamental workhorses of our biological systems, orchestrating countless essential processes that sustain life. The complex macromolecular machines, assembled from individual proteins, ligands, small molecules, nucleic acids, and ions, form intricate structures that enable their diverse functions. Understanding the three-dimensional atomic arrangement of these biomolecules provides invaluable insights into their mechanisms and biological functions. However, understanding the atomic structures of large molecular complexes presents significant challenges, particularly because proteins undergo dynamic structural changes as they interact with other molecules to perform their biological functions.

In recent years, we have witnessed remarkable progress in the field of protein structure prediction through machine learning approaches such as AlphaFold [15], ESMFold [16], and RoseTTAFold [17]. While these computational methods have revolutionized our ability to predict *static* protein structures, they struggle to fully capture the *dynamic* conformational changes that occur as proteins transition between functional states. There are several experimental techniques available for studying these molecules in motion, such as Nuclear Magnetic Resonance (NMR), X-ray Crystallography, and Cryogenic Electron Microscopy (cryo-EM). NMR provides insights into molecular motions but is limited to smaller molecules due to signal overlap and sensitivity constraints. X-ray Crystallography provides high-resolution structural information but requires crystallization, which may not reflect the native dynamics of molecules in the solution. With the technological advances in both software and hardware, cryo-EM has emerged as a transformative technology in structural biology, enabling the determination of large macromolecular complexes at near-atomic resolution without the need for crystallization and has become an increasingly powerful tool for structural biologists seeking to understand protein structure and function in near-native states.

This dissertation addresses a significant challenge in cryo-EM-based structure determination

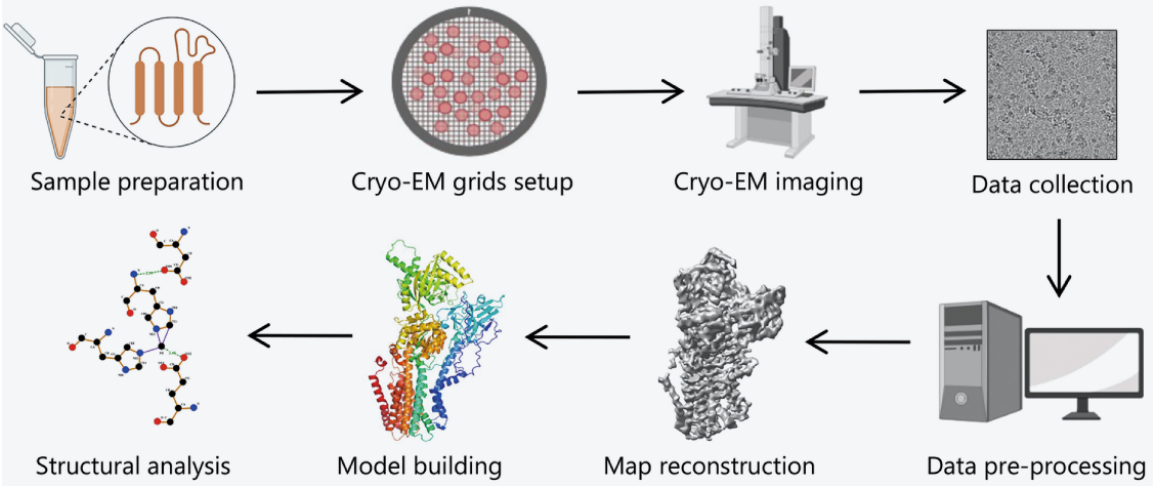


Figure 1.1: Cryo-EM Workflow. Image from [5].

process: atomic structure modeling from cryo-EM density maps. Inspired by recent advances in deep neural networks capable of learning from large-scale, high-dimensional data, this work leverages neural network architectures and builds upon relevant contemporary approaches for 3D protein structure determination from cryo-EM density maps.

1.1 Cryo-EM

Cryo-electron microscopy (cryo-EM), which was recognized with the Nobel Prize in Chemistry in 2017, refers to the imaging of biological specimens at cryogenic temperatures using a transmission electron microscope (TEM). Among various cryo-EM techniques, this dissertation focuses on *single-particle analysis* (SPA), which generates high-resolution 3D density maps of macromolecular complexes by imaging purified samples of target molecules, commonly referred to as '*particles*'. The target molecule of interest often consists of multiple subunits of a biological complex, as such we refer them as '*particles*'.

1.1.1 The Cryo-EM Workflow

The single-particle cryo-EM process as shown in Figure 1.1 begins when a protein sample is spread onto a grid and rapidly frozen into a thin layer of ice at extremely low temperatures (ideally -183°C), a process called vitrification. This freezing preserves molecules in their native state and environment, capturing them at a specific moment in time.

The grid is then placed into a TEM, where an electron beam passes through the sample in

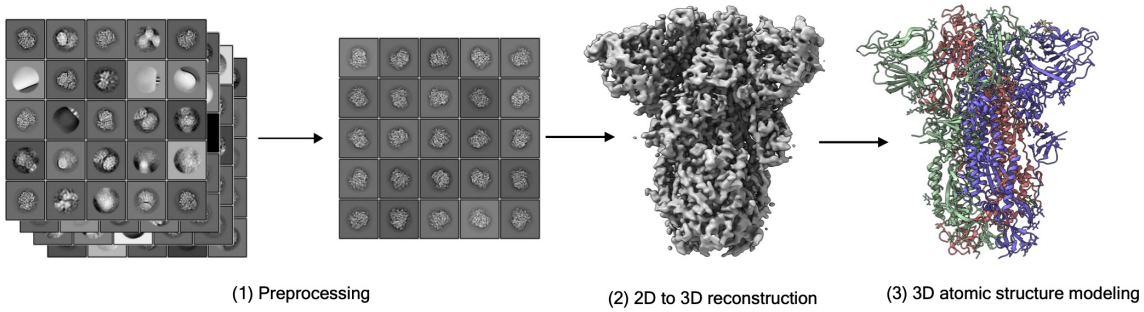


Figure 1.2: Cryo-EM Image Processing Workflow.

vacuum, producing two-dimensional (2D) projections of the molecules onto a detector. Unlike light microscopy, electrons can reveal minute details down to the atomic level. A detector beneath the sample captures these electrons, creating images that are relayed to a computer for processing. This process is repeated many times to capture multiple angles and details of the sample.

Since the molecules are frozen in random orientations, each projection provides a slightly different view. The selected 2D particle images are then computationally combined to reconstruct a 3D model of the molecule referred to as a 3D cryo-EM density map. Each voxel (*pixel in 3D*) in this map represents the electron density value, indicating the strength of the atomic signal in that region. This reconstructed 3D map serves as the foundation for building the atomic structure of the protein.

1.1.2 Computational Pipeline for Structure Determination

The computational workflow of single-particle cryo-EM analysis as shown in Figure 1.2 involves three key stages.

(I) Micrograph Preprocessing: The noisy 2D particle image undergoes processing and segmentation, during which bounding boxes containing individual particles are identified and extracted, a process known as particle picking. The preprocessing stage includes beam-induced motion correction and the estimation of contrast transfer function (CTF) parameters. Various particle-picking algorithms are available [18], ranging from template-based methods to deep learning-based approaches.

Once extracted, particles typically undergo 2D classification, a clustering process that groups images based on similar poses and viewing directions. Particles within the same cluster are averaged to improve signal quality. Finally, a visual inspection is performed to discard clusters that contain false positives (junk) from the particle-picking step.

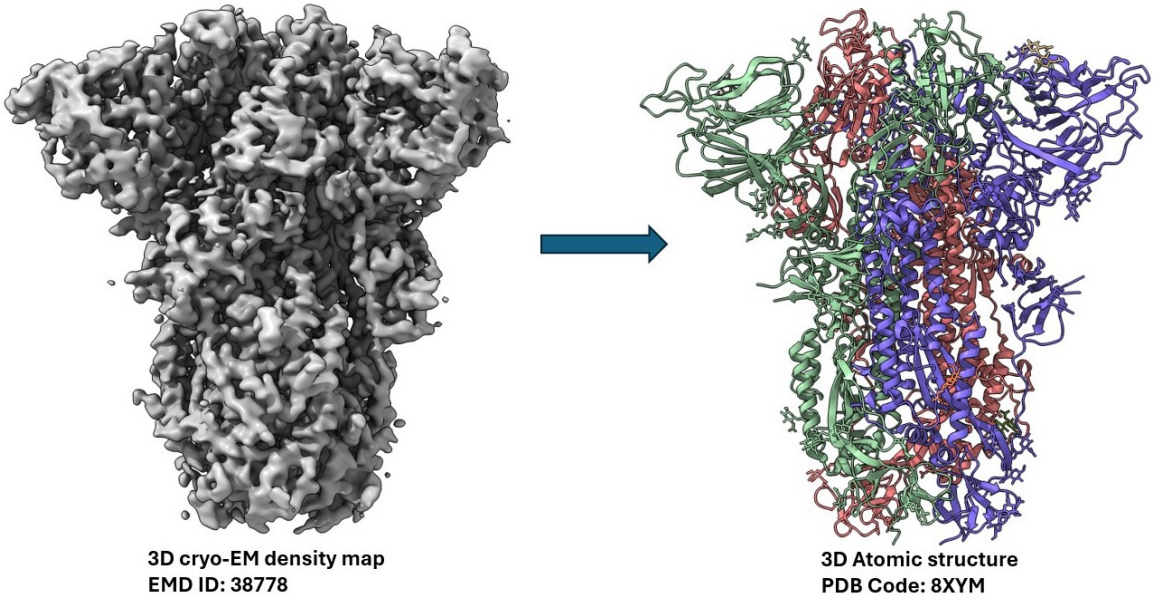


Figure 1.3: 3D Atomic Structure Modeling Task.

(II) 3D Cryo-EM Density Map Reconstruction: The stack of extracted 2D particle images are used for reconstructing 3D cryo-EM density maps. By processing 10^4 to 10^6 of such images, a 3D cryo-EM density map can be obtained [19]. However, these images often suffer from low signal-to-noise ratios, unknown particle orientations relative to the electron beam, and the presence of both *conformational* and *compositional* heterogeneity. *Conformational* heterogeneity refers to different states of the same protein within the dataset, whereas *compositional* heterogeneity refers to different molecules (i.e, small molecules or subunits of a macromolecular complex) within the same dataset. In general, reconstruction can be divided into two parts. In homogeneous reconstruction, the goal is to recover a single 3D volume from a set of 2D projection images. However, when *conformational* heterogeneity is present, the single homogeneous reconstruction model fails to capture this heterogeneity, resulting in an average volume or sometimes a distorted one. The output of a continuous heterogeneity model should capture the full range of conformations of the protein. This step produces the high-resolution cryo-EM density map.

(III) Atomic Structure Building: The high-resolution cryo-EM density map is then used to build the atomic structure of the protein as shown in Figure 1.3, which is the primary focus of this dissertation.

1.1.3 Atomic Structure Building from Cryo-EM Density Maps

The objective of atomic structure building is to predict the 3D atomic structure of a protein from its 3D cryo-EM density map and associated amino acid sequence. Given these inputs, we aim to build a 3D atomic Cα backbone structure of the protein.

1.2 Dissertation Structure and Contributions

The content of each chapter in this dissertation is described as follows:

Chapter 1 : The introduction chapter (the current chapter) provides an overview of structure determination using cryo-electron microscopy (cryo-EM), highlighting the key computational processes involved. Additionally, this chapter briefly discusses the significance and relevance of cryo-EM in structural biology.

Chapter 2 : This chapter explores recent advances and future directions in cryo-EM-based structure modeling. It summarizes existing methods and outlines developments in the field. Throughout this chapter, we refer to the process of deriving atomic structures from cryo-EM density maps as reconstruction.

The contents of this chapter is based on the publication:

Giri, N., Roy, R. S., & Cheng, J. (2023). Deep learning for reconstructing protein structures from cryo-EM density maps: Recent advances and future directions. Current opinion in structural biology, 79, 102536. https://doi.org/10.1016/j.sbi.2023.102536

Chapter 3 : This chapter details the preparation of cryo-EM data for AI-based protein structure modeling. It includes technical validation, step-by-step instructions for dataset preparation, and guidelines on accessing the dataset.

The contents of this chapter is based on the publication:

Giri, N., Wang, L., & Cheng, J. (2024). Cryo2structdata: A large labeled cryo-em density map dataset for ai-based modeling of protein structures. Scientific Data, 11(1), 458. https://doi.org/10.1038/s41597-024-03299-9

Chapter 4 : This chapter describes the method developed for modeling atomic structure from cryo-EM density map.

The contents of this chapter is based on the publication:

Giri, N., & Cheng, J. (2024). De novo atomic protein structure modeling for cryoEM density maps using 3D transformer and HMM. Nature Communications, 15(1), 5511. https://doi.org/10.1038/s41467-024-49647-6

Chapter 5 : This chapter presents a method for modeling atomic structures using multi-task learning, incorporating features extracted from a protein language model and using AlphaFold3 to refine the protein structures.

The contents of this chapter is based on the preprint paper:

Giri, N., & Cheng, J. (2025). Atomic Protein Structure Modeling from Cryo-EM Using Multi-Modal Deep Learning and AlphaFold3. https://www.biorxiv.org/content/10.1101/2025.03.16.643561v1

Chapter 6 : This chapter introduces a method for modeling protein-ligand interactions using cryo-EM density maps. Throughout this chapter, we refer to the process of deriving atomic structures from cryo-EM density maps as reconstruction.

The contents of this chapter is based on the publication:

Giri, N., & Cheng, J. (2023). Improving protein–ligand interaction modeling with cryo-em data, templates, and deep learning in 2021 ligand model challenge. Biomolecules, 13(1), 132. https://doi.org/10.3390/biom13010132

Chapter 7 : This chapter describes the label generation process for AI-based cryo-EM density map enhancement.

The contents of this chapter is based on the preprint paper:

Giri, N., Wang, L., & Cheng, J. (2025). A Labeled Dataset for AI-based Cryo-EM Map Enhancement. https://doi.org/10.1101/2025.03.16.643562

Chapter 8 : This chapter provides detail instructions on the usage of the developed tools of Chapter 3, 4, 5, 6, and 7. It also includes the links to the code repository where the source codes are publicly available.

Chapter 2

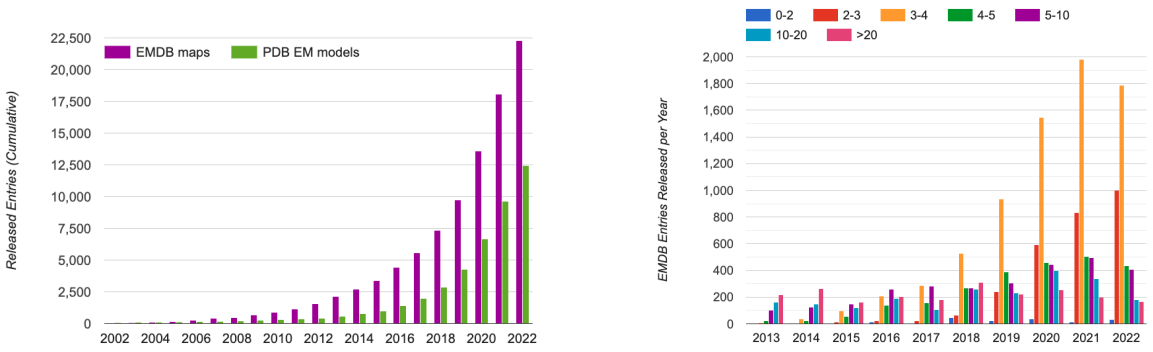
DEEP LEARNING FOR RECONSTRUCTING PROTEIN STRUCTURES FROM CRYO-EM DENSITY MAPS: RECENT ADVANCES AND FUTURE DIRECTIONS

2.1 Abstract

Cryo-Electron Microscopy (cryo-EM) has emerged as a key technology to determine the structure of proteins, particularly large protein complexes and assemblies in recent years. A key challenge in cryo-EM data analysis is to automatically reconstruct accurate protein structures from cryo-EM density maps. In this review, we briefly overview various deep learning methods for building protein structures from cryo-EM density maps, analyze their impact, and discuss the challenges of preparing high-quality data sets for training deep learning models. Looking into the future, more advanced deep learning models of effectively integrating cryo-EM data with other sources of complementary data such as protein sequences and AlphaFold-predicted structures need to be developed to further advance the field.

2.2 Introduction

Cryo-EM is revolutionizing structural biology due to its unique capability of determining the structures of large protein complexes and assemblies. The atomic resolution structure determination for proteins enabled by cryogenic electron microscopy (cryo-EM) [20], allows us to understand the complex biological processes carried out by proteins as well as to identify potential therapeutic protein targets for drug discovery. However, reconstructing de novo protein structures from high resolution ($\sim$ 3 - 4 Å) cryo-EM density maps, which accounts for a large portion of cryo-EM density maps deposited



(a) Growth of cryo-EM density maps and corresponding protein structures

(b) The distribution of the resolution of the cryo-EM density maps ($\AA$)

Figure 2.1: The growth of cryo-EM density maps and cryo-EM-derived protein structures and the distribution of the resolution of the density maps. The statistics was obtained from EMDataResource [6], an unified data resource for 3-Dimension electron microscopy (3DEM) on 2022-09-14.

currently in the EMDB [6], is time-consuming and challenging when homologous template structures for target proteins are not available. For instance, as shown in Figure 2.1, in the year 2022, only about 12,500 out of 22,300 density maps of high resolutions deposited to EMDB have a complete atomic structure available in Protein Data Bank (PDB) [21].

Accurately reconstructing protein structures from cryo-EM maps is a challenging process because the data is often noisy and incomplete and target protein structures can be large and complex. Traditional methods based on energy optimization such as EM-Fold [22], Gorgon [23], Rosetta [24], Pathwalking [25], MAINMAST [26, 27], VESPER [28], and Phenix [29] have made valuable progress in reconstructing protein structures from cryo-EM density maps. These methods rely on extensive physics-based or statistical potential-based optimization algorithms that require high computational resources. These methods often need manual intervention and trials to extract features from the cryo-EM density maps to obtain accurate reconstruction of protein structure.

A different strategy to automatically determine protein structures from cryo-EM density maps is to use the data-driven machine learning approach [30], a kind of artificial intelligence (AI) technology, to directly learn a mapping from cryo-EM density maps to protein structures from the large amount of known cryo-EM data and their corresponding protein structures (i.e., labels). Early AI methods in the field are based on shallow machine learning techniques such as k-nearest neighbor, support-vector machines, or k-means clustering techniques. These methods such as RENNSH [31], SSELearner [32], and Pathwalking [25] are able to identify only secondary structures or simplified backbone structures and often are unable to achieve the optimal solution.

To overcome the challenges of the traditional optimization methods and early machine learning

methods, deep learning methods [33] have been developed to automatically reconstruct three-dimensional (3D) protein structures from cryo-EM density maps with significant success in recent years (see Figure 2.2 for a summary of a general cryo-EM protein structure determination pipeline powered by deep learning). In this article, we review the recent development of deep learning technology in the field, analyze their impacts, investigate the challenging issues in preparing data to train deep learning models, and discuss some new trends to further advance the field.

2.3 Deep learning for reconstruction of protein structures from cryo-EM density maps

Deep learning, also called deep neural network, is currently the most powerful machine learning method of predicting the properties of an object from the input data describing the object. It has achieved great success in many fields including a recent major breakthrough in predicting protein structure from sequence by AlphaFold [34]. Compared to other machine learning methods, deep learning has a unique capability of extracting informative features for pattern recognition from raw data automatically, making it suitable for reconstructing protein structures from raw density maps in which only a large amount of numbers rather than informative features are available.

It is worth noting that deep learning has been applied to almost all the areas of cryo-EM data analysis [35, 36, 37, 38, 39, 40, 41] from sample preparation, particle picking, density map denoising, and to the final step of 3-D structure determination. Due to the space limit this review is focused on the last step of cryo-EM data analysis - reconstructing protein structures from density maps. The deep learning architectures designed for this task and how to prepare data to train them are discussed in the two subsections below.

2.3.1 Deep learning architectures for reconstructing protein structures from cryo-EM density maps

Deep learning methods for inferring protein structures from cryo-EM density maps can be classified into different categories based on the neural network architectures, for example, convolutional neural network (CNN) [42], U-Net [43, 44], graph convolutional network (GCN) [45], and long- and short-term memory network (LSTM) [46] they use and the output (e.g., 3D structure and secondary structure) they generate from density map input. Early deep learning methods aimed to identity secondary structures from low- and medium-resolution density maps [47]. As more and more

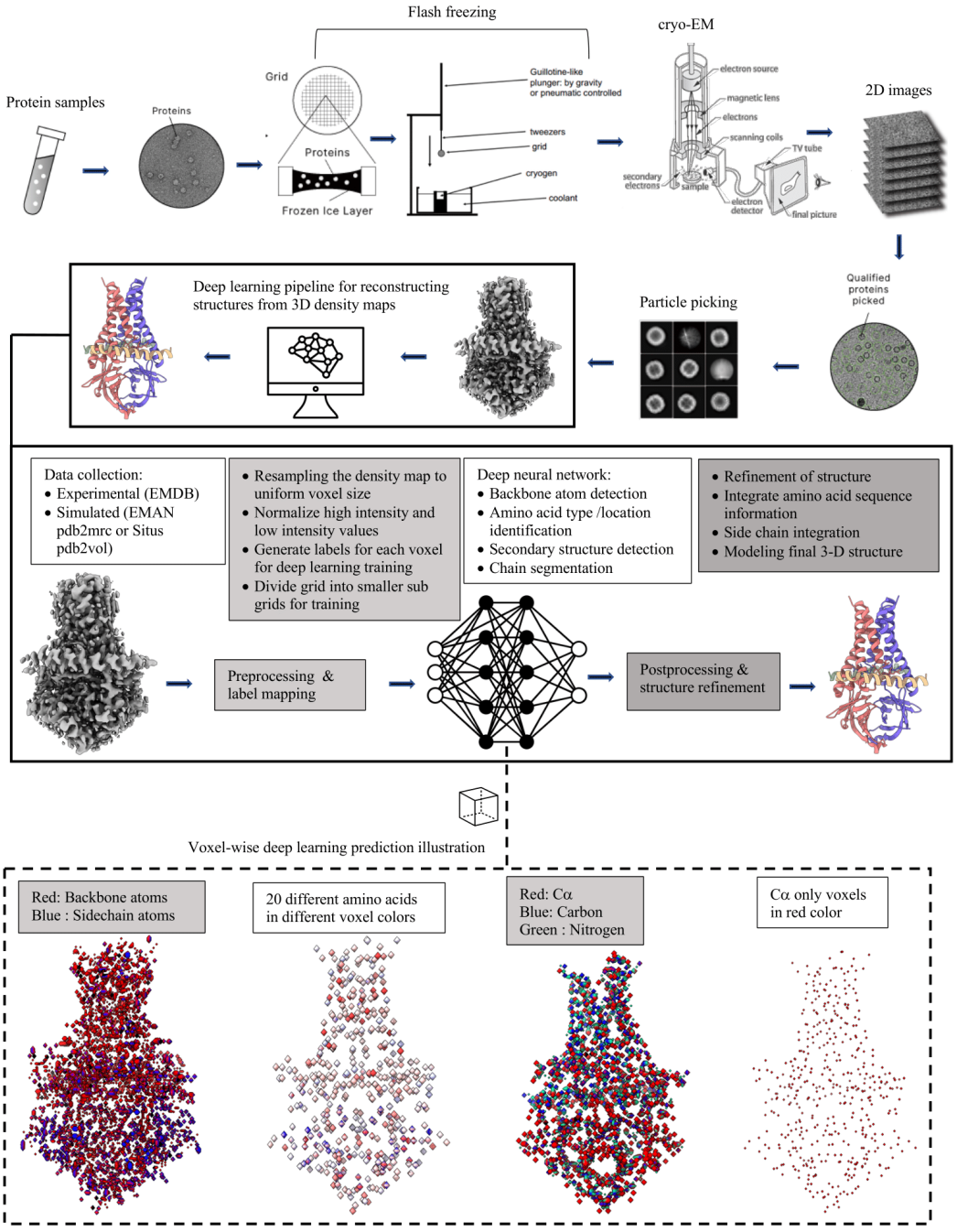
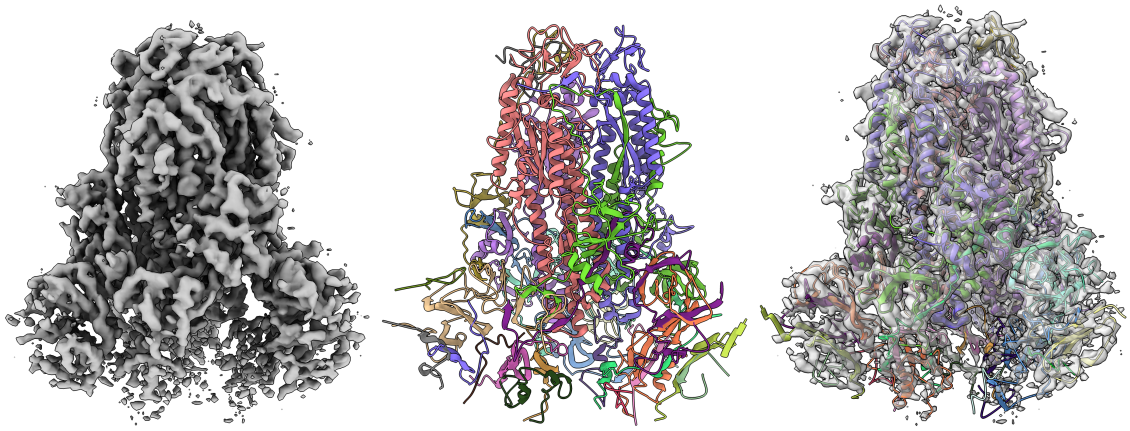


Figure 2.2: A summary of a cryo-EM density map generation and protein structure reconstruction pipeline powered by deep learning. The density map (EMD-22898) illustrated in the figure is for SARS-CoV-2 ORF3a [7]. PDB ID: 7KJR.



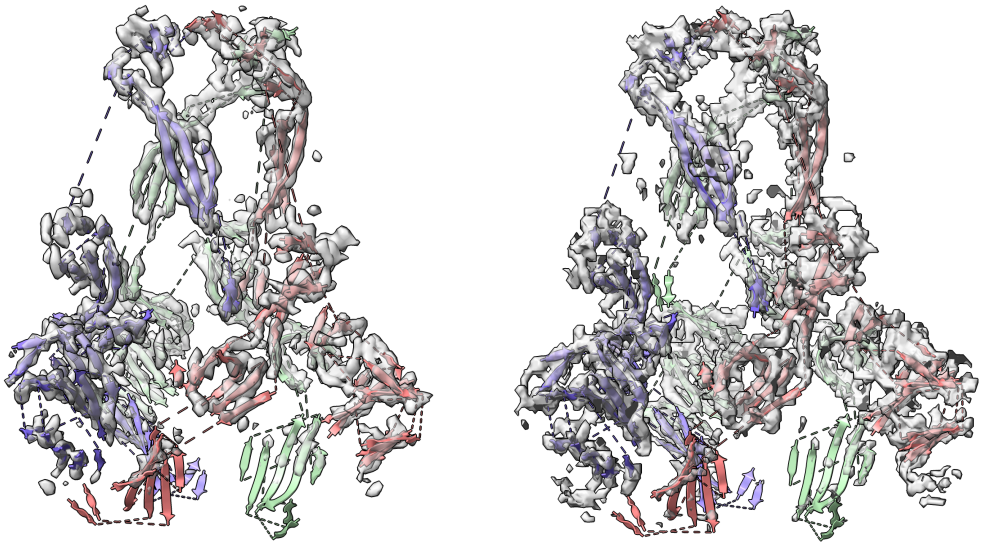
(a) (b) (c)

Figure 2.3: An example of reconstructing a structure from the cryo-EM density map of SARS-CoV spike gycoprotein by deep learning. **(a)** Density map of SARS-CoV spike glycoprotein [8] (EMD-6732) in resolution of 3.8 $\mathring{A}$ at recommended contour level of 0.06 (11.0 σ). **(b)** The structure reconstructed from EMD-6732 by a deep learning method - DeepTracer. The RMSD is 1.023 $\mathring{A}$ with respect to the ground truth structure (PDB ID: 5XLR). **(c)** The overlay of the density map and reconstructed structure at 0.5 transparency level by UCSF ChimeraX [9].

high- resolution density maps became available, recent deep learning methods targeted at directly reconstruct 3D backbone structures (i.e., locations of carbon and nitrogen atoms on the protein backbone) and even full atom 3D structures (i.e., locations of all/most heavy atoms and amino acid identity/type) from density maps [47, 48, 49, 50, 51]. An example of deep learning reconstruction of protein structure from cryo-EM density map is showed in Figure 2.3.

One of the most widely used deep learning architectures of obtaining protein structural information from density maps is convolution neural network (CNN). CNNs use a mathematical operation known as convolution to extract features from spatially organized data such as a 2D-image or 3D density map to predict the properties of the data (e.g., classifying voxels in a density map into amino acid types). Several CNN methods (mostly 3D-CNN architecture) including Generator [48], Emap2sec [52], AAnchor [53], CNN Based [54], Cascaded-CNN [47], and CR-I-TASSER (mostly 3D CNN) [50] have been developed to determine secondary structures [52, 54], backbone-/full-atom 3D structures [50, 48, 53] or both from cryo-EM density maps [47]. Cascaded- CNN is the first deep learning de novo method of directly reconstructing 3D structures of proteins from cryo-EM density maps, even though it focuses on building backbone structures. CR-I-TASSER combines the 3D-CNN prediction from cryo-EM maps and an advanced protein structure prediction method - I-TASSER [55] to build full-atom protein structures.

Another widely used convolutional neural network architecture in the field is U-Net [43], originally designed for biomedical image classification and segmentation tasks. U-Net consists of a series of

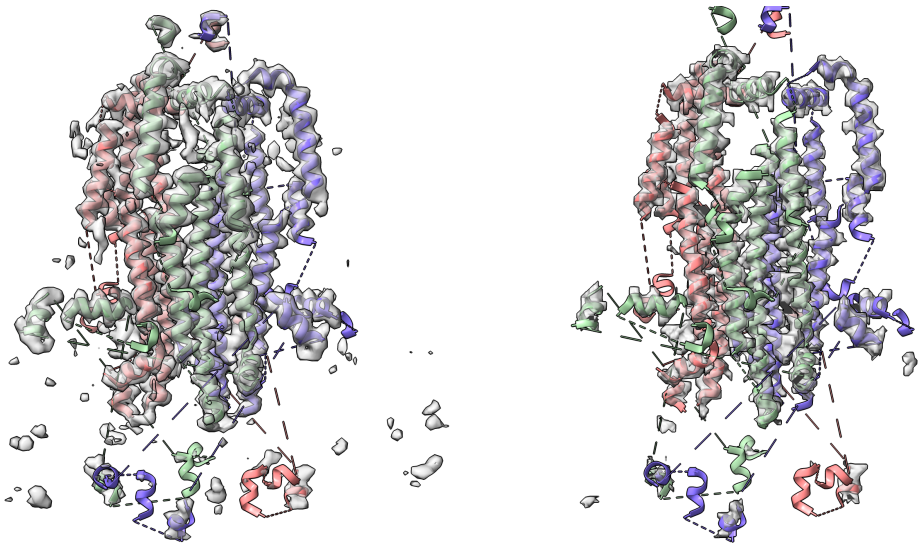


(a) (b)

Figure 2.4: An example of secondary structure annotation in cryo-EM density map of SARS-CoV spike gycoprotein [8] (EMD-6732) by deep learning. PDB ID: 5XLR. **(a)** Haruspex [10] predicted strands in transparent gray overlapped with deposited PDB structure strands. **(b)** EMNUSS [11] predicted strands in transparent gray overlapped with deposited PDB structure strands.

convolution-based down-sampling layers to condense the input images into smaller dimensions and a series of convolution-based up-sampling counterpart layers to reconstruct the data of the same dimension as in the down-sampling process to classify/segment pixels in the input images. Compared to the standard CNN architectures, U-Nets can be more effective in extracting multi-level abstract representations of the data through the down-sampling and up-sampling processes. The 2D U-Net architecture has been generalized to 3D U-Net architectures in Haruspex [10] and EMNUSS [11] to detect secondary structures from cryo-EM density maps (e.g., Figures 2.5 and 2.4), and in DeepTracer [56] and EMBuild [57] to reconstruct 3D protein structures from cryo-EM density maps. DeepTracer has been successfully applied to reconstruct the structures of some SARS-CoV proteins from cryo-EM density maps (e.g., Figure 2.3).

In addition to CNN and U-Net, other deep learning architectures such as graph convolutional networks (GCN) and long- and short-term memory network (LSTM) have also been used with CNN to reconstruct protein structures from cryo-EM density maps [48]. A summary of different deep learning-based methods, their function (e.g., input and output) and availability is presented in Table 2.3.1.



(a) (b)

Figure 2.5: An example of secondary structure annotation in cryo-EM density map of SARS-CoV spike gycoprotein [8] (EMD-6732) by deep learning. PDB ID: 5XLR. **(a)** Haruspex [10] predicted helices in transparent gray overlapped with deposited PDB structure helices. **(b)** EMNUSS [11] predicted helices in transparent gray overlapped with deposited PDB structure helices.

Methods	Architecture	Function	Open source
Structure Generator[48]	3-D CNN, GCN, Bidirectional LSTM	First use 3-D CNN to identify amino acids and their rotameric identities in an EM map and then GCN and LSTM to ***build protein structures***	✓
Emap2sec[52]	3-D CNN	Take voxel cubes as input to ***identify secondary structures of protein***	✓
AAnchor[53]	3-D CNN	Take in voxel cubes to ***identify amino acid types and locations***	✓
A CNN Based Method[54]	3-D CNN	Take in voxel cubes to ***detect secondary structures of protein from background***	×
CascadedCNN [47]	Cascaded 3-D CNN	Take in voxel cubes to ***identify Cα atoms of protein backbone and secondary structures to generate 3D protein structures***	✓

Haruspex[10]	3-D U-Net	Take in voxel cubes to predict the probabilities of 4 different classes; α-helix, β-sheet, nucleotide, or unassigned to ***assign secondary structures***	✓
DeepTracer[56]	3-D U-Net	Take in voxel cubes to ***identify the location of backbone atoms, secondary structures and amino acid types simultaneously to build 3D structure***	×
DeepTracer ID [49]	DeepTracer (3-D U-Net) and pre-calculated AlphaFold2 protein library	Use DeepTracer to generate an initial 3D protein structure to search AlphaFold2DB to ***identify similar structural hits for refinement***	×
CR-I-TASSER [50]	3-D CNN, I-TASSER	Predict Cα using 3-D CNN for selecting structural templates for I-TASSER to ***generate 3D protein structure***	✓
EMBuild [57]	3-D U-Net++, AlphaFold	Integrate AlphaFold structure prediction, FFT-based global fitting, domain-based semi-flexible refinement, and graph-based iterative assembling with main-chain probability maps predicted by U-Net++ to ***build 3D protein structure***	✓
EMNUSS [11]	3-D U-Net++	Take in voxel cubes to ***identify secondary structures of protein***	✓
ModelAngelo [58]	Graph Neural Network	***Refines geometry of protein chains and classifies amino acid for each nodes***	×

Table 2.1: Summary of deep learning based methods for protein structure reconstruction from cryo-EM density maps.

Inspired by the recent breakthrough in developing deep learning methods of predicting protein structures from sequences such as AlphaFold [34] and RoseTTAFold [17], a new trend is to integrate

deep learning methods of reconstruct protein structures from cryo-EM density maps with the advanced computational (e.g., deep learning) methods of predicting protein structures from sequences to obtain more accurate structural models. For instance, DeepTracer ID [49] first uses DeepTracer [56] to build an initial structure from cryo-EM density maps and then search the structure against a database of AlphaFold-predicted structures to identify similar structural hits to enhance the reconstructed structure. EMBuild [57] combines the structures reconstructed from cryo-EM maps, AlphaFold-predicted structural models and other protein structural refinement methods to construct accurate structures for protein complexes. ModelAngelo [58] refines the geometry of protein chains by combining information extracted from cryo-EM data, prior knowledge of protein geometries, and amino acid sequence data. DeepProLigand [59] integrates the protein structural models reconstructed from cryo-EM density maps by DeepTracer with the known template structures containing ligands to model protein-ligand interaction, which was ranked first in the ligand prediction in 2021 EMDataResource Ligand Model Challenge.

2.4 Data preparation for training deep learning methods to reconstructing protein structures from cryo-EM density maps

2.4.1 Cryo-EM density map data collection

Collecting a sufficient amount of high-quality data to train and test deep learning models is critical for any deep learning task. The common way to acquire the experimental cryo-EM density maps is through the Electron Microscopy Data Bank [6]. An alternative approach employed by some methods such as Cascaded-CNN and SSELearner is to simulate the density map from the PDB protein structure. Cascaded-CNN applies *pdb2mrc* from EMAN2 package [60], and VESPER [26] uses *pdb2vol* from Situs package [61] to generate the simulated maps. However, simulated maps lack complex noise, missing density values, and experimental artifacts which can arise from particle alignment errors, interaction of electron beam with the atoms, or movement of atoms during image capture. Therefore, the deep learning models trained on simulated maps may not work as expected on very noisy experimental data. To address the problem, CR-I-TASSER [50], EMNUSS [11] and Emap2sec [52] employs a hybrid training approach that uses both simulated maps and experimental maps in the training and validation process.

2.4.2 Training data preprocessing

Prior to using the cryo-EM density map to train deep learning models, it is generally necessary to normalize and standardize the data to make them suitable for deep learning as shown by Cascaded-CNN and DeepTracer, which perform data grid resampling, density value normalization, and grid division. These preprocessing steps ensure the uniformity among density maps and help deep learning models to extract features and recognize patterns more easily. During the grid division, the 3D cryo-EM is split into the cubes of a specific size (e.g., $64 \times 64 \times 64$ Å^3 by Cascaded-CNN and DeepTracer, $50 \times 50 \times 50$ Å^3 by CR-I-TASSER, $40 \times 40 \times 40$ Å^3 by Haruspex, and $11 \times 11 \times 11$ Å^3 by Emap2sec and AAnchor). Each of these cubes is then processed by the deep learning method to classify the voxels into the targeted classes such as amino acid types (identities) and secondary structures.

2.5 Future directions

Deep learning has made a significant impact on protein structure reconstruction from cryo-EM density maps. However, the field is still in the early stage of development. The latest deep learning technology such as graph neural networks [62] and attention mechanisms [63] have not been extensively used in the field. While CNNs and U-Nets based on convolution are currently the most used methods for structure reconstruction, they have some short-coming for 3D structural modeling. CNNs are translation-equivariant, but not fully rotation invariant that is desirable for 3D structure analysis. Moreover, the convolution mechanism propagates message in the constrained local receptive field, which is not as effective as the attention mechanism [63] that can leverage all the input information by automatically weighting the input features according to their relevance as demonstrated by the remarkable success of AlphaFold2 in protein structure prediction. More sophisticated deep learning models like attention-based Transformer models [63], 3D equivariant graph neural networks [64], and AlphaFold2-like deep learning models need to be developed to better use cryo-EM data to improve reconstruction accuracy.

Another important direction is to use deep learning to integrate cryo-EM data with multiple other sources of complementary data such as protein structural models predicted from sequences, structural templates in the Protein Data Bank (PDB), and protein sequences to more accurately reconstruct protein structures from noisy density maps that often miss the density values of some atoms. The current integration process is limited to shallow data combination. For instance, DeepTracerID uses

AlphaFold models to refine the structural models predicted from structural models reconstructed from deep learning. More comprehensive, end-to-end deep learning models to combine multiple sources of data to generate accurate final protein structures can be developed to automatically and accurately reconstruct protein structures from the data.

Moreover, it is important to integrate cryo-EM based deep learning methods of reconstructing protein structures with the advanced methods developed in the field of protein structure prediction. The structural models directly reconstructed from cryo-EM data by deep learning generally have correct overall topology, but the reconstructed models may not satisfy physico-chemical restraints such as bond length and bond angles and not have all the molecular details (e.g., the precise location of all side chain atoms) [47, 59]. Linking the atoms of amino acids identified from the density maps into full peptide chains consistent with protein sequences and physical-chemical restraints is still challenging. However, the modeling techniques such as protein structure refinement and molecular dynamics to fix these problems have been established for protein structure prediction [34]. Some methods such as CR-I-TASSER [50] have started to integrate the two kinds of technologies. More synergistic integration of the two are needed to generate high quality realistic protein structures from cryo-EM data.

The development of high-quality deep learning models to reconstruct protein structures from cryo-EM density maps critically depends on the availability of sufficient high-quality training data. Although experimental cryo-EM data and its related ground truth structure are freely accessible through EMDB [6] and RCSB PDB [21], these datasets still need to be pre-processed and labeled before they can be used for deep learning training. Curating a large amount of high-quality training and test data is challenging and time consuming, but often receives little attention. Currently, there are few well-curated experimental cryo-EM data sets publicly available for training and evaluating deep learning models in the field. Therefore, more effort needs to be devoted to creating such data sets and make them to publicly available for the community to use.

2.6 Conclusion

A number of useful deep learning models have been developed to reconstruct protein structures from cryo-EM density maps, demonstrating deep learning is a promising technology to further push the frontier of applying cryo-EM technology to determine protein structures. As the deep learning field is evolving very fast, many more state-of-the-art deep learning architectures (e.g., AlphaFold2-like models and transformers) have yet to be applied to further advance the emerging field. More

sophisticated deep learning methods need to be developed to seemlessly integrate cryo-EM data with other complementary data such as predicted protein structures, protein sequences, and template structures to further improve cryo-EM-based structure determination. A synergistic integration of cryo-EM based protein structure determination techniques and latest protein structure prediction techniques is also important for generating highly accurate native-like protein structures To speed up the development, more effort is need to create a large amount of high-quality cryo-EM training and test data for the community to use. In the next chapter of this dissertation, the dataset preparation strategy is explained.

Chapter 3

A LARGE LABELED CRYO-EM DENSITY MAP DATASET FOR AI-BASED MODELING OF PROTEIN STRUCTURES

3.1 Abstract

The advent of single-particle cryo-electron microscopy (cryo-EM) has brought forth a new era of structural biology, enabling the routine determination of large biological molecules and their complexes at atomic resolution. The high-resolution structures of biological macromolecules and their complexes significantly expedite biomedical research and drug discovery. However, automatically and accurately building atomic models from high-resolution cryo-EM density maps is still time-consuming and challenging when template-based models are unavailable. Artificial intelligence (AI) methods such as deep learning trained on limited amount of labeled cryo-EM density maps generate inaccurate atomic models. To address this issue, we created a dataset called Cryo2StructData consisting of 7,600 preprocessed cryo-EM density maps whose voxels are labelled according to their corresponding known atomic structures for training and testing AI methods to build atomic models from cryo-EM density maps. Cryo2StructData is larger than existing, publicly available datasets for training AI methods to build atomic protein structures from cryo-EM density maps. We trained and tested deep learning models on Cryo2StructData to validate its quality showing that it is ready for being used to train and test AI methods for building atomic models.

3.2 Background & Summary

Accurately determining three-dimensional (3D) structure of proteins is critical for unlocking key insights into their molecular functions [65] and interactions with other proteins as well as small molecules like ions, ligands, and therapeutic drugs [66]. In recent years, cryo-EM [67] has emerged

Table 3.1: Summary of the currently used number of experimental cryo-EM density maps in the datasets for various AI-based methods and their public availability. The dataset size reported for Cryo2StructData[1] corresponds to the version released as of March 2023. DT refers to DeepTracer. MA refers to ModelAngelo.

	EMNUSS[11]	Haruspex[10]	DT[56]	MA[58]	Cryo2StructData
Approach	3D U-Net++	3D U-Net	3D U-Net	CNN+GNN	Cryo2Struct[72]
Dataset Size	163	415	1,800	3,892	7,600
Train Data Size	120	293	1440	3715	7392
Test Data Size	43	122	360	177	208
Data released	×	×	×	×	✓

as the most important technology for experimentally determining the structures of large protein complexes that are difficult or impossible for other experimental techniques such as X-ray crystallography or Nuclear Magnetic Resonance (NMR) to solve. The field of cryo-EM is advancing at a rapid pace with improvements in image data collection and processing techniques [20]. This progress has resulted in large amount of high-quality cryo-EM images of proteins and their complexes, and high-resolution 3D density maps reconstructed from two-dimensional (2D) images. EMPIAR, the Electron Microscopy Public Image Archive [68], is an archive for raw images, such as micrographs and particle stacks [69], that is used in the construction of cryo-EM density maps. Subsequently, these reconstructed 3D cryo-EM density maps are deposited into EMDB, the Electron Microscopy Data Bank [6].

However, the building of atomic 3D models of large protein complexes from cryo-EM density maps is still a very challenging problem that often requires extensive manual intervention. The task is particularly difficult if the structures of the individual chains (units) in large protein complexes are not available or cannot be predicted from sequences by cutting-edge protein structure prediction methods such as AlphaFold2 [34] which serve as templates to fit into cryo-EM density maps for building the models of the complexes.

To address this problem, significant efforts have been put into developing machine learning, particularly deep learning (DL) methods [70], to directly build atomic models from cryo-EM density maps as the density in each voxel (i.e., 3D pixel) of the high-resolution density map contains sufficient information about protein structures in most cases. Therefore, sophisticated deep learning models trained with density maps as input and their corresponding structures as labels can potentially infer protein models that fit the cryo-EM density maps correctly and accurately [59, 71].

However, designing and implementing deep learning-based methods for building atomic models from cryo-EM density maps requires a large labeled cryo-EM dataset. Different methods have been

trained previously to predict different aspects of protein structures from cryo-EM density maps, nevertheless, up to now, as shown in Table 3.1, there is still no work on generating a publicly available, large, labeled dataset to push the frontier of the model building from cryo-EM density maps. Recently, the exponential growth of cryo-EM has led to a surge in the deposition of high-resolution cryo-EM density maps in the EMDB and their corresponding protein models in the Protein Data Bank (PDB) [21]. Leveraging these precious resources, we created Cryo2StructData [1], a large labeled cryo-EM density map dataset for developing and testing AI-based methods to build atomic models from cryo-EM density maps. We also trained and tested deep learning models on the dataset to rigorously validate its quality. Cryo2StructData is the first, large, publicly available cryo-EM dataset with standardized input features and well-curated output labels that is fully ready for the development of AI-based atomic structure modeling tools in the field.

3.3 Methods

3.3.1 Related Works

Protein structure modeling from simulated density maps: Early methods to build atomic models from cryo-EM density maps utilized protein structures in the Protein Data Bank [21] to generate theoretical density maps at different resolution, usually referred to as simulated density maps for training and testing. For instance, Cascaded-CNN [47] utilized *pdb2mrc* from the EMAN2 package [60], and VESPER [26] utilized *pdb2vol* from the Situs package [61] to generate simulated density maps. However, simulated density maps lack the complexity of real-world density maps such as high noise, missing density values, and experimental artifacts that arise from protein particle picking errors, or atom movement during image capturing in the cryo-EM data collection [73]. As such, deep learning models trained on simulated density maps may not perform well on noisy experimental density maps. Therefore, real-world cryo-EM density maps with labels need to be created to further advance the field.

Protein structure modeling from experimental density maps: As more and more experimentally determined density maps became readily available in the EMDB, different methods have utilized these maps to predict atoms in cryo-EM density maps, as shown in Table 3.1. A recent method, DeepTracer [56], offers users the ability to build models through a web interface. It utilized 1,800 experimental maps to predict the positions and amino acid types of Cα atoms of proteins, which were then aligned with protein sequences to model protein structures. ModelAngelo [58] used

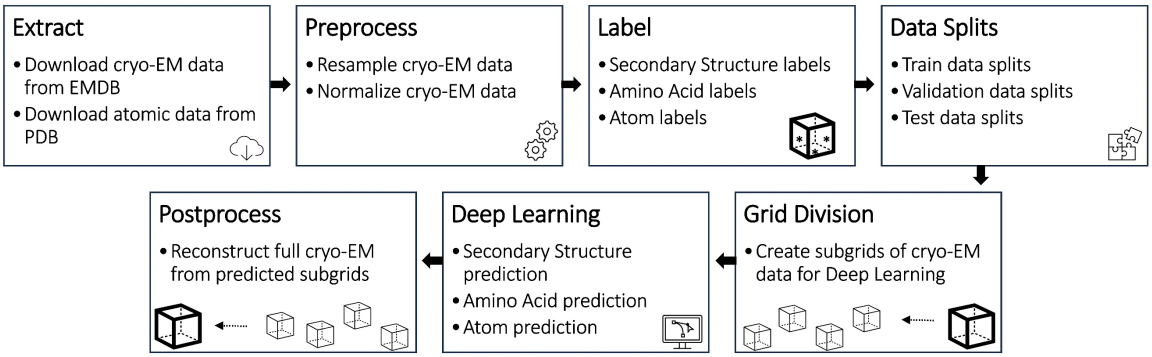


Figure 3.1: The data preparation and evaluation pipeline for Cryo2StructData.

3,715 experimental maps for training and a test set of 177 maps for testing. Both DeepTracer and ModelAngelo automatically generate the atomic models of the protein, but they have not released their processed data. Haruspex [10] used 293 experimental maps for training and an independent test set of 122 experimental maps for testing, and EMNUSS [11] used 120 and 43 experimental density maps for training and testing respectively. CR-I-TASSER [50], EMNUSS [11], and Emap2sec [52] employed a hybrid approach that combined both simulated maps and experimental maps in their training and validation processes. Haruspex, EMNUSS, and Emap2sec are designed for secondary structure prediction, but they have not made their processed data available.

Furthermore, despite the significant progress, the datasets used in these works only account for a small portion of high-resolution density maps available in the EMDB and they are not publicly available for the AI and machine learning community to use. Significantly, Giri et al. [70] emphasize the critical importance of developing high-quality cryo-EM datasets, specifically for training and evaluating deep learning methods. In this context, we aim to prepare, preprocess, label, validate, and consolidate complementary information that can be utilized for accurate atomic model building through AI-based methods from experimental cryo-EM density maps.

3.3.2 Cryo2StructData preparation

The Cryo2StructData was prepared by a data processing pipeline shown in Figure 3.1. The data generated from each stage of the pipeline was verified to ensure the details of original experimental density maps were preserved (see the Technical Validation section for details). The source code of the data preparation pipeline is released at the GitHub repository of Cryo2StructData for users to reproduce the process. The details of each data preparation stage are described in the following subsections.

Data extraction: Cryo2StructData was curated from the experimental cryo-EM density maps released till 27 March 2023. We downloaded relatively high-resolution cryo-EM density maps for proteins and their complexes with resolutions between 1 and 4 Angstrom (Å). In total 9,500 cryo-EM density maps were collected from the EMDB [6]. Similarly, we downloaded the atomic models (i.e., PDB files) corresponding to the density maps from the PDB [21]. The downloaded PDB files were used to label voxels of density maps. The associated PDB ID for each cryo-EM density maps was extracted from the metadata downloaded from Electron Microscopy Data Bank (EMDB).

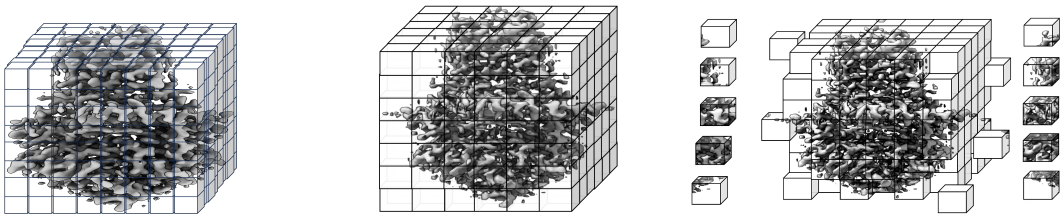
We further filtered out the density maps that do not have structures in the PDB. After the filtering, 7,600 cryo-EM density maps were left for further processing. A metadata document containing the meta information such as the EMD ID and corresponding PDB entry for each density map, resolution of the density map, structure determination method, software used to determine the density map, the title and the journal of the article describing the density map is provided in the Cryo2StructData Dataverse [74].

Preprocessing: The experimental cryo-EM density maps are stored in MRC [75] format. The MRC format contains the data in a 3-dimensional (3D) array, often referred to as 3D grid. Each voxel also known as 3D pixel contains a value indicating how strong an atom's signal is present in the voxel. We refer to these density maps as original raw density maps, which need to be further preprocessed so they can be used to train AI methods. Building upon established practices and techniques [47, 56], the preprocessing is implemented in two steps: **(a)** cryo-EM density map resampling and **(b)** cryo-EM density map normalization, as follows:

a) **Resample cryo-EM density maps:** Different raw cryo-EM density maps usually have different voxel sizes, which need to be standardized. We resampled the density maps to a uniform voxel size of 1 Angstrom (Å), using *vol resample* command within UCSF ChimeraX [9] in the non-interactive mode. The idea of resampling density map is illustrated in Figure 3.2a and 3.2b. All the resampled density maps have uniform, the same voxel size and are used for the following normalization step.

b) **Normalize cryo-EM density maps:** We applied scaling and clipping to normalize the density values of the resampled cryo-EM density maps. In the cryo-EM density maps, positive density values represent regions (voxels) where the protein is likely present. The range of these values can differ from maps to maps with some maps containing values in one range (e.g., [-2.32, 3.91]) and others in another range (e.g., [-0.553, 0.762]). To make the density values of different

density maps comparable, we perform the percentile normalization by first calculating 95th percentile of the positive density values in a density map and then dividing the density values in the map by this value. To deal with the extreme outliers in the density values, we set all values below 0 to 0 and all values above 1 to 1 after the division. The normalization removes the cross-map difference caused by differences in experimental conditions, and software used to process the maps, allowing AI methods to learn patterns across different cryo-EM density maps. Finally, the resampled and normalized map is saved in the MRC format [75] with the filename as `emd_normalized_map.mrc`. This file is available inside the directory named after the EMDB entry ID and is used as an input for AI (e.g., deep learning) models and for the label generation process.

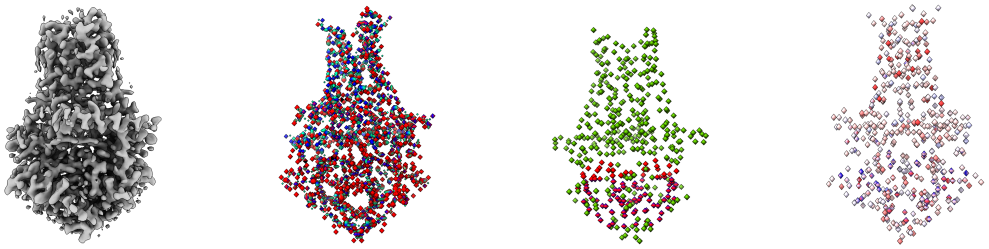


(a) (b) (c)

Figure 3.2: An example of density map grid resampling and division of a cryo-EM density map. (**a**) Density map (EMD-22898) in the original grid. (**b**) Resampled density map to the uniform grid size of 1Å. In this illustration, it is 1 Å× 1 Å× 1 Å. A coarser grid is shown for the purpose of illustration. (**c**) Grid division of the density map where each cube is sub-grid with specific size.

Labeling density maps: Each voxel of a cryo-EM density map has a density value which positively correlates with the possibility of the presence of an atom's signal in the voxel, which can be used as input features to predict positions and types of atoms. After a broad review [70] of the structural properties of current machine learning methods to build atomic models from density maps, we created the following labels for voxels in a density map: atom labels, amino acid labels, and secondary structure labels.

For atom labeling, we created an empty (with values of 0) mask with the same dimension as the `emd_normalized_map.mrc`, indicating that the mask was empty. We then utilized the corresponding PDB file for the density map to label each voxel containing backbone carbon-α atom (Cα) as 1, backbone nitrogen atom (N) as 2, and the carbonyl backbone carbon atom (C) as 3. The mask created from the `emd_normalized_map.mrc` map is a 3D grid, where the location of each voxel is determined by indices (i, j, k). But the corresponding PDB file used to label the voxels are in another 3D coordinate system (x, y, z). Therefore, we calculated the corresponding indices of each backbone atom (Cα, N and



(a) (b) (c) (d)

Figure 3.3: An example of labeling a cryo-EM density map. (**a**) The density map of SARS-CoV spike gycoprotein, EMD-22898 visualized at recommended contour level of 0.7. (**b**) Three different types of protein backbone atoms (Cα, N, C) labeled in different colors. (**c**) Three different secondary structure elements labeled in different colors. (**d**) Twenty different amino acid labeled in different colors. The image are generated using UCSF ChimeraX's [9] surface color by volume data value.

C) in the mask from it's atomic coordinates using the Formula 3.1, where i, j, k are the grid indices of the atom in the mask, x, y, z are the coordinates in the PDB file, $origin_x, origin_y, origin_z$ are the origin of x, y, z axis respectively found in the `emd_normalized_map.mrc` map, and $voxel_x, voxel_y, voxel_z$ are the voxel size of x, y, z axis respectively found in `emd_normalized_map.mrc` map.

$$i = \lceil \left(\frac{\lfloor (z - origin_z) \right)}{voxel_z} \right) ; j = \lceil \left(\frac{\lfloor (y - origin_y) \right)}{voxel_y} \right) ; k = \lceil \left(\frac{\lfloor (x - origin_x) \right)}{voxel_x} \right) \tag{3.1}$$

Similar to the atoms labeling, we created a second mask containing amino acid type labels. We labeled each voxel containing a Cα atom with one of 20 different types of standard amino acids, represented by 20 numbers from 1 to 20, while 0 denotes the absence of Cα atom or unknown amino acid type. Moreover, following the same approach used for atoms and amino acid types labeling, we created a third mask for secondary structure labels. We used UCSF ChimeraX [9] to identify and extract the secondary structure type (coil, β-strand, and α-helix) for each Cα atom from the PDB file. The extracted secondary structures were then used to label each voxel of the mask that contains a Cα atom. We used 1, 2, and 3 to represent coil, α-helix, and β-strand, respectively. Figure 3.3 shows an example of density map labels.

Data splits: In the phase of training deep learning models, the training dataset is used to optimize the parameters (i.e., weights) of the models, and the validation dataset is used to evaluate their performance on the data not used to train the model, which can be used to reduce/avoid the overfitting and select a final trained model for test. Subsequently, the selected model is blindly tested on the test dataset to provide an unbiased evaluation of the model's capability of generalizing to new cryo-EM density maps not used in both the model training and selection. We selected 208 cryo-EM density

Table 3.2: The resolution distribution of the maps in the train and validation datasets. 'Full' denotes the entire dataset[1], while 'Small' denotes a smaller subset of the dataset[2]. The IDs for the training and validation data are provided in the Cryo2StructData Dataverse[3, 4].

	Train Set		Validation Set	
Resolution Range (Å)	Full	Small	Full	Small
1.0 - 2.0	62 (0.93%)	34 (2.02%)	5 (0.67%)	4 (2.14%)
2.0 - 3.0	2147 (32.27%)	535 (31.8%)	222 (30%)	55 (29.41%)
3.0 - 4.0	4443 (66.79%)	1111 (66.13%)	513 (69.32%)	128 (68.45%)
Total	6652 (100%)	1680 (100%)	740 (100%)	187 (100%)

maps, which had previously been used as test data in the previous works [56, 29], from our dataset to create the test dataset [76]. The remaining 7,392 density maps were split into the training dataset consisting of 6,652 maps and the validation dataset consisting of 740 maps according to 90% and 10% ratio. Table 3.2 reports the number/percentage of maps in the training and validation datasets for each resolution range, indicating the statistics for the two datasets is largely consistent. The resolution value of a cryo-EM maps refers to the level of detail at which the structural features of the biological molecules can be visualized within the 3D map. A smaller resolution value indicates higher resolution and better definition of structural features, while a larger resolution value corresponds to lower resolution and less distinct features. As such, we used resolution-based splitting of the cryo-EM density maps for training and validation of our deep learning model. Additionally, the ID-based splits for the full dataset are provided in the Cryo2StructData Dataverse [1] because the unique IDs assigned to each cryo-EM density map can be influenced by factors such as the type of entry, metadata, and curation criteria. If necessary, users may choose to split Cryo2StructData into training, validation and test datasets differently.

Grid Division: The dimensions of density maps of different proteins vary and are usually too large to fit into the memory of a standard GPU for training deep learning models. Similar to the approach employed in DeepTracer [56], Haruspex [10], CR-I-TASSER [50], Emap2sec [52], and Cascaded-CNN [47], we performed grid division to divide the density maps into 3D subgrids with dimension of 32 $\times$ 32 $\times$ 32 overlapped by 6 voxels on each face of the subgrid to train deep learning methods. We choose the dimension 32 as it is big enough to capture the patterns (e.g. six turns of alpha helix) in the data and is small enough to be used with GPUs effectively. In the inference stage, the predictions for the sub-grids can be stitched back to obtain the prediction for the full density map by concatenating the predictions for the central 20 $\times$ 20 $\times$ 20 core voxels of the sub-grids. This approach allowed us to preserve the spatial information of the density maps and overcome the abnormal predictions for

the voxels being cut off at boundaries of sub-grids during the grid division. Figure 3.2c shows an example of grid division. The sub-grids of each density map along with their corresponding labeled mask maps are saved as a single entity in numpy array for deep learning training and test. In total, there are $\sim$ 39 million training sub-grids and $\sim$ 4 million validation sub-grids. Moreover, the scripts of dividing density maps are provided at the Cryo2StructData's GitHub repository for users to create sub-grids of their defined dimensions if necessary.

3.4 Data Records

Cryo2StructData Dataverse [1] contains the necessary metadata, 3D density map, and mask files to enable AI experts without much domain knowledge to develop AI methods for modeling protein structures from cryo-EM density maps.

The following data files for each cryo-EM density map are provided in its individual directory:

- The deposited cryo-EM density map with its corresponding protein model in the PDB format, the original protein sequence in the FASTA format [77], and the sequence extracted from the PDB file.
- The Clustal Omega [78] alignment between the primary sequence listed in the metadata of the PDB entry and the sequence extracted from the 3D coordinates in the PDB file. The two sequences are usually highly similar but not identical because some residues in the original sequence may be disordered and has not been modeled (i.e, no x, y, z coordinates) in the PDB file.
- The resampled and normalized cryo-EM density map generated from the depoisted density map. The values of the voxels in this density map were normalized into the range [0, 1] from their values in the deposited density map and are ready for being used as input for AI models.
- The labeled masks in which the voxels containing the key backbone atoms (carbon-alpha (Cα), nitrogen (N) and carbonyl backbone carbon (C)), the Cα atoms only, the twenty different types of amino acids that Cα atoms belong to, and the three different types of secondary structures (helix, strand, coil) of the Cα atoms are labeled. The labeled masks can be used as targeted outputs for AI and machine learning models to predict from the resampled and normalized density maps (input).

A comprehensive documentation of the Cryo2StructData is available within the Cryo2StructData

Dataverse [1, 2, 74]. This documentation provides in-depth insights into the dataset, elucidating the composition of data files, the structure of directories, and the overall organization of the dataset.

3.5 Technical Validation

We validated every step of the data processing pipeline as shown in Figure 3.1 and described in `Cryo2StructData Preparation` 3.3 section. The source code used to validate data preparation pipeline is released at the GitHub repository of Cryo2StructData for users to reproduce the process.

3.5.1 Validation of Preprocessing Step

The preprocessing phase of Cryo2StructData preparation performs two steps : **(a)** cryo-EM density map resampling and **(b)** cryo-EM density map normalization. We verified both steps for all the cryo-EM density maps present in Cryo2StructData. Additionally, since each cryo-EM density map contains a unique EMDB-ID, we verified the availability of duplicate density maps and found none.

Resampling validation: During the initial stage of data preparation, the density maps were resampled to a uniform voxel grid with a voxel size of 1 Å. To assess the accuracy of this resampling process, we performed validation checks to confirm the voxel size of all density maps within the Cryo2StructData. The validation procedure utilized the `mrcfile` package [75], a Python implementation of the MRC2014 file format [75], widely utilized in structural biology for storing cryo-EM density maps. The results of the validation phase clearly demonstrated that all cryo-EM density maps in the Cryo2StructData exhibited a voxel size of 1 Å. This validation, in turn, confirms the precision and consistency of the resampling step, substantiating the attainment of the desired voxel size of 1 Å.

Normalization validation: To assess the validity of the normalization step in the data preparation process, we performed checks to identify any outliers and ensure that all density values were confined within the range of [0, 1] for each density map. The validation check did not find any outliers, and all density values were found to lie within the specified range [0, 1]. This validation further corroborates the consistent implementation of the preprocessing step, as all density maps in the Cryo2StructData exhibited values within the defined range [0, 1].

Figure 3.4 displays the statistical characteristics of a density map, both before and after the preprocessing step, as visualized using UCSF Chimera [9]. By analyzing Figure 3.4b, we can further confirm that the voxel size is 1 Å, and the density values fall within the range of [0, 1], as expected.

This provides additional evidence of the correct implementation of the resampling and normalization steps.

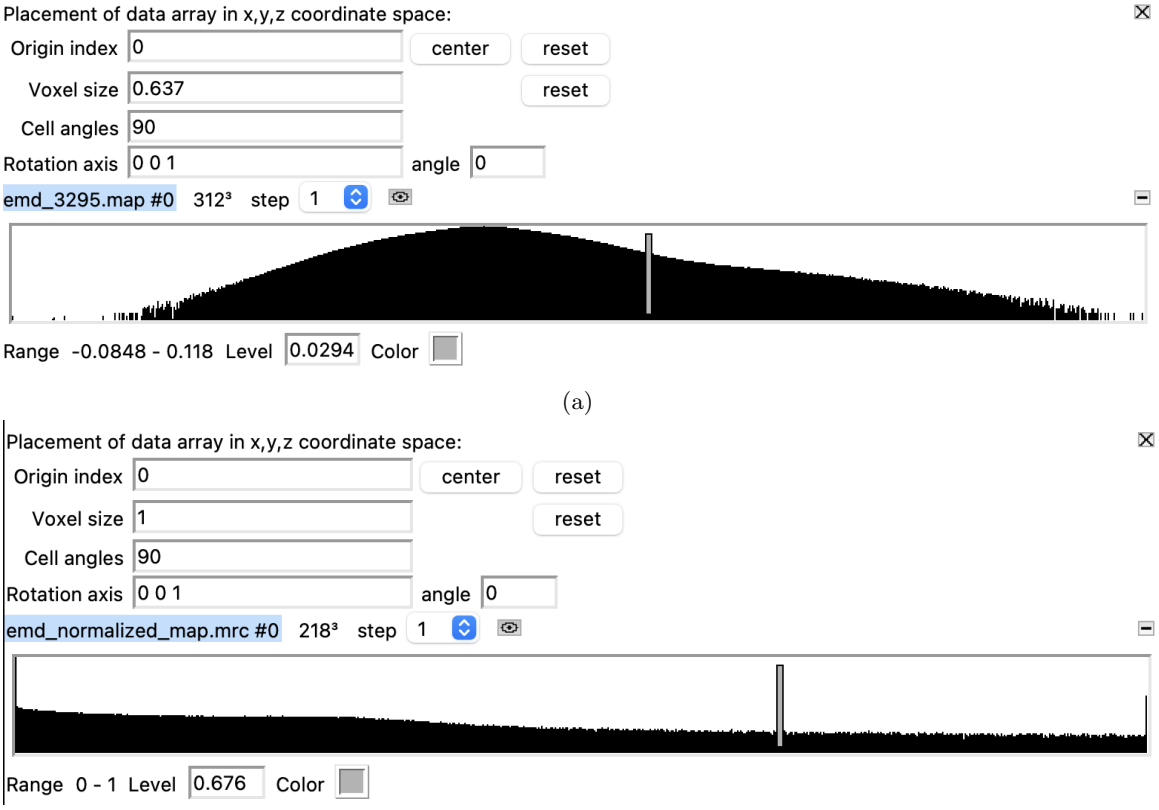


Figure 3.4: The cryo-EM density map of human p97 bound to UPCDC30245 inhibitor (EMD-3295). (a) Density values in raw cryo-EM density map. (b) Density values after preprocessing (resampled and normalized). The voxel size is 1 Å, and the density values are between [0 - 1]. The image is extracted from volume viewer of UCSF Chimera [9].

3.5.2 Validation of Labeling Step

We validated the labeling step by leveraging the mask map of labeled Cα atoms and their corresponding known atomic backbone structure. To obtain approximate 3D coordinates (x, y, z) from their respective indices (i, j, k), we applied Formula 3.2. The grid indices, (i, j, k), were assigned values of either 0 (indicating the absence of a Cα atom) or 1 (indicating the presence of a Cα atom) by the data preparation pipeline. After converting the indices to their associated coordinates for each Cα atom, we observed, as expected, that the protein backbone structure was not aligned. Consequently, a direct structure-to-structure comparison for evaluation was not feasible. To assess the accuracy of the conversion step, we performed a comprehensive search for the converted coordinates within the

known Cα backbone structure. During the search process, we kept track of the number of precise Cα atoms matching with the known backbone structure. We randomly selected approximately 4,450 (>55%) of the Cryo2StructData to verify the conversion process. We found that around 4,185 (94%) of the maps had exact matching Cα positions for all the voxels. For the remaining 6% of the maps, more than 93% of the voxels had exact match, while less than 7% of the voxels deviated slightly from the exact match. The loss of precision in a small portion of Cα positions in a small portion of density maps is expected due to the interchange between floating-point numbers and integers during the labeling and evaluation steps. Therefore, this validation confirm the accuracy of the labeling step, as smaller than 0.5 angstrom (Å) deviations of a small portion of Cα (about 6% $\times$ 7%) positions is expected.

$$x = (k \times voxel_x) + origin_x; y = (j \times voxel_y) + origin_y; z = (i \times voxel_z) + origin_z \tag{3.2}$$

3.5.3 Validation of MRC files

We conducted a comprehensive validation of the entire Cryo2StructData by executing a series of tests to verify its compliance with the MRC2014 format specification [75] . These tests were performed on the data listed below, which pertains to each density map within the Cryo2StructData:

a) `emd_normalized_map.mrc` : This file is used as an input to the deep learning-based model for training, validation and inference step. As such, this is an important file that needs to pass all the tests.

b) `atom_ca_emd_normalized_map.mrc` : This is the labeled Cα-only mask map used as labels for training the deep learning-based model, which identifies the Cα atoms in the input density map.

c) `atom_emd_normalized_map.mrc` : This is the labeled backbone atom (Cα, C, and N) mask map used as labels for training the deep learning-based model, which identifies the backbone atoms in the input density map.

d) `amino_emd_normalized_map.mrc` : This is the labeled amino acid type mask map used as labels for training the deep learning-based model, which identifies the amino acid residue types of Cα atoms in the input density map.

e) `sec_struc_emd_normalized_map.mrc` : This is the labeled secondary structure (coil, α-helix, and β-strand) mask map used as labels for training the deep learning-based model, which

identifies the secondary structure types of Cα atoms in the input density map.

Each of the above files went through a series of tests and checks to identify any potential problems. These files undergoes validation using the `mrcfile` package, specifically the `mrcfile.validate` function. The `mrcfile` Python library is maintained by the Collaborative Computational Project for Electron cryo-Microscopy (CCP-EM) [79] and serves for reading, writing, and validating MRC2014 files [75]. The following tests were conducted to validate the Cryo2StructData. We provide a brief description of these tests below and refer users to the MRC2014 paper [75] for detailed explanations of these tests.

a) `MRC format ID string` : The `map` field in the header should contain "MAP". The character string 'MAP ' is used to identify file type.

b) `Machine stamp` : The machine stamp should contain one of `0x44 0x44 0x00 0x00`, `0x44 0x41 0x00 0x00` or `0x11 0x11 0x00 0x00`.

c) `MRC mode` : The `mode` field should be one of the supported mode numbers:

   (a) **0** 8-bit signed integer

   (b) **1** 16-bit signed integer

   (c) **2** 32-bit signed real

   (d) **4** transform : complex 32-bit reals

   (e) **6** 16-bit unsigned integer

   (f) **12** 16-bit float

d) `Map and cell dimensions` : The header fields `nx` , `ny`, `nz`, `mx`, `my`, `mz`, `cella.x`, `cella.y`, and `cella.z` must all be positive numbers.

e) `Axis mapping` : The header fields `mapc`, `mapr`, and `maps` must contain the values 1, 2, and 3 - in any order.

f) `Volume stack dimensions` : If the spacegroup is in the range 401-630, representing a volume stack, the `nz` field should be exactly divisible by `mz` to represent the number of volumes in the stack.

g) `Header labels` : The `nlabl` field should be set to indicate the number of labels in use, and the labels in use should appear first in the label array.

h) `MRC format version` : The `nversion` field should be 20140 or 20141 for compliance with the MRC2014 standard.

i) `Extended header type` : If an extended header is present, the `exttyp` field should be set to indicate the type of extended header.

j) `Data statistics` : The statistics in the header should be correct for the actual data in the file, or marked as undetermined.

k) `File size` : The size of the file on disk should match the expected size calculated from the MRC header.

During the checks, if the file is completely valid, the `mrc.validate` function returns `True`; otherwise, it returns `False`. Seriously invalid files trigger a `RuntimeWarning`. All data items listed above successfully passed the validation checks for the entire collection of cryo-EM density maps in the Cryo2StructData. This affirms the dataset's full compliance with the MRC2014 format, confirming its validity as a set of density maps and masks suitable for application in AI-based methodologies.

3.5.4 Validation using Deep Learning

To further validate the utility and quality of Cryo2StructData, we trained and test two deep transformer-based models [72] on Cryo2StructData to predict backbone atoms and amino acid types from density maps respectively. Moreover, we used predicted Cα atom positions and probabilities of 20 amino acid types to construct a Hidden Markov Model (HMM) model [80, 72] whose hidden states represent predicted Cα atoms. The HMM model was used to align protein sequences with predicted Cα positions to build the backbone structures of proteins via a customized Viterbi algorithm [72, 80]. In the subsequent sections, we present a concise overview of the deep learning model and the HMM-guided alignment technique applied to predicted Cα voxels. For a comprehensive understanding of implementation details and in-depth analysis involving additional atomic structures, we direct readers to the Cryo2Struct paper [72] or the next chapter of this dissertation.

Training and Testing Deep Transformer Models on Cryo2StructData

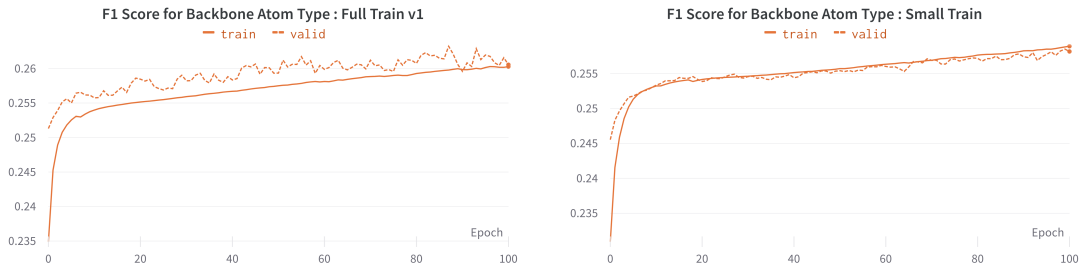
One transformer model was trained to classify each voxel of a sub-grid of a density map into one of four different classes representing three backbone atoms (Cα, C and N) and absence of any backbone atoms. Another model was trained to classify each voxel of the sub-grid into one of twenty-one different amino acid classes representing twenty different amino acids and unknown or absence of

amino acid. Additionally, we also trained a transformer model to predict the secondary structure of voxels, which was not used to construct the HMM. In the inference stage, the predictions for all the sub-grids of a full density map are combined to generate a prediction for the entire map. We trained different sets of models with different parameters (measured in millions), dataset sizes, and extraction layers, as shown in Table 3.3.

Table 3.3: Training on the Cryo2StructData with different deep learning models and data sizes. Refer to Table 3.2 for details on training and validation splits. "AHead" refers to the number of attention heads for each transformer layer. "Models" refers to the trained and released model for amino acid type (amino), atom type (atom), and secondary structure type (ss) prediction in the Cryo2StructData Dataverse [3, 4]. ST refers to Small Train. FT refers to Full Train.

	Data Size	Params (M)	Layers	AHead	Extraction layer	Models
ST	1,867	35.60	4	6	[1, 2, 3, 4]	amino, atom, ss
FT v1	7,392	92.28	12	12	[3, 6, 9, 12]	amino, atom
FT v2 + Seq	7,392	179.46	24	12	[3, 6, 12, 24]	amino, atom

For the `Small Train`, we used 25% of the Cryo2StructData (small subsample [2]), with 4 transformer layers and 6 attention heads. For `Full Train v1`, we utilized all the training and validation data from the Cryo2StructData (full data [3]), with 12 transformer layers and attention heads. We used the full trained models [3] to construct HMMs (Section '*HMM-Guided Alignment of Cα Atoms and Protein Sequences*'). The training and validation macro F1 score plots generated for `Full Train v1` and `Small Train` are shown in Figure 3.5. These plots illustrate the increase in F1 scores for both full and small training, providing further evidence that models trained on Cryo2StructData data can successfully learn and identify patterns in the data.

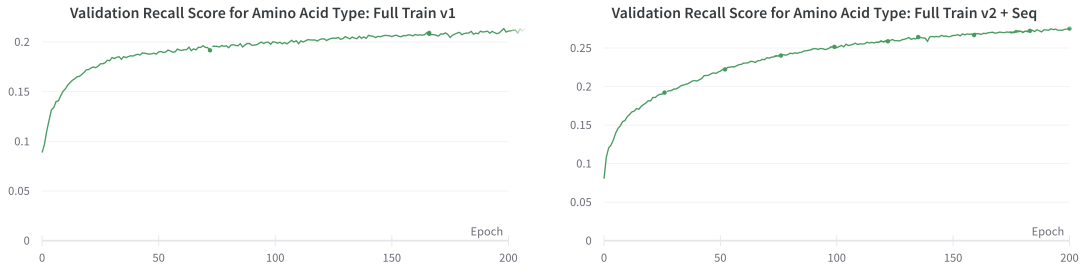


(a) (b)

Figure 3.5: The training and validation F1 scores for transformer-model trained on Cryo2StructData. (a) Full Train v1. (b) Small Train.

For `Full Train v2 + Seq`, we utilized all the training and validation data from the Cryo2StructData (full data [3]), with 24 transformer layers and 12 attention heads. In addition, we integrated a protein language model (ESM-2 [16]) generated sequence features into each protein density map, enhancing

the sub-grid input with sequence information before passing it into the transformer layer. The per-residue representation is extracted using the ESM-2 pretrained model and then averaged to generate the per-sequence representation. We employed the ESM-2 pretrained model, which comprises 33 layers and 650M parameters, to generate embedding dimensions of 1,280. This computation was performed on the CPU of the Andes supercomputer [81], utilizing its 256 GB RAM. To accommodate memory constraints, we utilized a sequence cutoff of 2,750 residues for each protein sequence. We extracted the encoded sequence representation from the 3^{rd}, 6^{th}, 12^{th}, and the final 24^{th} layers. We noticed an increase in both validation recall and precision for the amino and atom type prediction tasks. A comparison of recall scores between `Full Train v1` and `Full Train v2 + Seq` is presented in Figure 3.6. By incorporating sequence information and enhancing the model's complexity, the transformer-based model trained on the Cryo2StructData demonstrates improved learning. The sequence for each density map is made available within the Cryo2StructData, providing a convenient resource for AI-based researchers to utilize this complementary data and enhance the prediction capabilities of deep learning models.



(a) (b)

Figure 3.6: The validation recall scores for amino acid type prediction during training. (a) Full Train v1. (b) Full Train v2 + Seq. Incorporating sequence information to train the model improves the recall.

HMM-Guided Alignment of Cα Atoms and Protein Sequences

Connecting the predicted carbon-α atoms into protein chains and determining their amino acid types accurately is one of the most challenging aspects of building protein model from a cryo-EM density map. To address this challenge, we constructed a HMM [72] from Cα atoms and their amino acid types predicted by the deep transformer models. This model aligns the known protein sequences with the predicted hidden states denoting Cα atoms in the HMM, allowing us to link Cα atoms into protein chains and assign amino acid types to the atoms in a single step.

HMMs are generative models described by transition probability between hidden states and

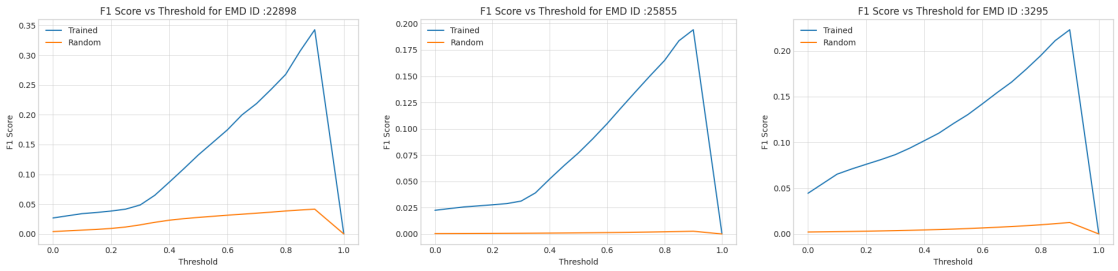
emission probabilities of generating observed symbols from hidden states. Specifically, the predicted Cα atoms (voxels) from the atom type prediction are used as hidden states in the HMM, which are fully connected to generate the sequence of a protein. The amino acid emission probabilities of each Cα hidden state are the normalized geometric mean of the probabilities of twenty different amino acid types predicted by the amino acid type prediction model and their background probabilities precomputed from the protein sequences in the training data. The geometric mean is computed as $\sqrt{a \times b}$, where a and b represent the predicted probability for an amino acid type and its background frequency, respectively. The transition probability between any two Cα hidden states are calculated according to the euclidean distance (x) between two Cα atoms based on the Gaussian distribution with a mean (μ) of 3.8047 and a standard deviation (σ) of 0.036 times a fine-tune able scaling factor (Λ) ($\Lambda = 10$).

Because it is not known which Cα state generates the first amino acid of a protein chain, the HMM allows every Cα hidden state to be the initial state. The probability for a Cα state to be the initial state is equal to the probability of it emitting the first amino acid divided by the sum of the probability of every state emitting the first amino acid. The constructed HMM is then used by a customized Viterbi algorithm [72] to compute the most likely path of aligning protein sequences with the Cα hidden states, resulting a determined protein backbone structure. The customized Viterbi algorithm allows any Cα state to occur at most once in the path because one Cα position can be occupied by only one amino acid of a protein.

Evaluation Results on SARS COVID-19 Proteins

We tested the trained deep learning models on three SARS-CoV-2 proteins [12, 82, 13] and seven other proteins. These proteins were not part of training dataset. The voxel-wise predictions of Cα atom were evaluated using F1-score (i.e., geometric mean of precision and recall of Cα predictions). We used F1-score as it is a more balanced metric than the accuracy when there is a significant imbalance in the class distribution (i.e., the portion of voxels containing Cα atoms is very small in this case). We compare the F1 scores of our predictions with those of the random predictions in Figure 3.7. The former is much better the latter.

We further evaluated the Cα backbones aligned by the HMM for the two SARS COVID-19 proteins [12, 13] and the human p97 protein [14] against the known protein structures as shown in Figures 3.9, 3.11, and 3.13. To adhere to the commonly practiced approach in the literature [56, 29], we used phenix.chain_comparison tool to compute the root mean squared distance (RMSD), the percentage of matching Cα atoms, and the percentage of sequence identity. Phenix's chain_comparison



(a) (b) (c)

Figure 3.7: F1 scores of Cα atom predictions for the density maps of two SARS COVID-19 (EMD-22898 [12], EMD-25855 [13]) and a human p97 (EMD-3295 [14]) test proteins. The curve in blue denotes how F1 score of the deep transformer trained on Cryo2StructData changes with respect to the threshold on predicted Cα atom probabilities and the curve in orange the random predictions.

tool compares two structures to identify the number of matching Cα atoms. Using this approach, it calculates the matching percentage, which represents the proportion of residues in the known structure that have corresponding residues in the reconstructed backbone structure. Similarly, it reports the sequence matching percentage indicating the percentage of matched residues that have the same amino acid type. The results in Table 3.4 shows that using the deep learning predictions generated from the model trained on Cryo2StructData, combined with HMM alignment strategy, we were able to determine the backbone structures of the 10 proteins having varying resolutions and number of residues with the good accuracy on average. The average RMSD, where lower values are preferable, is 1.51 Å. The matching percentage and sequence identity (ID) percentage, where higher values are better, is 60.4% and 37.94%, respectively. The average resolution and the number of residues modeled are 2.77 Å and 2222.3, respectively. Furthermore, the Cryo2Struct paper [72] presents a thorough examination of atomic structures generated by a deep learning model trained on Cryo2StructData. Figures 3.8, 3.10, 3.12, 3.9, 3.11, and 3.13 shown in `the Several examples of predicting C`α` atoms and reconstructing protein backbone structures section` illustrate several good, detailed examples of predicting Cα atoms and modeling protein backbone structures from density maps.

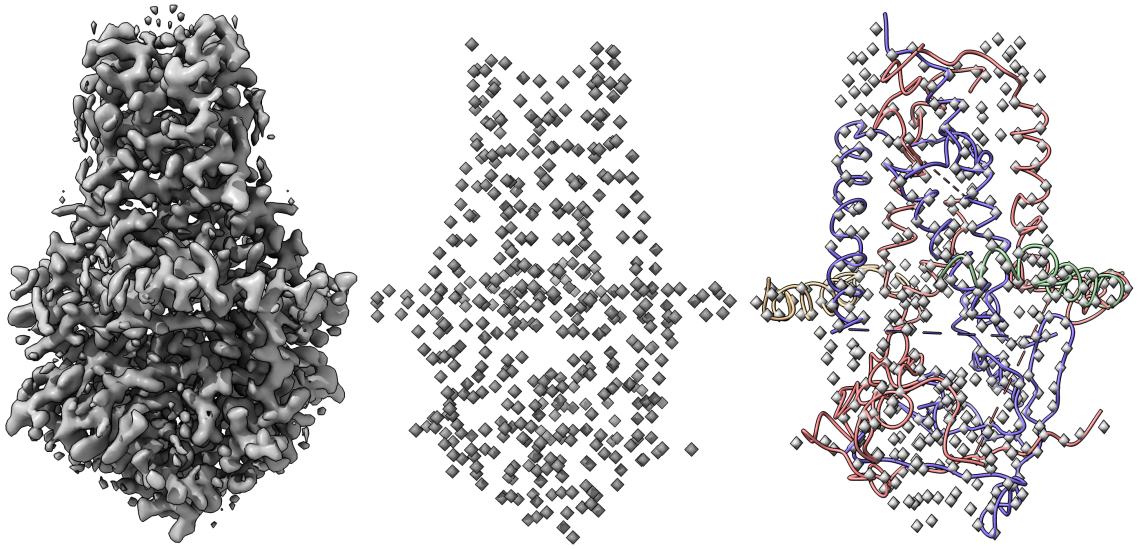
3.5.5 Several examples of predicting Cα atoms and building protein backbone structures

Figures 3.8, 3.10, 3.12 visualize and analyze the predicted Cα backbone structures of three proteins with respect to the true Cα atoms of the density maps. The predicted backbone models were built by the deep transformer trained on Cryo2StructData and Hidden Markov Model. Figures

Table 3.4: Evaluation scores for predicted backbone structures of proteins with varying resolutions and numbers of residues.

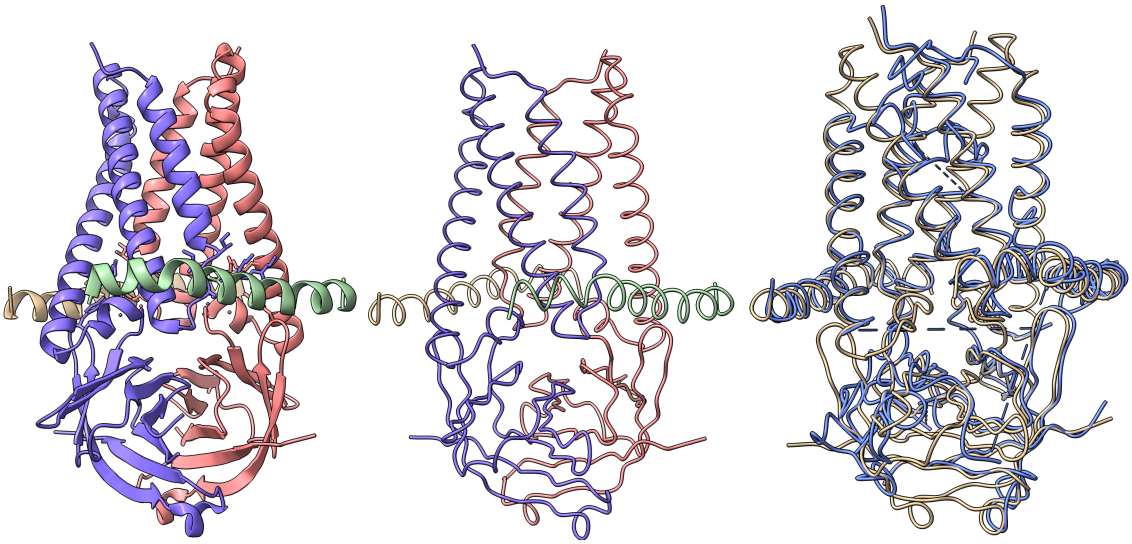
EMDB ID	Reso.(Å)	PDB ID	Residues	RMSD(↓)	Match(%, ↑)	Sequence ID(%, ↑)
22898	2.08	7KJR	448	1.09	80.4	86.7
30210	2.50	7BV2	1036	1.31	69.7	73.0
3061	3.4	5A63	1223	1.66	49.6	14.2
8117	2.95	5IRX	1726	1.64	50.0	28.2
6551	3.80	3JCF	1745	1.83	52.0	15.0
8764	2.94	5W3S	1940	1.56	58.5	30.7
25855	2.25	7TEY	2703	1.42	71.6	85.0
6634	3.30	3JD1	2976	1.71	52.3	13.2
2984	2.20	5A1A	4088	1.40	57.2	12.4
3295	2.3	5FTJ	4338	1.55	62.7	21.0
Average	2.77		2222.3	1.51	60.4	37.94

3.9, 3.11, 3.13 compare the modeled backbone structures of the same three proteins with their true backbone structures extracted from the known protein structures. We used UCSF ChimeraX [9] for the visualization of the cryo-EM density maps and the protein structures. Generally, the modeled backbone structures match the true Cα atoms in the density maps and the true backbone structures reasonably well.



(a) (b) (c)

Figure 3.8: (a) The cryo-EM density map of SARS-CoV-2 ORF3a (EMD-22898) visualized at the recommended contour level of 0.7 (10.3 σ). The map dimension is $300 \times 300 \times 300$ and has density values between $-2.319 - 3.909$. The voxel dimensions is $0.727 \times 0.727 \times 0.727$ (Å). (b) The true Cα atom voxels (mask) extracted from the density map. (c) The predicted Cα model is overlaid with true Cα atom voxels.



(a) (b) (c)

Figure 3.9: (a) The known protein structure corresponding to the density map EMD-22898 (PDB code 7KJR). The known structure has 448 residues. (b) The true Cα backbone structure extracted from the known PDB structure. (c) The superimposition of the predicted backbone (blue) structure with the known backbone structure (gold).

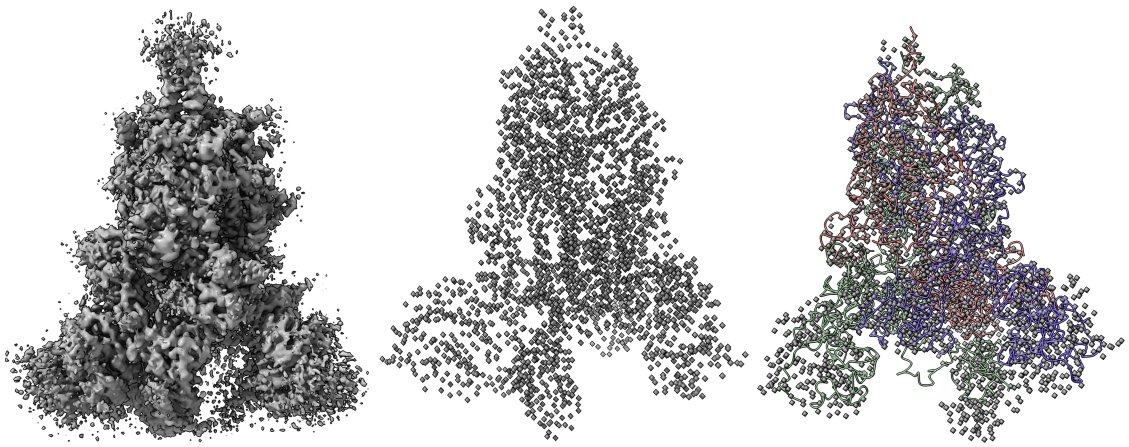
3.6 Usage Notes

Cryo2StructData is specifically designed for the task of building protein models from cryo-EM density maps. The prepared cryo-EM density maps available in Cryo2StructData Dataverse[1, 2] can be visualized using UCSF Chimera [9] . For the analysis of these maps, it is recommended to use the `mrcfile package` [75] in Python.

The data preprocessing scripts in the Cryo2StructData GitHub repository can be modified or utilized to generate custom data tailored to user's specific needs or reproduce the data generation process. The label generation process scripts are also included in the GitHub repository.

Cryo2StructData contains two types of *data splits*. The first type is resolution-based splits, and the second type is ID-based splits. We trained our deep learning model on the resolution-based splits dataset, as described in Table 3.2. Users have the option to use the splits we have created or can create their own splits based on their needs. Additionally, we provide a smaller downsampled version [2] of the entire Cryo2StructData for users who do not have access to large computational resources. The split-based information is available in Cryo2StructData Dataverse [3, 4].

The provided *trained deep learning models* [3, 4] for atom and amino acid-type prediction, are essential for predicting voxels within the input cryo-EM density maps. The Cryo2StructData GitHub repository contains a deep learning inference program that utilizes these trained models for voxel-wise



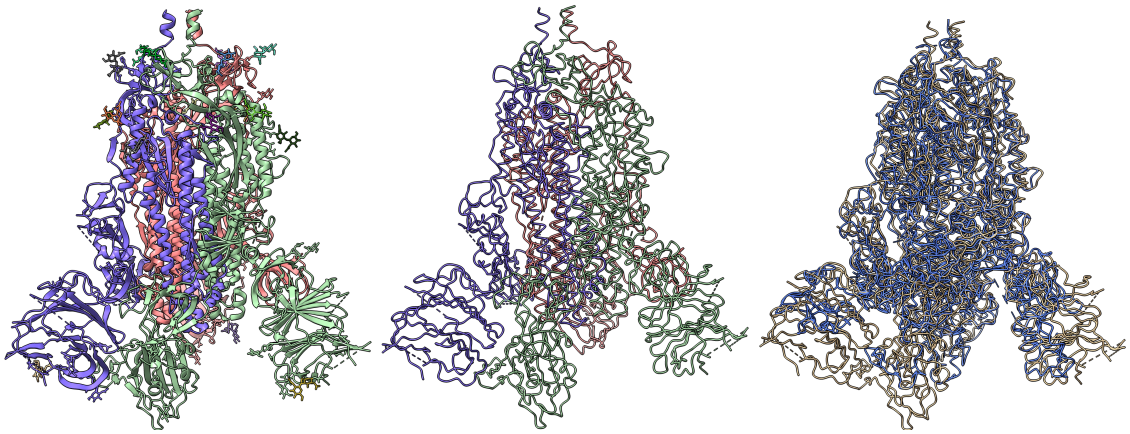
(a) (b) (c)

Figure 3.10: (a) The cryo-EM map of SARS-CoV-2 Delta (B.1.617.2) spike protein (EMD-25855) visualized at recommended contour level of 0.121 (5.0 σ). The map dimension is $400 \times 400 \times 400$ and has density values between $-0.467 - 1.315$. The voxel dimensions is $1 \times 1 \times 1$ (Å). (b) The true Cα atom voxels (mask) extracted from the density map. (c) The predicted Cα model is overlaid with true Cα atom voxels.

prediction. Our innovative *Hidden Markov Model* as detailed in `HMM-Guided Alignment of C`α `Atoms and Protein Sequences`, a fast and efficient method for aligning predicted voxels to form a protein backbone structure in a single step, is written in C++ and is available in the Cryo2StructData GitHub repository. Additionally, both the Cryo2StructData Dataverse and the Cryo2StructData GitHub repository contain detailed information about the dataset and instructions for building the protein backbone model from cryo-EM density map, enabling users to seamlessly utilize the data and code. Additionally, we direct readers to the Cryo2Struct paper [72], which utilizes Cryo2StructData for training its deep learning model. Cryo2Struct [72] also provides comprehensive analyses of structures modeled from cryo-EM density maps. The next chapter in this dissertation presents Cryo2Struct.

The integration of *ESM-2 [16] generated sequence features* into the protein density maps has yielded promising results, notably improving validation recall and precision for amino acid type prediction tasks, as shown in Figure 3.6. This plot demonstrates the impact of incorporating sequence information into the density map. The sequence for each cryo-EM density map is made available within the Cryo2StructData Dataverse [1, 2], providing a convenient resource for researchers to utilize this complementary data and enhance the prediction capabilities of deep learning models.

The cryo-EM density map data structure is prepared as a 3D (three dimension) grid of values organized into layers of rows and columns. The voxel grids can be inferred as a regularly spaced point cloud where each point is a voxel. Other approaches, such as modeling input cryo-EM density



(a) (b) (c)

Figure 3.11: (a) The known protein structure of the density map EMD-25855 (PDB code 7TEY). The known structure has 2,703 residues. (b) The true Cα backbone structure extracted from the known protein structure. (c) The superimposition of the predicted backbone structure (blue) with the known backbone structure (gold).

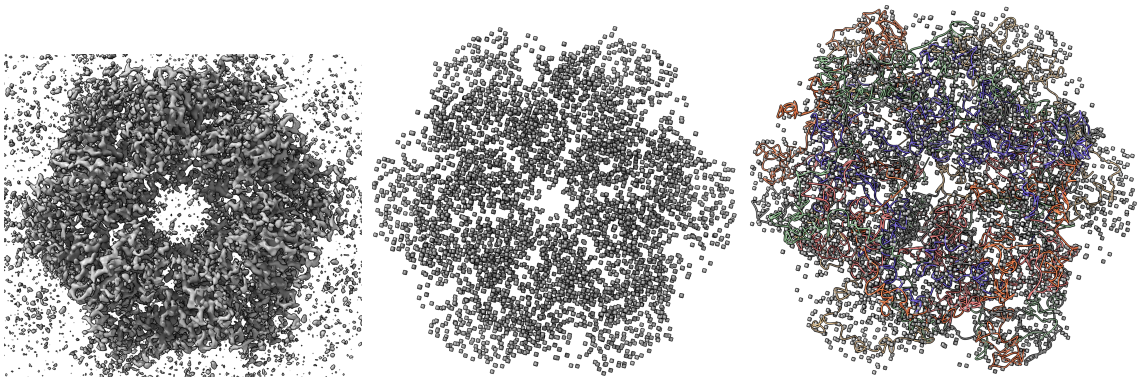
maps as 3D point clouds and leveraging 3D Graph Neural Networks [62], can be considered.

The dataset of this scale allows researchers to train and test robust and powerful deep learning models to predict the positions of protein backbone atoms (e.g., Cα atoms) and their amino acid types in 3D cryo-EM density maps, which can be linked together to build 3D protein models from scratch without using any known structural information as templates. The AI-powered de novo modeling of protein structures from cryo-EM density maps will significantly extend the capability and efficiency of cryo-EM techniques to solve the structures of large protein complexes and assemblies.

3.7 Code availability

The source code and instructions to reproduce our results are freely available at

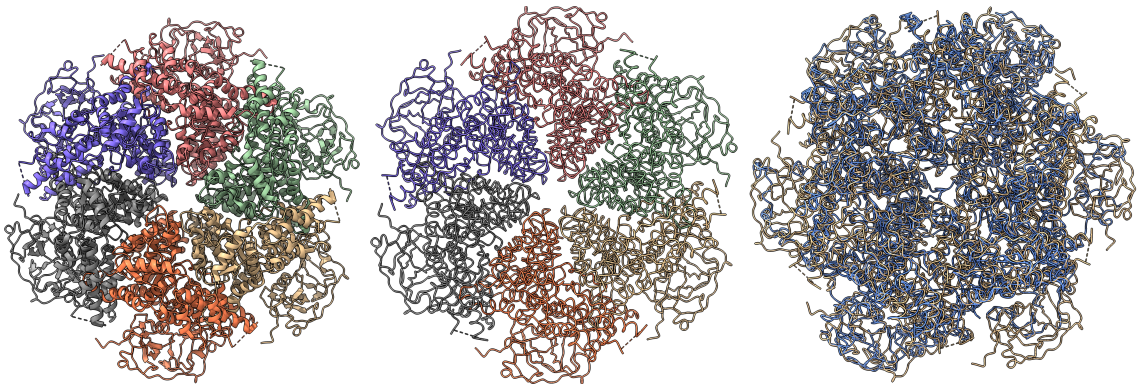
`https://github.com/BioinfoMachineLearning/cryo2struct`. To keep the data files of Cryo2StructData permanent, we published all data to the Harvard Dataverse (`https://dataverse.harvard.edu/dataverse/Cryo2StructData`), an online data management and sharing platform with a permanent Digital Object Identifier number for each dataset. The Cryo2StructData Dataverse comprises the Full Cryo2StructData, referred to as `Cryo2StructData: Full Dataset` [1](`https://doi.org/10.7910/DVN/FCDGOW`), along with its associated trained deep transformer model and data splits, referred as `Cryo2StructData: Trained Model and Data Splits (Full)` [3] (`https://doi.org/10.7910/DVN/SXNYRE`). Similarly, within the Cryo2StructData Dataverse, we find the Small Subsample of the complete Cryo2StructData, denoted as `Cryo2StructData: Small`



(a) (b) (c)

Figure 3.12: (a) The cryo-EM density map of human p97 bound to UPCDC30245 inhibitor (EMD-3295) visualized at recommended contour level of 0.0375 (3.4 σ). The map dimension is $312 \times 312 \times 312$ and has density values between $0.085 - 0.118$. The voxel dimensions is $0.637 \times 0.637 \times 0.637$ (Å). (b) The true Cα atom voxels (mask) extracted from the deposited density map. (c)The predicted Cα model is overlaid with true Cα atom voxels.

`Subsample Dataset` [2] (https://doi.org/10.7910/DVN/CGUENL), accompanied by its respective trained deep transformer model and data splits, recognized as `Cryo2StructData: Trained Model and Data Splits (Small Subset)` [4] (https://doi.org/10.7910/DVN/DTV4JF). Finally, the test dataset has been made available as `Cryo2StructData: Test Dataset` [76] (https://doi.org/10.7910/DVN/2GSSC9). The metadata of Cryo2StructData is available at https://doi.org/10.7910/DVN/JMN60H.



(a) (b) (c)

Figure 3.13: (a) The known protein structure of the density map EMD-3295 (PDB code 5FTJ). The structure has 4,338 residues. (b) The true Cα atom backbone structure extracted from the known protein structure. (c) The superimposition of the predicted backbone structure (blue) with the known backbone structure (gold).

Chapter 4

DE NOVO ATOMIC PROTEIN STRUCTURE MODELING FOR CRYO-EM DENSITY MAPS USING 3D TRANSFROMER AND HMM

4.1 Abstract

Accurately building 3D atomic structures from cryo-EM density maps is a crucial step in cryo-EM-based protein structure determination. Converting density maps into 3D atomic structures for proteins lacking accurate homologous or predicted structures as templates remains a significant challenge. Here, we introduce Cryo2Struct, a fully automated *de novo* cryo-EM structure modeling method. Cryo2Struct utilizes a 3D transformer to identify atoms and amino acid types in cryo-EM density maps, followed by an innovative Hidden Markov Model (HMM) to connect predicted atoms and build protein backbone structures. Cryo2Struct produces substantially more accurate and complete protein structural models than the widely used ab initio method Phenix. Additionally, its performance in building atomic structural models is robust against changes in the resolution of density maps and the size of protein structures.

4.2 Introduction

Determining the three-dimensional (3D) atomic structures of macromolecules, such as protein complexes and assemblies [59, 83, 66], is fundamental in structural biology. The 3D arrangement of atoms provides essential insights into the mechanistic understanding of molecular function of proteins [65]. In recent years, cryo-electron microscopy (cryo-EM) [84] has emerged as a key technology for experimentally determining the structures of large protein complexes and assemblies [85, 69, 86].

However, modeling atomic protein structures from high-resolution cryo-EM density maps, which constitute a significant portion of the maps deposited in the EMDB [6], is both time-consuming and challenging, especially in the *de novo* setting when accurate homologous or predicted structures for target proteins or their units (chains) are unavailable [70, 87]. Modeling atomic protein structures from cryo-EM maps faces the challenges of identifying atoms of proteins in density maps as well as tracing the atoms into chains to form the backbone structures and registering amino acid sequences with them [26].

Despite the importance of the problem, only a small number of methods have been developed for determining atomic structures from cryo-EM maps, such as Phenix [29], DeepMainmast [26], DeepTracer [56], and ModelAngelo [58]. Phenix is the most widely used standard tool of building atomic protein structures from cryo-EM density maps using classic molecular optimization. DeepTracer provides a web-based deep learning tool for users to predict atomic structures from density maps. ModelAngelo combines information from cryo-EM map data, amino acid sequences, and prior knowledge about protein geometries to refine the geometry of the protein chain and assign amino acid types. DeepMainmast, a recently developed method, integrates AlphaFold2 [34] with a density tracing protocol to determine atomic models from cryo-EM maps. Incorporating accurate AlphaFold-predicted structures into the modeling has significantly improved the quality of the structures determined from cryo-EM density maps [26].

However, modeling multi-chain protein structures from cryo-EM density maps remains a challenging task for the existing methods, particularly when there are inaccurate predicted structures for target protein complexes or their chains to be used as templates. The *de novo* modeling of protein structures from only density maps without using templates is not only practically relevant in this situation, but also can help answer an important question: how much structural information can be extracted from cryo-EM density maps alone? In the *de novo* modeling context, we introduce Cryo2Struct (i.e., cryo-EM to structure), a fully automated, ab initio modeling method that does not require predicted or homologous structures as input to generate 3D atomic structures from cryo-EM density maps. Cryo2Struct first uses a Transformer-based deep learning model with an attention mechanism [63] to identify atoms and their amino acid types in cryo-EM density maps. Then it uses an innovative generative Hidden Markov Model (HMM) [80] and a tailored Viterbi Algorithm [80] to align protein sequences with the predicted atoms and amino acid types to generate atomic backbone structures. Cryo2Struct is rigorously tested on 628 density maps in the stringent ab initio modeling setting in which no homologous/predicted structure is used as template and yields substantially improved modeling accuracy.

4.3 Results

4.3.1 Atomic Structure Modeling Workflow

Cryo2Struct takes a 3D cryo-EM density map and the corresponding amino acid sequence of a protein as input to generate a 3D atomic protein structure as output automatically (Fig. 4.1**a-e**). As in [47], we divide the problem of atomic structure determination from cryo-EM density map into an atom classification (recognition) task and a sequence-atom alignment task. The two tasks are performed by a Deep Learning (DL) block based on a transformer (Fig. 4.1**b**) and an alignment block based on a HMM (Fig. 4.1**d**), respectively. The DL block classifies each voxel (3D pixel) within the cryo-EM density map into different types of backbone atoms (e.g., Cα) or non-backbone voxel and predict their amino acid types, while the alignment block constructs a HMM [80] from predicted Cα atoms (corresponding to the hidden states in the HMM) and aligns the amino acid sequences with them using a customized Viterbi Algorithm, resulting in a sequence of Cα atoms connected as protein chains to form the atomic backbone structure of the protein. Additional details are available in the Methods Section of this chapter.

4.3.2 Predicting Backbone Atom and Amino Acid types using 3D Transformer

The first step of the atomic structure modeling is to detect the voxels in cryo-EM density map that contain backbone atoms and predict their amino-acid types. We designed and trained a 3D transformer-based model to classify each voxel of the cryo-EM density map into one of four different classes representing three backbone atoms (Cα, C, and N), and the absence of any backbone atom. Another 3D transformer-based model was designed and trained to classify each voxel of the cryo-EM density map into one of twenty-one different amino acid classes representing twenty standard amino acids and the absence of an amino acid or unknown amino acid. The models were trained as a sequence-to-sequence predictor, utilizing a Transformer-Encoder [63] to capture long-range voxel-voxel dependencies and a skip-connected decoder to combine the extracted features at different encoder layers to classify each voxel. The models were trained using the large Cryo2StructData dataset [88]. Cryo2StructData is a comprehensive labeled dataset of cryo-EM density maps curated specifically for deep learning-based atomic structure modeling in cryo-EM density maps. The data preparation of Cryo2StructData is described in previous chapter of this dissertation. The models were trained and validated on the entire dataset comprising 6,652 cryo-EM maps for training and 740 cryo-EM

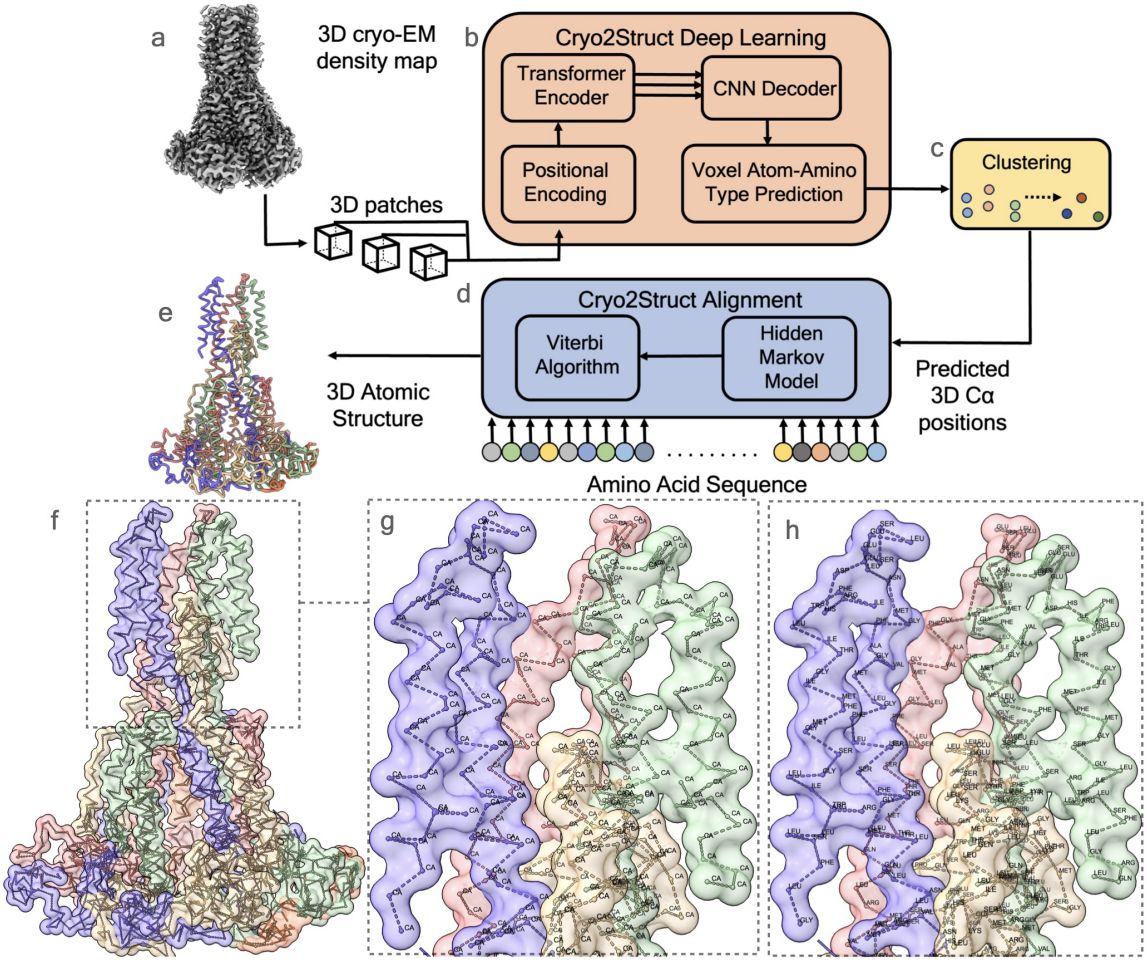


Figure 4.1: An overview of the automated prediction workflow of Cryo2Struct. Given a 3D cryo-EM density map of a protein as input (**a**), the Deep Learning block based on a transformer (**b**) generates a voxel-wise prediction of Cα atoms and their amino acid type. A clustering step (**c**) is used to merge nearby predicted Cα atoms into one atom to remove redundancy. The predicted Cα atoms and their amino acid type probabilities are used by the Alignment block (**d**) to build a Hidden Markov Model (HMM), which is used by a customized Viterbi Algorithm to align the sequence of the protein with it to generate a 3D backbone atomic structure for the protein (**e**). (**f**) shows the skeleton of the Cryo2Struct modeled structure for a test cryo-EM density map having less than 25% sequence identity with the training data released on September 13, 2023 (EMD ID: 41624; resolution 2.8Å), where each chain is colored differently. (**g**) depicts the connected Cα atoms, and (**h**) shows the amino acid types assigned to the Cα atoms; the modeled structure has 1,585 amino acid residues; and the F1 score of Cα atom prediction is 89.1%.

maps for validation first and then were blindly tested on two test datasets. Because some predicted Cα voxels are spatially very close and likely correspond to the same Cα atom, Cryo2Struct employs a clustering strategy to group predicted Cα voxels within a 2Å radius into clusters and select the centrally located Cα voxel in each cluster as the final predicted Cα atom (see the Method Section of this chapter for details).

4.3.3 Aligning Protein Sequence with Predicted Cα atoms

The goal of this step is to connect the predicted, disjoint Cα atoms into peptide chains and assigns amino acid types to them (sequence registering). To achieve the goal, the alignment block constructs a HMM from the predicted Cα atoms and their predicted amino acid type probabilities, in which each predicted Cα atom is represented by one hidden state. The transition probability between two hidden states is assigned according to the spatial distance between their corresponding Cα atoms, and the emission probability of each hidden state for generating 20 different amino acids is assigned according to the predicted probability of 20 different amino acid types for its Cα atom (see more details in Methods Section). The sequence of each chain of the protein is aligned to the HMM by a customized Viterbi algorithm to generate the most probable path of hidden states (Cα atoms). The path for a chain represents the connected Cα atoms of its backbone structure. The paths for multiple chains of a protein together with the sequences aligned with them form the final atomic backbone structure of the protein. Fig. 4.1**f** illustrates a high-quality structure modeled by Cryo2Struct, while Fig. 4.1**g** and 4.1**h** provide a detailed view of the structure. In Fig. 4.1**g**, the predicted Cα atoms are depicted and connected by the alignment block. Fig. 4.1**h** reveals the amino acid-type assignment for each Cα atom.

4.3.4 Comparing Cryo2Struct with Phenix on a Standard Dataset

After Cryo2Struct was trained and validated, we first compared the modeling performance of Cryo2Struct and Phenix [29] on a standard test dataset that was used to benchmark Phenix's map_to_model tool [89]. Most density maps in the dataset are for multi-chain protein complexes, while some of them are associated with single-chain proteins. Their resolution ranges from 2.08 Å to 5.6 Å. The average resolution of the density maps is 3.68 Å. The number of amino acid residues included in the maps varies from 448 to 8,416. These test maps were not present in the training and validation dataset used to train the Cryo2Struct DL model. We chose Phenix as a reference here because it built the structures from the density maps in the same ab initio mode as Cryo2Struct is

designed to do without using homologus or predicted protein structures as input. The structures built by Phenix were downloaded from its website ([89]). The structural models built for the 128 test cryo-EM maps in the test data by Cryo2Struct and Phenix were compared with the true structures in the Protein Data Bank (PDB) to evaluate their quality. We consider the known PDB structure available in PDB as the true structure in this study. The evaluation results in terms of six metrics are presented in Fig. 4.2.

Fig. 4.2**a** plots the recall of Cα atoms of each model built by Cryo2Struct for each of 128 density maps against that by Phenix. The recall (sensitivity) represents the fraction of actual Cα atoms in the true structure that are correctly identified by a model. Cryo2Struct achieves an average recall score of 65%, much higher than 40% of Phenix, indicating that Cryo2Struct recovers a much higher percentage of Cα atoms correctly than Phenix. On 126 out of 128 density maps, Cryo2Struct has a higher recall than Phenix.

Fig. 4.2**b** plots F1 score of Cα atoms of Cryo2Struct against Phenix. The F1 score is the harmonic mean of precision and recall of Cα atoms. The precision (specificity) is the percentage of predicted Cα atoms that are correct ones. The F1 score is a balanced measure because it considers both the specificity and sensitivity of predicted Cα atoms. The average F1 score of Cryo2Struct and Phenix is 66% and 52%, respectively. On 105 out of 128 maps, Cryo2Struct has a higher F1 score.

Fig. 4.2**c** plots the global normalized TM-scores of the models built by the two methods. A standard TM-score measures the similarity between a model and the corresponding known structure, which was calculated by a protein complex structure comparison tool - US-align [90] by enabling its options for aligning two multi-chain oligomeric structures and all the chains, as recommended for aligning biological assemblies. In this analysis, to fairly compare the models built by Cryo2Struct and Phenix that usually have different lengths (numbers of residues), the global TM-score is normalized by the same length of the experimental structure. The TM-score ranges from 0 to 1, with 1 being the best possible score. The average global normalized TM-score of Cryo2Struct is 0.2, more than double 0.084 of Phenix. On 114 out of 128 density maps, Cryo2Struct has a higher normalized TM-score than Phenix.

However, the average global normalized TM-score of both methods is still low. One reason is that TM-score is a sequence-dependent global measure and obtaining a high normalized TM-score requires a high portion of Cα atoms of a large protein complex being not only correctly identified (high recall) but also all correctly linked at the same time, which is still very challenging for the de novo atomic model building from only the density maps that may have missing density values in some regions causing disconnection of Cα atoms. Another reason is that the TM-score computed by

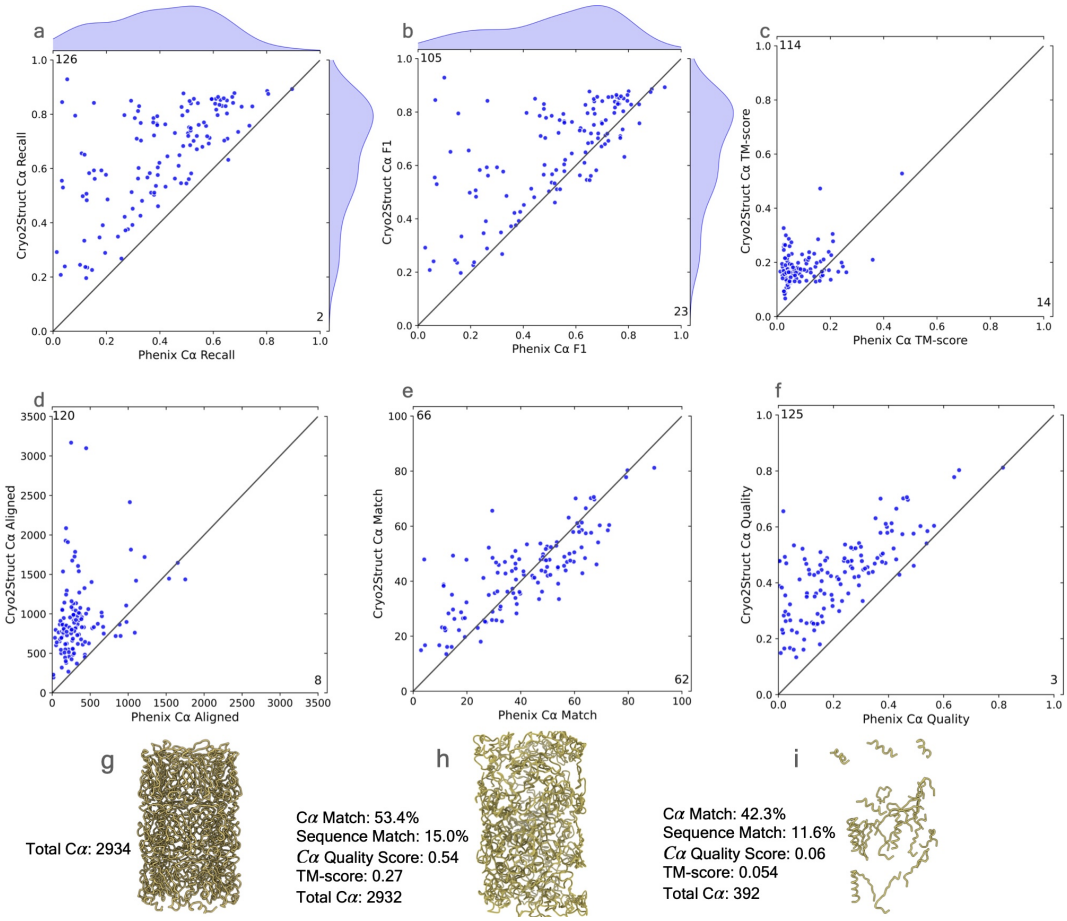


Figure 4.2: The comparative analysis of atomic models built for 128 test cryo-EM maps by Cryo2Struct and Phenix in terms of six metrics. In each panel of an evaluation metric, the score of the model built by Cryo2Struct for each map is plotted against that by Phenix for the same map. A dot above the 45 degree line indicates that Cryo2Struct has higher score than Phenix for the map. The number in the top-left corner represents the total number of maps on which Cryo2Struct has higher scores, while the number in the bottom-right corner denotes the total number of maps on which Phenix has higher scores. (**a**) The Cα recall of the atomic models of Cryo2Struct against Phenix; the recall is defined as the number of Cα atoms in the predicted model that are placed within 3Å of the correct position in the corresponding known structure, divided by the total number of Cα atoms in the known structure. (**b**) The F1 score of Cα, which is the harmonic mean of precision and recall of Cα; it is a balanced measure quantifying a method's ability to make accurate Cα predictions while also capturing as many Cα atoms as possible. (**c**) The TM-score of the atomic models normalized by the length of the known structure; the normalized TM-score is calculated by using US-align to align the atomic models with their corresponding known structures. (**d**) The length of aligned Cα atoms; it is calculated by using US-align to align the predicted model and the known structure. (**e**) The Cα match score of the atomic models; it is calculated by using Phenix.chain_comparison tool to compare them with the known structures. (**f**) The Cα quality score; it is the product of the Cα match score and the total number of predicted residues divided by the total number of residues in the experimental structure; the total number of predicted residues is calculated by Phenix.chain_comparison tool. (**g**) The true structure of EMD ID: 8767 (PDB ID: 5W5F); the map was released on 2017-08-16 with resolution of 3.4 Å. (**h**) The Cryo2Struct model and its scores. (**i**) The Phenix model and its scores.

US-align is normalized by the total length of the known structure that is usually very large (average length of the true structure = 3794.95 residues) rather than the length of a structurally aligned region between a model and the true complex structure. So, if the aligned region has a high TM-score but is only a faction of the entire known structure, the normalized TM-score would still be low. We expect that complementing density maps with the features extracted from protein sequences or AlphaFold-predicted structures as input for deep learning to predict Cα atoms and amino acid types can further improve the normalized TM-score [70, 26].

Fig. 4.2.**d** compares the aligned Cα length of the structural models built by Cryo2Struct and Phenix, which was computed using US-align. The aligned length is the number of Cα atoms denoting residues that have been successfully matched or aligned between the predicted model and its true structure. The average length of the true structures for all the 128 test maps is 3794.95. The average aligned length of Cryo2Struct's models is 945.55 (about 24.9% of the length of the known structure on average), 2.6 times the average length 358.51 of Phenix (about 9.4% of the length of the known structure on average). On 120 out of 128 density maps, Cryo2Struct has a larger aligned length than Phenix. Another interesting phenomena is that the models constructed by Cryo2Struct always have the same or very similar number of residues as the corresponding true structures (supplementary Fig. **S1.a**) and therefore capture the overall shape of the true protein structure well despite of some errors in the local regions and atom connections, while the models constructed by Phenix usually are much smaller than the true structures (supplementary Fig. **S1.b**) and therefore only renders a portion of the true structures.

In addition to using US-align to compare the models with the known structures, we also used the phenix.chain_comparison tool to compare a model and the true structure to compute the percentage of matching Cα atoms, as shown in Fig. 4.2**e**. It calculates the Cα match score, the percentage of Cα atoms (residues) in the model that have corresponding residues within a 3 Å distance in the true structure. It also reports the sequence match score, i.e., the percentage of the matched residues that have the same amino acid type (identity) as their counterparts in the true structure. The models built by Cryo2Struct have an average Cα match score of 43%, higher than Phenix's 41.2%. The average sequence match score is 13.4% for both Cryo2Struct and Phenix. It is worth noting that the Cα match score measures the match precision of Cα atoms in a model without considering the Cα atom coverage of the model. For instance, a partial model may have a high Cα match score but can only cover a small portion of its corresponding true structure. Because Cryo2Struct tends to build much more complete models than Phenix, their difference in terms of the Cα match score is less pronounced than in terms of the other metrics.

To remedy the shortcoming of the Cα match score calculated by the phenix.chain_comparison tool, we introduce a new Cα quality score considering both the Cα match precision and Cα coverage, which is the product of the Cα match score and the total number of predicted residues of a model obtained from the Phenix.chain_comparison tool divided by the number of the residues in the true structure, as described in Equation 4.1. It is in the range [0, 1]. A higher score signifies a more accurate and complete structural model. Fig. 4.2**f** compares the Cα quality scores of the structures modeled by Cryo2Struct and Phenix. The average Cα quality score for Cryo2Struct is 0.43, substantially higher than 0.23 of Phenix. On 125 out of 128 maps, Cryo2Struct has a higher Cα quality score than Phenix. The result shows that Cryo2Struct is capable of building structural models with higher average coverage and Cα matching score than Phenix.

$$\textbf{C}\alpha\ \textbf{Quality} = \frac{C\alpha \text{ Match} \times \#\text{ of modeled residues}}{\#\text{ of residues in the true protein structure}} \tag{4.1}$$

Fig. 4.2**g-i** illustrates such an example (EMD ID: 8767). The true structure for the map (Fig. 4.2**g**) has 2,934 residues. The model built by Cryo2Struct (Fig. 4.2**h**) has 2,932 residues, about 7.5 times 392 residues of the model built by Phenix (Fig. 4.2.**i**) that is very fragmented, while the Cα match score and sequence match score of the former (i.e., 53.4% and 15%) are only 26-29% higher than 42.3% and 11.6% of the latter. In contrast, the Cα quality score of the Cryo2Struct constructed model is 0.54, 9 times 0.06 of the Phenix model, more accurately reflecting the difference in the quality of the two models.

Finally, we analyzed how the performance of the two methods change with respect to the resolution of the cryo-EM density maps. Fig. 4.3**a-c** plot the F1 scores, global normalized TM-scores, and Cα quality scores of the models built by the two methods against the resolution of the cryo-EM density maps measured in Angstrom (Å), respectively. In terms of each of the three scoring metric, as expected, the accuracy of models built by the two methods decreases as the value of the resolution of the cryo-EM density maps increases (i.e., the resolution gets worse). The linear regression line for Cryo2Struct models is above that for Phenix, indicating that for the maps of the same resolution, the average score of the models built by Cryo2Struct is higher than that of Phenix. Moreover, the gap between the two increases as the value of resolution gets larger. This indicates that the quality of the models built by Phenix decreases faster than Cryo2Struct as the resolution of the cryo-EM density maps gets worse, i.e., Cryo2Struct is more robust against (or less sensitive to) the change of the resolution of density maps than Phenix. This is reflected by the less steep negative slope of the regression line for Cryo2Struct than that of Phenix and the less negative correlation between the

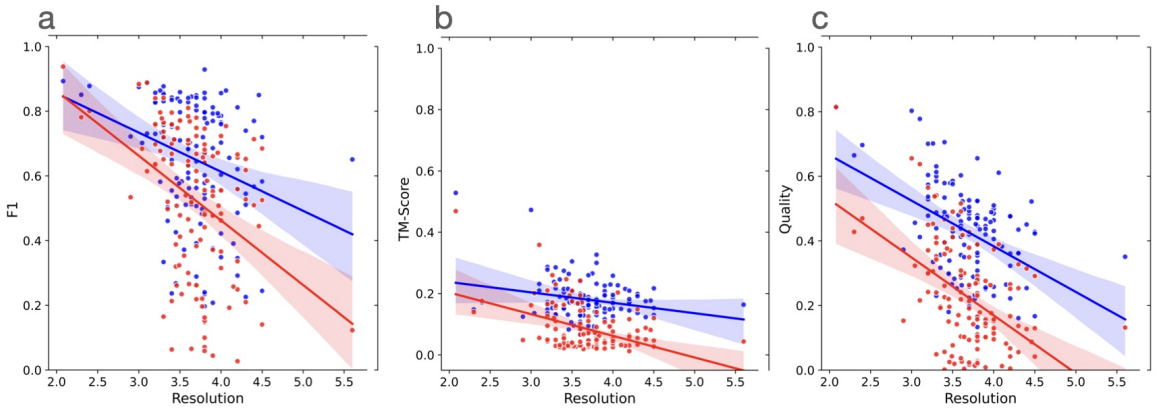


Figure 4.3: The plots of the scores (F1 score, global normalized TM-score, and Cα quality score) of the models built by Cryo2Struct and Phenix against the resolution of the 128 cryo-EM density maps. Blue dots denote Cryo2Struct constructed models and red dots the Phenix models. The solid lines depict linear regression lines, and the colored area represents a 95% confidence interval. The confidence interval is narrower (i.e., the linear estimation is more certain) in the resolution range [3Å- 4.5Å] where there are more data points. (**a**) F1 score against resolution. The equation of the regression line for Cryo2Struct (blue) is $y = -0.1209x + 1.0966$, while for Phenix (red), it is $y = -0.1998x + 1.2618$. The correlation between F1 score of Cryo2Struct and the resolution is -0.28, while for Phenix, it is -0.40. (**b**) The normalized global TM-score against resolution. The equation of the regression line for Cryo2Struct is $y = -0.0339x + 0.3057$, while for Phenix, it is $-0.0706x + 0.3447$. The correlation for Cryo2Struct is -0.24, while for Phenix, it is -0.43. (**c**) Cα quality score against resolution. The equation of the regression line for Cryo2Struct is $-14.1318x + 94.8512$, while for Phenix, it is $-17.9190x + 88.6207$. The correlation for Cryo2Struct is -0.43, while for Phenix it is -0.49.

score of Cryo2Struct and the resolution of the cryo-EM density maps than Phenix's. For instance, the Pearson correlation coefficient between the F1 score of Cryo2Struct and the resolution is -0.28, weaker than -0.40 of Phenix. This observation is consistent in terms of all three metrics, indicating that that Cryo2Struct generally builds better models from cryo-EM density maps than Phenix and therefore can be used to improve the quality of the models built from both the existing cryo-EM density maps in the Electron Microscopy Data Bank (EMDB) and the new ones to be generated.

Additionally, to assess the computational efficiency of Phenix and Cryo2Struct, we modeled atomic structures using both methods. Table 4.1 presents the compute time required for each. The evaluations were conducted on the same computing system, equipped with 192 CPU cores and 1 TB of memory. Phenix required 129 hours to model a structure with 2,395 residues, whereas Cryo2Struct completed the modeling of a significantly larger structure with 8,828 residues in just 9 hours. This demonstrates the efficiency of Cryo2Struct, which not only scales effectively to larger structures but also significantly reduces the computational time required for atomic structure modeling.

Method	EMD-ID	Residues	Weight (kDa)	Unique Protein Chains	Time (hrs)
Phenix	9565	2,395	297.39	1	129
Cryo2Struct	**40492**	**8,828**	**1,013.55**	**10**	**9**

Table 4.1: Compute time evaluation for Cryo2Struct and Phenix.

4.3.5 Evaluating Cryo2Struct on a Large New Dataset

We further evaluated the performance of Cryo2Struct on a large independent test dataset of 500 new maps with resolutions ranging from 1.9 Å to 4.0 Å. The average resolution of the density maps is 2.88 Å. These maps, released after April 2023, do not exist in the training and validation data in Cryo2StructData [1] that contains the cryo-EM density maps released before April 2023. The number of residues in the 500 maps ranges from 234 to 8,828.

On the new dataset, the average recall, F1 score, global normalized TM-score, Cα quality score, Cα sequence match score, and Cα match score of Cryo2Struct are 70%, 70%, 0.22, 0.50, 20.1%, and 49.5%, respectively, higher than 65%, 66%, 0.2, 0.43, 13.4%, and 43% on the standard test dataset, suggesting that the average quality of the cryo-EM density maps in the new dataset is higher than the standard dataset, which is consistent with the fact that the new cryo-EM density maps has the average resolution of 2.88 Å better than the average resolution of 3.68 Å of the old density maps in the standard test dataset. The relatively high recall, F1 score, Cα quality score, and Cα match score show that Cryo2Struct performs very well in identifying individual Cα atoms, while the relatively

lower global normalized TM-score and Cα sequence match score indicates it is still very challenging to build correct connected models that cover and match most regions of a large protein structure and its sequence.

Fig. 4.4**a-f** illustrate the relationship between each of the six scores (the recall, F1 score, global normalized TM-score, Cα quality score, Cα sequence match score, and Cα match score) of the models and the resolution of the density maps. In terms of each metric, there is a negative relationship between the metric and the resolution, i.e., the quality of model decreases as the resolution value of cryo-EM density map increases (i.e., the resolution gets worse) as observed on the standard test dataset. The Pearson correlation coefficient (PCC) for the recall, F1 score, global normalized TM-score, Cα quality score, Cα sequence match, and Cα match scores with respect to the resolution are -0.201, -0.202, -0.11, -0.298, -0.234, and -0.299, respectively, indicating that the negative relationship is rather weak and Cryo2Struct is robust against the deterioration of the resolution of cryo-EM density maps. Fig 4.4.**g** illustrates a high-quality model for a very large protein complex (EMD ID: 16963). The model has 6,316 residues and high quality scores (Cα quality score = 0.73, TM-score = 0.43, Cα match score = 72.8%, sequence match score = 50.6%, and F1 score = 90.5%). Furthermore, Fig 4.5 shows several additional good modeling examples, demonstrating that Cryo2Struct is capable of modeling some large structures with good overall accuracy.

Moreover, because only some regions of the models built by Cryo2Struct can be aligned with the true structures, we specifically analyzed the quality of the local regions of the models that can be aligned with the true structures by US-align in terms of aligned Cα length and RMSD (root mean squared distance) of the aligned regions. Supplementary Fig. **S2** plot RMSD of the aligned regions against their lengths for all the models built for the density maps in the new test dataset. The average length of the aligned regions of the models is 532.51 residues, accounting for 29% of the average length of true structures (i.e., 1837.43 residues). And the average RMSD of the aligned regions is 1.6Å. The results show that Cryo2Struct can build a significant portion of the protein structures with very high accuracy (low RMSD). And the RMSD decreases (i.e., the accuracy increases) with respect to the length of the aligned regions, according to the weak Pearson's correlation of -0.134 between the RMSD and the length of aligned regions. It is interesting to observe that Cryo2Struct can build high-accuracy models of large aligned regions up to thousands of residues.

Finally, we investigated how the global quality of the models changes with respect to the length (number of the residues) of the known structures (i.e., the size of the proteins) (supplementary Fig. **S3**). Unlike their similar relationship with the resolution of the cryo-EM density maps, the six metrics (recall, F1 score, global normalized TM-score, Cα quality score, Cα sequence match score,

and Cα match score) exhibit different relationship with the size of the proteins. The Cα recall and F1 score have a weak positive correlation (i.e., 0.259 and 0.258 respectively) with the size of proteins indicating that it is slightly easier to recognize individual Cα atoms for larger protein structures, while there is a weak negative correlation (i.e., -0.214) between the global TM-score and the size of proteins indicating it is slightly more difficult to build accurate full-length models for larger proteins. And the correlation for Cα quality score, Cα sequence match score, and Cα match score with respect to the size of proteins is almost 0, indicating that these scores are largely independent of the size of proteins.

4.3.6 Evaluating Cryo2Struct on Highly Sequence-Dissimilar Proteins

To investigate how well Cryo2Struct can generalize to proteins that are highly dissimilar to the proteins in the training and validation dataset, we used MMseqs2 [91] to compare the proteins in the standard test dataset and the the new test dataset with those in the training and validation dataset and removed any protein in each of them that contains one or more chains having more than 25% sequence identity with any chain of any protein in the training and validation dataset. The stringent 25% sequence identity is a threshold also utilized by DeepMainmast [92] in preparing a non-redundant test dataset. After the filtering, 22 out of 128 cryo-EM density maps in the standard test dataset are left to form a redundancy-reduced standard test dataset. The resolution of the density maps in the redundancy-reduced standard test dataset ranges 2.08 Å to 5.6 Å and has an average resolution of 3.72 Å and the number of residues in the maps ranges from 448 to 7,440. Likewise, 169 out of 500 cryo-EM density maps in the new test dataset are left to form a redundancy-reduced new test dataset. The resolution of the density maps in the redundancy-reduced new test dataset ranges 1.93 Å to 3.9 Å and has an average resolution of 2.89 Å and the number of residues in the maps ranges from 234 to 7,248.

Supplementary Fig. **S5** compares Cryo2Struct with Phenix on the redundancy-reduced standard test dataset in terms of three metrics: recall, F1, and quality score. Cryo2Struct performs better across the board, with 21 out of 22 structures having higher recall and quality scores and 16 out of 22 structures having higher F1 scores. The average recall score of Cryo2Struct is 67.8%, much higher than 40.7% of Phenix. Similarly, the F1 score for Cryo2Struct is 68%, higher than 51% of Phenix. The average quality score for Cryo2Struct is 0.51, much higher than 0.27 for Phenix.

Moreover, Cryo2Struct has an average sequence match of 18.2%, higher than 14.2% of Phenix. The Cα match for Cryo2Struct is 50.9%, higher than 48.5% of Phenix. The average length of

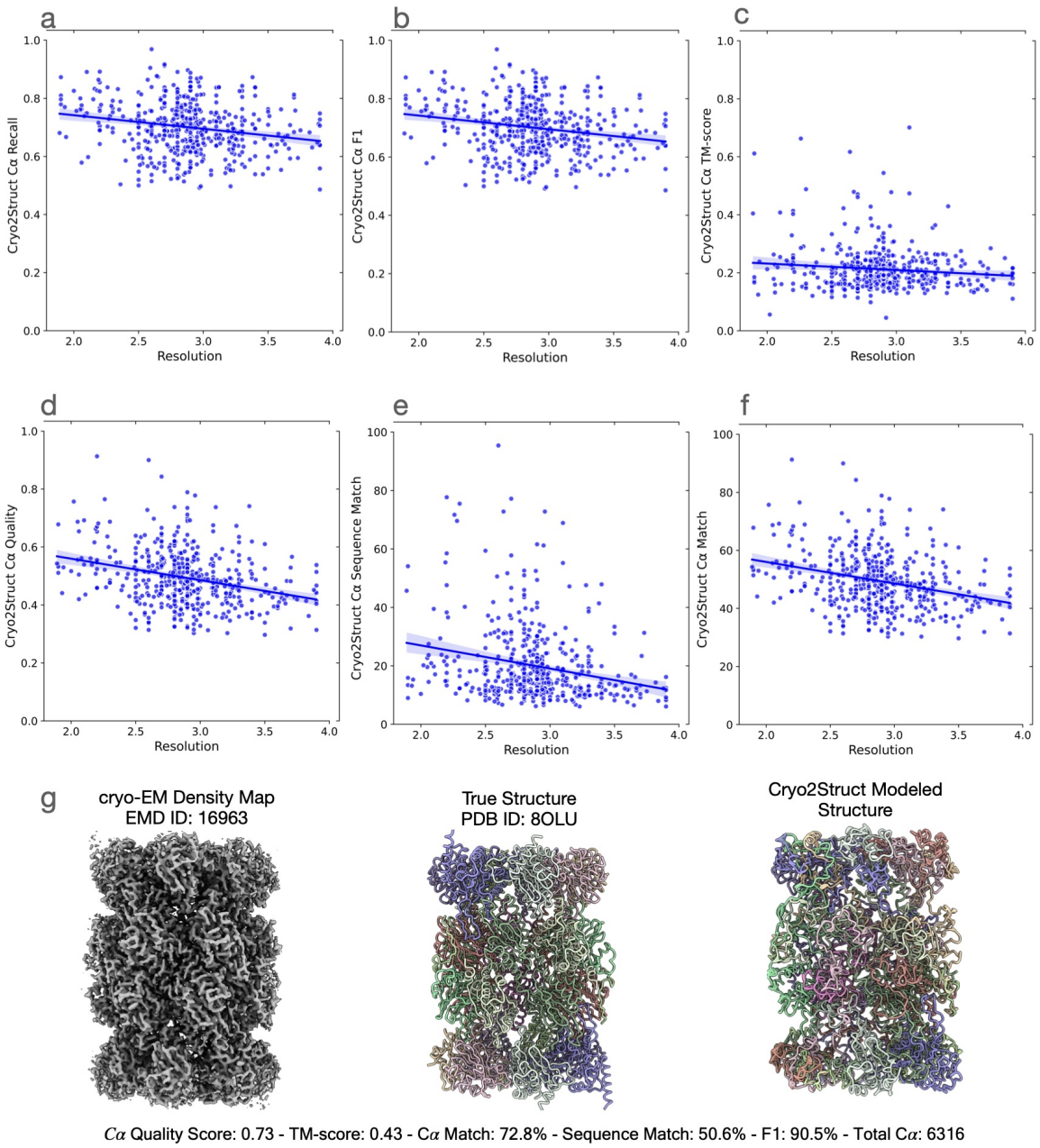


Figure 4.4: The quality of atomic models built for 500 test cryo-EM maps. The solid lines depict linear regression lines, and the colored area represents a 95% confidence interval. (**a**) The Cα recall versus resolution; the regression equation: $-0.0466x + 0.8350$; Pearson's correlation: -0.201. (**b**) The F1 score versus resolution; the regression equation: $-0.0468x + 0.8357$; the correlation: -0.202. (**c**) The normalized TM-score versus resolution; the regression equation: $-0.0222x + 0.2762$; the correlation: -0.11. (**d**) The Cα quality score versus resolution; the regression equation: $-0.0741x + 0.7080$; the correlation: -0.298. (**e**) The Cα sequence match score versus resolution; the regression equation: $-7.9226x + 42.8422$; the correlation: -0.234. (**f**) The Cα match score versus resolution; the regression equation: $-7.4408x + 70.8924$; the correlation: -0.299. (**g**) A modeling example. One on the left is the density map (EMD ID: 16963), in the middle is the true structure (PDB ID: 8OLU), and on the right is the model built by Cryo2Struct. The structure is a hetero 28-mer with a stoichiometry of A2B2C2D2E2F2G2H2I2J2K2L2M2N2 and a weight of 848.37 kDa. The total number of modeled Cα atoms is 6,316. Source data are provided as a Source Data file.

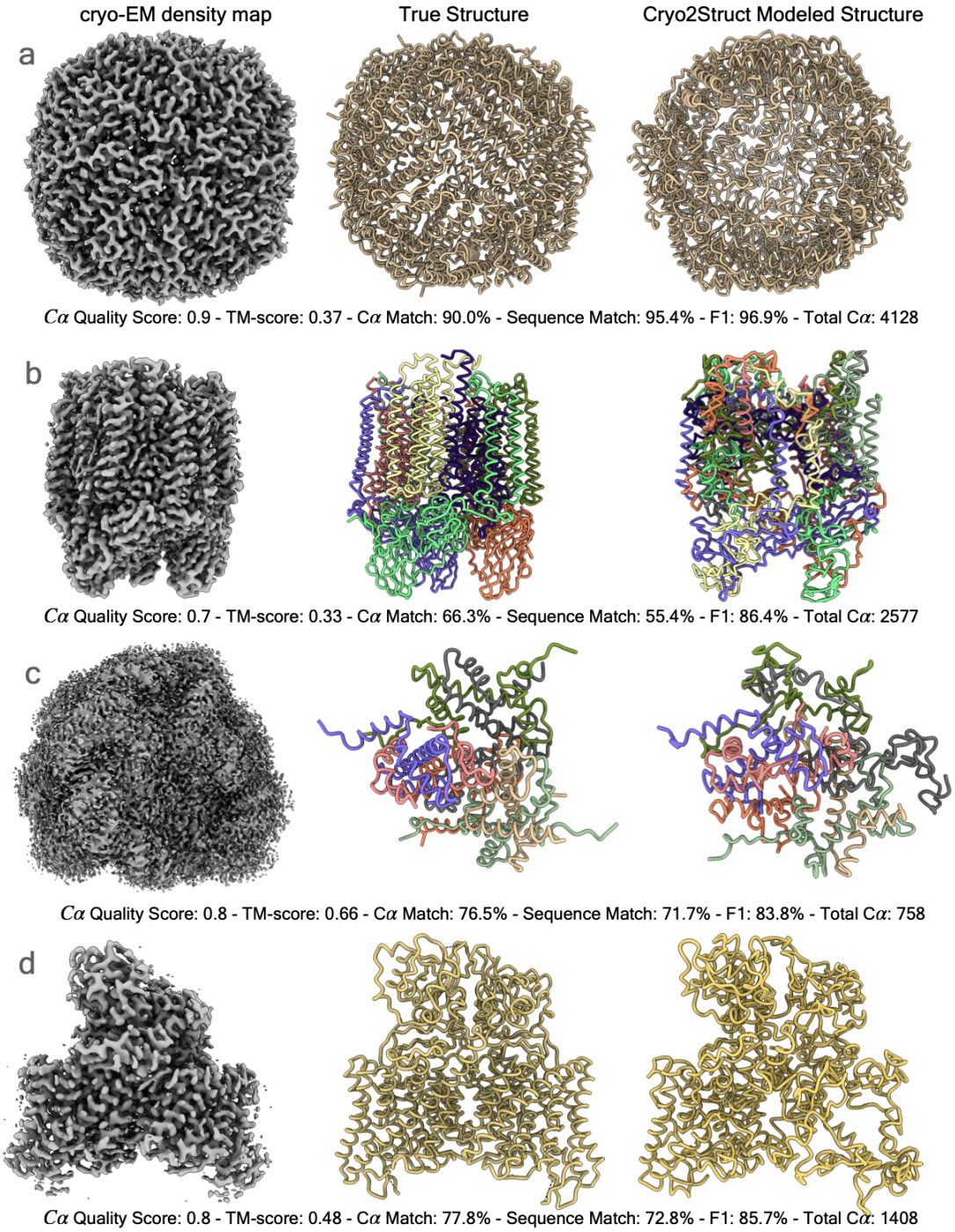


Figure 4.5: The high-quality models built for four test cryo-EM maps. In each panel from left to right are the cryo-EM density map, the true structure, and the model built by Cryo2Struct. The chains in both the true structure and the model are colored with distinct colors. The total Cα number shown in each panel is the total number of residues in a model. (**a**) The result for EMD ID: 17961 (PDB ID: 8PVC, released on 2023-11-29, and resolution of 2.6 Å). (**b**) The result for EMD ID: 17287 (PDB ID: 8OYI, released on 2023-11-08, and resolution of 2.2 Å. (**c**) The result for EMD ID: 37070 (PDB ID: 8KB5, released on 2023-10-18, and resolution of 2.26 Å). (**d**) The result for EMD ID: 35299 (PDB ID: 8IAB, released on 2023-08-02, resolution of 2.96 Å). Source data are provided as a Source Data file.

Cryo2Struct predicted structures is 3,224.5, close to the average length of the true structures (i.e. 3,237.31) and more than double the average length of Phenix predicted structures (i.e., 1,580.8). The aligned length for Cryo2Struct predicted structures is 955.8, much higher than 514.0 of Phenix. The global normalized TM-score for Cryo2Struct is 0.22, much higher than 0.13 of Phenix. The results demonstrate that Cryo2Struct substantially out perform Phenix on the test proteins that are highly dissimilar to the training and validation proteins.

Furthermore, Cryo2Struct's average recall, F1, global normalized TM-score, Cα quality score, Cα sequence match score, and Cα match score on the redundancy-reduced standard test dataset are 67.8%, 68%, 0.22, 0.51, 18.2%, and 51%, respectively, very similar to its scores of 65%, 66%, 0.2, 0.43, 13.4%, 43% on the standard test dataset without the redundancy reduction, indicating that Cryo2Struct's performance is independent of sequence similarity and it generalizes well to test proteins that have little or no sequence similarity with the proteins in the training and validation dataset.

Supplementary Fig. **S6** plots the recall, F1, and quality scores of Cryo2Struct predicted structures against the resolution of the density maps in the redundancy-reduced new test dataset. Similarly as on the new test dataset, the scores slightly trends down as the resolution gets worse. Cryo2Struct's average recall, F1, global normalized TM-score, Cα quality score, Cα sequence match score, and Cα match score on the redundancy-reduced new test dataset are 70.4%, 70.5%, 0.22, 0.51, 21.2%, and 50.5%, respectively, very similar to its scores of 70%, 70%, 0.22, 0.50, 20.1%, and 49.5% on the new test dataset without the redundancy reduction. This is same as observed on the redundancy-reduced standard test dataset and the standard test dataset, further confirming that Cryo2Struct's performance generalizes well to new proteins that are highly dissimilar to the proteins in the training and validation data. The decoupling of Cryo2Struct's performance and sequence similarity is probably because its transformer model only uses electron density values in cryo-EM density maps to predict Cα atoms and their amino acid types without using protein sequence information at all.

4.3.7 Confidence Scores by Cryo2Struct

Cryo2Struct provides a per-residue estimation of confidence within the range of [0, 1] for both Cα and amino acid type predictions, i.e., the chance (probability) that the predicted Cα or amino acid type is correct. Like the pLDDT scores that AlphaFold [34] assigns to predicted structures, the confidence scores reflect the degree of confidence Cryo2Struct has in predicted Cα atoms and their amino acid types, with higher scores indicating more reliable predictions, while lower scores suggest

more uncertainty that warrants scrutiny of the predictions.

Specifically, the confidence score for a predicted Cα atom is estimated by one logistic regression classifier ($P(y = 1|x) = \frac{1}{1+e^{-(\beta_0+\beta_1 \cdot x)}}$; β_0 and β_1: weights to be optimized), which utilizes the probability of the Cα atom predicted by the deep learning model as input (x) to assess its probability of correctness ($P(y = 1|x)$). Similarly, the confidence score for a predicted amino acid type is estimated by another logistic regression classifier using the emission probability of the amino acid type from the HMM assigned by the Viterbi algorithm, the probability of the Cα atom predicted by the deep learning model, and the one-hot encoding of the amino acid type as input (x) to predict the correctness of the amino acid type prediction.

To generate the binary labels (1: correct and 0: incorrect) to train the logistic regression classifiers, we utilized Phenix.chain_comparison to match a Cryo2Struct modeled structure with the corresponding true structure. We assigned a label of 1 to the Cα atoms in the Cryo2Struct modeled structure that have matching residues in the true structure, otherwise a label of 0. For the matched Cα atoms, we further assigned a label of 1 to the amino acid types matched with those in the true structure, and 0 otherwise.

We trained the two logistic regression classifiers on the 325 Cryo2Struct modeled structures of 325 targets in the new test dataset and tested them on the separate subset of 167 Cryo2Struct modeled structures that have less than 25% sequence identity with the Cryo2Struct training dataset, which is the same subset used in the section titled: Evaluating Cryo2Struct on highly sequence-dissimilar proteins.

To assess how well the confidence scores generated by the logistic regression can measure the quality of the 167 test structures built by Cryo2Struct, we correlated them with the true Cα and sequence match scores of the structures computed by phenix.chain_chain_comparison with respect to the true structures. The overall Cα confidence score for a Cryo2Struct modeled structure is the average of the confidence scores of all its Cα atoms, and the overall amino acid type confidence scores for a Cryo2Struct modeled structure is the average of the confidence scores of the amino acid types of all its residues. The Pearson's correlation coefficient between the Cα match score and the Cα confidence score for the 167 test structural models is 0.6, with a corresponding p-value of 3.06E-17. This correlation value suggests a strong positive linear relationship between the Cα match score and the Cα confidence score. The low p-value indicates that the observed correlation is statistically significant. Similarly, the correlation between the sequence match score and the overall amino acid confidence type score for the 167 test structural models is 0.7, with a p-value of 1.72E-24, confirming that the latter is a good indicator of the former.

Furthermore, there is a strong relationship between the per-residue Cα atom confidence scores and amino acid type confidence scores on the 167 test structural models built by Cryo2Struct. The correlation coefficient between them is 0.96, with a low p-value of 1.14E-92. Supplementary Fig. **S7** show an example plotting the confidence scores of Cα atoms against the confidence scores of amino acid types for a test protein, indicating a robust positive connection between the two kinds of confidence scores. Supplementary Fig. **S8** shows the two test examples of visualizing amino acid type confidence scores on top of the Cryo2Struct modeled structures using a color spectrum. Supplementary Fig. **S9** uses a detailed test example to demonstrate how different confidence scores can help users to identify high/local quality regions in the structural model in comparison with the true structure.

4.3.8 Refinement of Modeled Structures

In the sections above, we performed a comparison between the Cryo2Struct modeled structures and the structures generated by the Phenix.map_to_model tool. Phenix.map_to_model employs an integrated procedure that combines various independent modeling methods with an extensive real-space refinement technique [29, 93] to generate the structures. The models computed by Phenix.map_to_model benefit from the refinement through the Phenix.real_space_refine tool, which ensures their geometric integrity by resolving torsion angle outliers and rotamer outliers.

To investigate if the real-space refinement technique can further improve the Cryo2Struct modeled structures, we applied the same Phenix.real_space_refine tool employed by Phenix.map_to_model to refine the initial models built by Cryo2Struct. On the new test dataset, the average Cα match score of the refined models is 56%, 6.5% higher than that of the initial models, and the average RMSD of the refined models is 1.4 Å, 0.2 Å lower than that of the initial models. Similarly, on the standard test dataset, we observed an improvement in the average Cα match score by 8.9% along with a decrease of 0.2 Å in the RMSD, yielding an average Cα match score of 51.8% and an average RMSD of 1.62 Å for the refined models, respectively. This provides the compelling evidence that refining the initial models generated by Cryo2Struct further improves their quality by rectifying some geometrical issues.

4.4 Discussion

De novo modeling of protein structure solely from density maps, without using structural templates, is an interesting and important issue because it establishes a lower bound on the amount of structural

information that can be extracted from density maps. We developed Cryo2Struct, a de novo AI modeling method based on the transformer and HMM for building atomic protein structural models from medium- and high-resolution cryo-EM maps alone. The modeling process is fully automated, requiring no human intervention and no input from external tools. Cryo2Struct can rather accurately identify individual Cα atoms in density maps and is robust against the decrease of the resolution of density maps. Moreover, Cryo2Struct achieved substantially better performance than the most widely used de novo modeling method - Phenix in terms of multiple evaluation metrics including Cα recall, F1 score, global normalized TM-score, aligned Cα length, Cα match score, Cα sequence match score, and Cα quality score. In general, it can build much more accurate and more complete protein structures from cryo-EM density maps than Phenix, therefore advancing the state of the art of ab initio modeling of protein structures on cryo-EM density maps and providing a useful means for the community to build better protein structural models from both existing cryo-EM density maps and new ones to be generated to support biomedical research.

However, even though Cryo2Struct can identify most Cα atoms correctly with high F1-score and build high-accurate atomic models for some regions of large protein structures with very low RMSD, building high-accurate models covering most regions of large protein structures from density maps alone remains very challenging, reflected in low global TM-score and Cα sequence match score of the models. Obtaining high global TM-score and Cα sequence match score requires most if not all individual Cα atoms not only being correctly identified but also being correctly linked as peptide chains and assigned with correct amino acid types, which is combinatorially more challenging than predicting individual Cα atoms. A prediction error for only a few Cα atoms caused by missing or noise values in cryo-EM density maps that are very common may drastically lower the TM-score and Cα sequence score of the models because only when a long continuous stretch of chains are correctly predicted, the high TM-score and Cα sequence match score can be obtained. However, experimentally generating cryo-EM density maps that contain high-resolution density values covering every residue of a protein structure is still very challenging.

We envision that the global TM-score and Cα sequence match score of the structural models built from cryo-EM density maps can be further improved from the following aspects. The first is to develop more sophisticated and robust AI methods to predict protein atoms and their amino acid types with higher sensitivity and specificity from cryo-EM density maps to help build more accurate and complete protein chains. The second is to use additional inputs such as protein sequence information and AlphaFold-predicted protein structures to complement missing information in cryo-EM density maps to obtain more accurate and complete predictions. The third is to leverage the symmetry of

multiple chain in protein complexes to more accurately predict Cα atoms and amino acid types and align protein sequences with the HMMs. The fourth is to generate more accurate and complete cryo-EM density maps in the first place for the AI methods to use, which is being done by the community and would automatically improve the performance of Cryo2Struct as seen on the new test dataset in this work.

In the future, we plan to further expand Cryo2Struct to integrate cryo-EM density maps, protein sequences, and AlphaFold-predicted structures with deep learning together to build more accurate and complete protein structures. As more and more high-quality cryo-EM maps are being deposited in EMDB [6], such tools for automatically modeling atomic structure from them can enable scientists to better leverage this valuable resource to advance biomedical research.

4.5 Methods

4.5.1 Structure Modeling Process

As illustrated in Fig. 4.1, Cryo2Struct tackles the problem of building 3D atomic structural models from 3D cryo-EM density map in the following three main steps:

a) Predict Cα voxels and their amino acid types in the cryo-EM density map of a protein using a deep learning method based on transformer.

b) Construct a HMM model (λ) with predicted Cα voxels as hidden states and with emission and transition probability parameters set according to their predicted probabilities and their pairwise distance.

c) Align the amino acid sequence (i.e., $O = O_1, O_2, O_3 \dots O_T$, where $O_t \in V$; V: the set of 20 standard amino acids) with the HMM model λ to find the most likely Cα state sequence (path) ($X = x_1, x_2, x_3, \dots x_T$ where $x_t \in S$; S: the set of Cα hidden states) of generating the sequence to form the backbone structure of the protein.

4.5.2 Predicting Cα Voxels and Amino Acid Types

We designed a transformer-based model (Fig. 4.6), inspired by U-Net Transformers (UNETR) [94], for voxel classification in cryo-EM density maps. The model follows the contracting-expanding pattern of U-Net [43], utilizing a series of transformer-based encoders to extract features at multiple layers. The features extracted from different layers are utilized by a CNN-based decoder using skip

connections to classify the voxels into different classes. One model is trained to classify voxels into four different classes (Cα, C, N, and the absence of an atom) (atom type classification). Another model is trained to classify voxels into 21 classes, representing 20 amino acid types and an absent or unknown amino acid type.

Deep Learning Architecture

The deep learning model (Fig. 4.6) takes in an input sub-grid of cryo-EM density map represented as a 4D tensor with dimensions $H \times W \times D \times C$, where H is the height, W is the width, D is the depth, and C is the number of channels (C=1 for the input), denoted as $x \in \mathbb{R}^{H \times W \times D \times C}$. x is then divided into a series of flattened, uniform non-overlapping patches ($x_v \in \mathbb{R}^{N \times (P^3.C)}$), where P denotes the patch dimensions and $N = (H \times W \times D)/P^3$ is the number of the patches. The series of the patches are projected by a 3D convolution layer into a K-dimensional embedding space. A 1D learnable positional encoding $\mathbf{E}_{pos} \in \mathbb{R}^{N \times K}$ is then added to the projected patches, which subsequently serve as the input to the transformer encoder. Here, P is set to 16 and the embedding dimension (K) to 768.

Cryo2Struct uses an encoder of 12 blocks [63] each consisting of a normalization layer, [95], a multi-head attention layer, a normalization layer and a multi-layer perceptron to generate features for the input series of patches. The features from four different blocks: z_i (i.e., i $\in$ { 3, 6, 9, 12 }), with size $\frac{H \times W \times D}{P^3} \times K$ are reshaped into $\frac{H}{P} \times \frac{W}{P} \times \frac{D}{P} \times K$, respectively. The features of the four blocks and the original input are processed by deconvolution and/or convolution layers and concatenated together in a U-Net fashion step by step to generate the final feature tensor of the same dimension as the original input (see Fig. 4.6 for details), which is used by a $1 \times 1 \times 1$ convolution layer to classify each voxel.

Training and Validation

We used the Cryo2StructData [1] dataset, which includes maps with the resolution in the range [1.0 Å - 4.0 Å] , to train and validate the two transformer models. The cryo-EM density maps in the dataset were released till 27 March 2023. The dataset is split according to a 90% to 10% ratio into the training and validation datasets. The total dataset has $7,392$ cryo-EM density maps. The training dataset and validation dataset has $6,652$, and 740 cryo-EM density maps, respectively. The atom types and amino types of the voxels in the density maps are labeled.

The training was performed on sub-grids (dimension: $32 \times 32 \times 32$) of the density maps, utilizing a batch size of 720, the NADAM optimizer [96] with a learning rate of 1e-4, and a dropout rate of

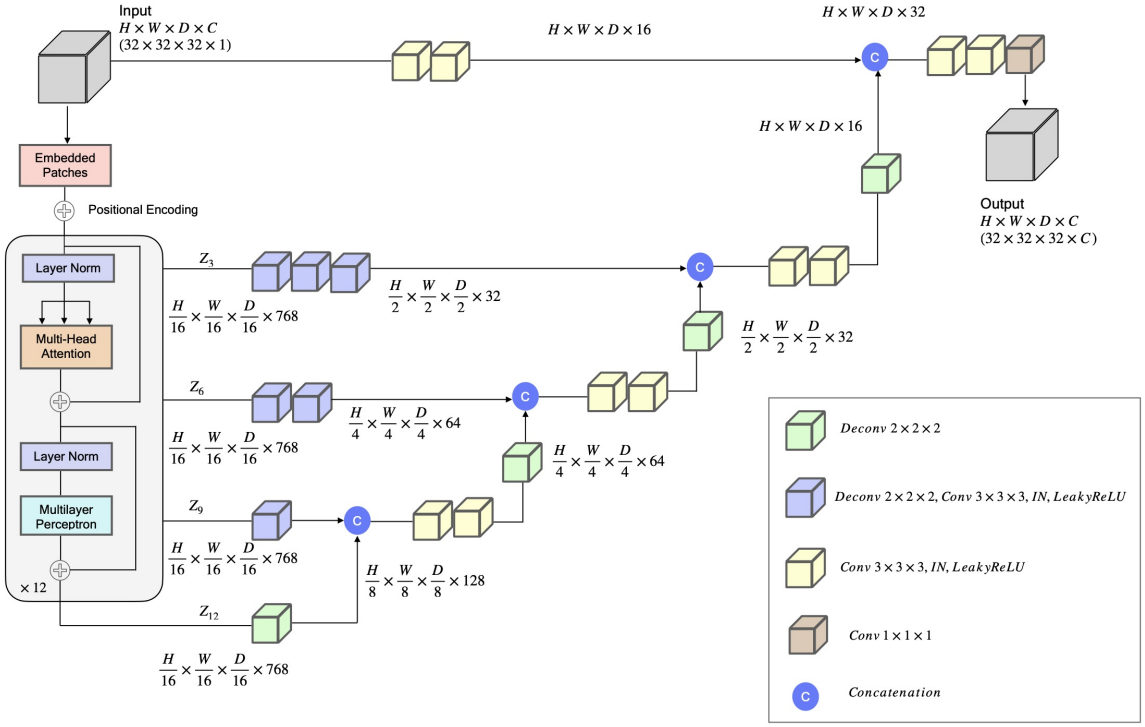


Figure 4.6: The Deep Learning architecture for backbone atom and amino acid type classification. The network takes a $32 \times 32 \times 32$ sub-grid of cryo-EM density map as an input with one channel representing the density value of voxels. The input is divided into a series of patches. The patches are projected into an embedding space by a 3D convolution layer, and then is added with a positional encoding. The patches are then processed by an encoder, comprising 12 identical blocks each with a normalization layer, a multi-head self-attention layer, a normalization layer, and a multi-layer perceptron (MLP). The encoded features of blocks 3, 6, 9 and 12 denoted as (z_3, z_6, z_9, z_{12}) and the original input are integrated into the decoders via skip connections in a U-Net fashion, each of which includes convolution and deconvolution layers with instance normalization (IN), Leaky ReLU activation, and feature concatenation. The last hidden features are used by a 1 x 1 x 1 convolution layer to generate the final 3D sub-grid output of the same size as the input, i.e., $32 \times 32 \times 32$, with (C) output channels (i.e., 4 for the backbone atom type classification (Cα, N, C and the absence of an atom) and 21 for the amino acid type classification (20 standard amino acids and no/unknown amino acid). The amino acid-type classification model has 92.281893 million parameters, whereas the atom type classification model has 92.281604 million parameters.

0.1. We used a distributed data parallel (DDP) technique to train the models on 24 compute nodes each equipped with 6 NVIDIA V100 32GB-memory GPUs in the Summit supercomputer [81].

The deep learning models were trained with the weighted cross entropy loss function described in Equation 4.2 to handle the class imbalance problem.

$$\mathcal{L}(x, y) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C} w_c \cdot y_{n,c} \cdot \log\left(\frac{\exp(x_{n,c})}{\sum_{i=1}^{C}\exp(x_{n,i})}\right) \tag{4.2}$$

where, $\mathcal{L}(x, y)$ represents the weighted cross-entropy loss. N is the number of samples in the minibatch. C is the number of classes. w_c is the weight for class c computed using Formula 4.3. $x_{n,c}$ is the logit for class c in sample n, and $y_{n,c}$ is a binary indicator (0 or 1) of whether class c is the correct classification for sample n. ω_c in Formula 4.3 represents the weight assigned to class c, n_c is the number of samples in class c, and $\sum_{k=0}^{\text{classes}} n_k$ is the total number of samples across all classes.

$$\omega_c = 1 - \frac{n_c}{\sum_{k=0}^{\text{classes}} n_k} \tag{4.3}$$

Throughout the training process, we monitored both training and validation loss along with the F1 score, known for its effectiveness in handling class-imbalanced data as it represents the harmonic mean of precision and recall. We implemented and trained the deep learning models using PyTorch Lightning [97], version 1.7.3. The evaluation metrics (F1, Recall, and Precision) were computed using TorchMetrics [98], version 0.9.3. We tracked the model's performance on both training and validation data using the Weights and Biases tool. If the validation loss did not improve for five consecutive epochs, we reduced the learning rate by a factor of 0.1. We saved the top 5 trained models with lowest validation loss during the training and selected the model with the highest F1 score on the validation dataset as the final trained model.

Cα Voxel Clustering

When applying the trained transformer to a density map to predict Cα voxels, it is common that multiple spatially close voxels corresponding to the same Cα atom are predicted as Cα atoms. To remove redundancy, Cryo2Struct employs a clustering strategy to group predicted Cα voxels within a 2Å radius into clusters. The average Cα probability and the amino acid type probability of Cα voxels in each cluster are computed. The centrally located Cα voxel in each cluster and the average

probabilities of the cluster are used to represent the Cα atom of the cluster, while the other Cα voxels in the same cluster are removed.

4.5.3 Connecting Cα Atoms into Protein Chains and Assigning Amino Acids to Cα atoms

Connecting predicted Cα voxels into chains and accurately assigning their amino acid types is a challenging task. We designed an innovative Hidden Markov Model (HMM) whose hidden states represent predicted Cα voxels to accomplish it seamlessly in a single step, which is used by a customized Viterbi algorithm to align the sequence of a target protein with the HMM. The hidden states (Cα voxels) aligned with the sequence are joined together to form the backbone of the protein, in which the amino acid type of each Cα voxel is set to the type of the amino acid aligned with it. Cα voxels with a probability higher than 0.4 are selected as the hidden states for the HMM.

The HMM uses K hidden states to represent predicted K Cα voxels. Let's denote individual Cα hidden states in the HMM as $S = S_1, S_2, S_3, \ldots, S_K$ and individual symbols (amino acid types) as $V = V_1, V_2, V_3, \ldots, V_N$, where N is equal to the number of standard amino acids (i.e., 20) generated from the hidden states. The hidden states in the HMM are fully connected, where there is a direct transition from any state to any other state, as depicted in supplementary Fig. **S4.a**. The transition probabilities between Cα hidden states are stored in the transition matrix, denoted as γ with a size of $K \times K$. The emission probabilities of generating observation symbols from the hidden states are stored in the emission matrix, denoted as δ, with a size of $K \times N$. The initial state distribution is denoted as $\Pi =< \pi_1, \pi_2, \pi_3, \ldots, \pi_K >$, where π_i is the probability that the HMM starts from state i. A hidden path may start from and end at any state. We use a compact notation, $\lambda = (\gamma, \delta, \Pi)$, to represent the HMM.

Hidden Markov Model Construction

The transition probability matrix (γ) is constructed based on the distance between two predicted Cα states (voxels) in the 3D space, calculated from their coordinates using Formula 4.4. The distance x is converted into a probability using the modified Gaussian probability density function (PDF) in Equation 4.5 ($f(x)$), with a mean (μ) of 3.8047 Å and a standard deviation (σ) of 0.036 Å. Both μ and σ were estimated from the distances between two adjacent Cα atoms in the true protein structures in the training dataset. Additionally, we introduce a fine-tune able scaling factor (Λ) that multiplies with (σ) to make the model adjustable. We set (Λ) to 10. The transition probabilities

from one state to all other states are normalized by dividing each of them by their sum.

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \tag{4.4}$$

$$f(x) = \frac{1}{\Lambda\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2(\Lambda\sigma)^2} \tag{4.5}$$

The emission probability matrix (δ) for each Cα state (voxel) is calculated from both its predicted amino acid type probability and the background (prior) probability of 20 amino acids in the nature. Specifically, the geometric mean of the two is calculated as $\sqrt{a \times b}$, where a corresponds to the predicted probability for each amino acid type, and b represents the background frequency of the amino acid type, as shown in supplementary Fig. **S10**, that was precomputed from the true protein structures in the training dataset. The geometric means for 20 amino acid types are normalized by their sum as their final emission probability. An example of emission matrix is shown in supplementary Fig. **S4.b**.

The initial probability for a Cα state (π_i) is the probability that it generates the first amino acid of the protein sequence normalized by the sum of these probabilities of all the Cα states.

Aligning Protein Sequence with HMM using a Customized Viterbi Algorithm

The customized Viterbi algorithm is used to find the most likely path in the HMM to generate a protein sequence with the maximum probability. The only difference between the customized Viterbi algorithm and the standard Viterbi algorithm is that the former allows a hidden state occurs at most once in the aligned hidden state path while the latter does not have such a restriction. The restriction is needed because one hidden state denoting a Cα voxel can only be aligned to (occupied by) one amino acid in a protein sequence. The details of the algorithm is depicted in Algorithm below, generating a path $X = x_1, x_2, x_3, \ldots, x_T$, which is a sequence of states $x_t \in S$ aligned with a protein sequence (the observation O). For a multi-chain protein complex, the sequence of each chain is aligned with the HMM one by one. Once a chain is aligned, the states in the hidden path aligned with it are removed from the HMM before another chain is aligned. In the alignment process, it is ensured that any Cα state occurs at most once in one hidden state path. One distinct strength of this HMM-based alignment approach is that every amino acid of the protein is assigned to a Cα position as long as the number of the predicted Cα voxels is greater than or equal to the number of the amino acids of the protein, which is usually the case when the 0.4 probability threshold is used to select predicted Cα atoms to construct the HMM. This is the reason that Cryo2Struct builds

complete structural models from density maps.

Input:

- O: Amino acid sequence of observations
- S: Set of hidden states
- Π: Initial state probabilities
- γ: State transition probabilities
- δ: Observation likelihood probabilities (emission probability)

Initialization:

a) For each state i (from 1 to K):

- Set $T_1[i, 1] = \Pi_i \cdot \delta_{i,O_1}$
- Set $T_2[i, 1] = 0$

b) Initialize an empty set of visited states:

$$\text{visited\_states} = \{\}$$

Recursion:

a) For each observation j (from 2 to T):

(a) For each state i (from 1 to K):

- If $i \notin$ visited_states:
  - Compute the path score:

$$\text{path} = T_1[i, j-1] \cdot \gamma_i \cdot \delta_{i,O_j}$$

  - Update:

$$T_1[i, j] = \max_k(\text{path})$$

$$T_2[i, j] = \arg\max_k(\text{path})$$

(b) Determine the most probable state at time j:

$$z_j = \arg\max_k(T_1[:, j])$$

(c) Add z_j to visited states:

$$\text{visited\_states} \leftarrow \text{visited\_states} \cup \{z_j\}$$

Backtracking:

a) Determine the final state:

$$z_T = \arg\max_k(T_1[:, T])$$

$$x_T = S_{z_T}$$

b) For each previous time step j (from $T-1$ to 1):

- Backtrack to find the previous state:

$$z_{j-1} = T_2[z_j, j]$$

- Assign the corresponding state:

$$x_{j-1} = S_{z_{j-1}}$$

Output: Return the sequence X of most likely states.

The customized Viterbi algorithm is a dynamic programming algorithm implemented in C++ to achieve high computational efficiency. The source code is compiled with a high level of optimization and is provided as a shared library, which is then linked with the Python program of constructing the HMM.

4.5.4 Inference and Testing

After Cryo2Struct was trained and validated, it was blindly tested on a standard test dataset of 128 density maps and a large new dataset of 500 density maps. For each test map, the Cryo2Struct inference process consisting of the deep learning prediction and the HMM alignment was executed on compute nodes each with a 40GB GPU, 150 GB RAM, and 64 CPU cores. The deep learning prediction was carried out on the GPU, whereas the HMM alignment was executed on the CPU cores. The model building for the largest map (EMD ID: 40492 with resolution 2.9 Å), involving 8,828 modeled residues, was completed in 9 hours on a compute node, while it took only 2.90 minutes to build a model for the smallest map (EMD ID: 36426 with resolution 3.3 Å) with 234 residues.

4.6 Data Availability

The dataset used to train and validate Cryo2Struct (Cryo2StructData) is available on the Harvard Dataverse [1], and the description of the data preparation and labeling process can be found in [88] and in the previous chapter of this dissertation. The detailed information about the test datasets including the EMD IDs of the density maps and the evaluation scores are provided in two Excel files (`Standard_test_data.xlsx` for the standard test dataset and `Cryo2Struct_test_data.xlsx` for the new test dataset) available at https://doi.org/10.7910/DVN/GQCTTD, and the true structures and the structural models built by Cryo2Struct and Phenix for the test density maps are also available at the same website. The two Excel files (`Standard_test_data.xlsx` for the standard test dataset and `Cryo2Struct_test_data.xlsx`) for the new test dataset) are also available in source data file. Additionally, two Excel files (`Standard_test_data_low_sim.xlsx` and `Cryo2Struct_test_data_low_sim.xlsx` for the redundancy-reduced standard test dataset and for the redundancy-reduced new test dataset, respectively) are available on Harvard Dataverse, accessible at https://doi.org/10.7910/DVN/GQCTTD.

4.7 Code Availability

The source code for Cryo2Struct is available in the GitHub repository: https://github.com/jianlin-cheng/Cryo2Struct. This repository also includes instructions on running Cryo2Struct on cryo-EM maps to generate 3D atomic protein structures. Furthermore, to keep the codes of Cryo2StructData permanent, we published all code and instructions required to reproduce the results on Zenodo, an online research sharing platform with a permanent Digital Object Identifier number [99].

4.8 Supplementary

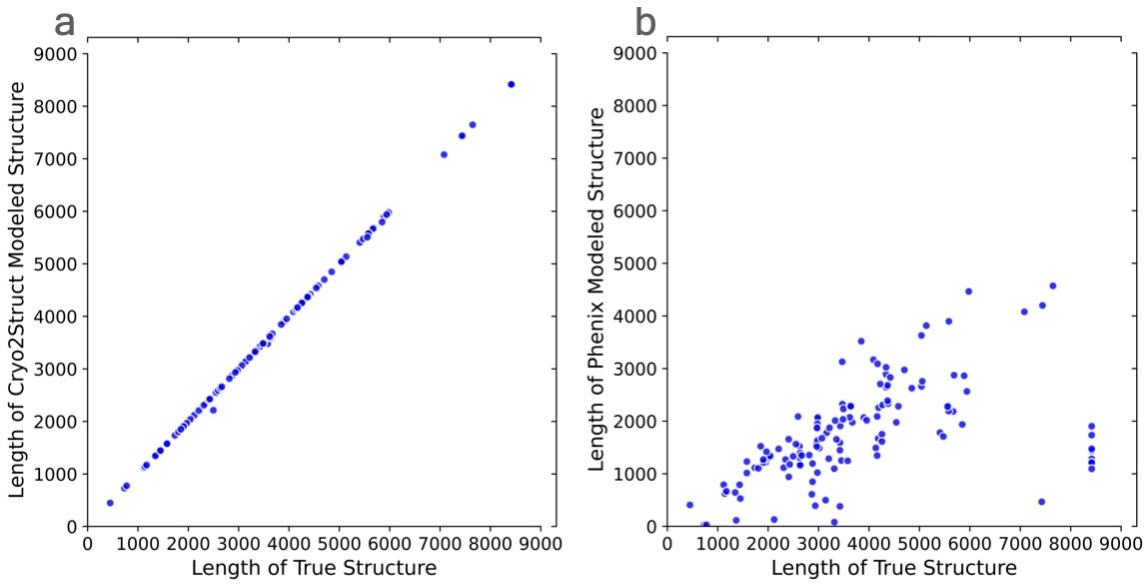


Figure S1: **Length of structural models built by Cryo2Struct and Phenix versus (VS) length of the true structures in the standard test dataset.** (**a**) Cryo2Struct models VS true structures. (**b**) Phenix models VS true structures.

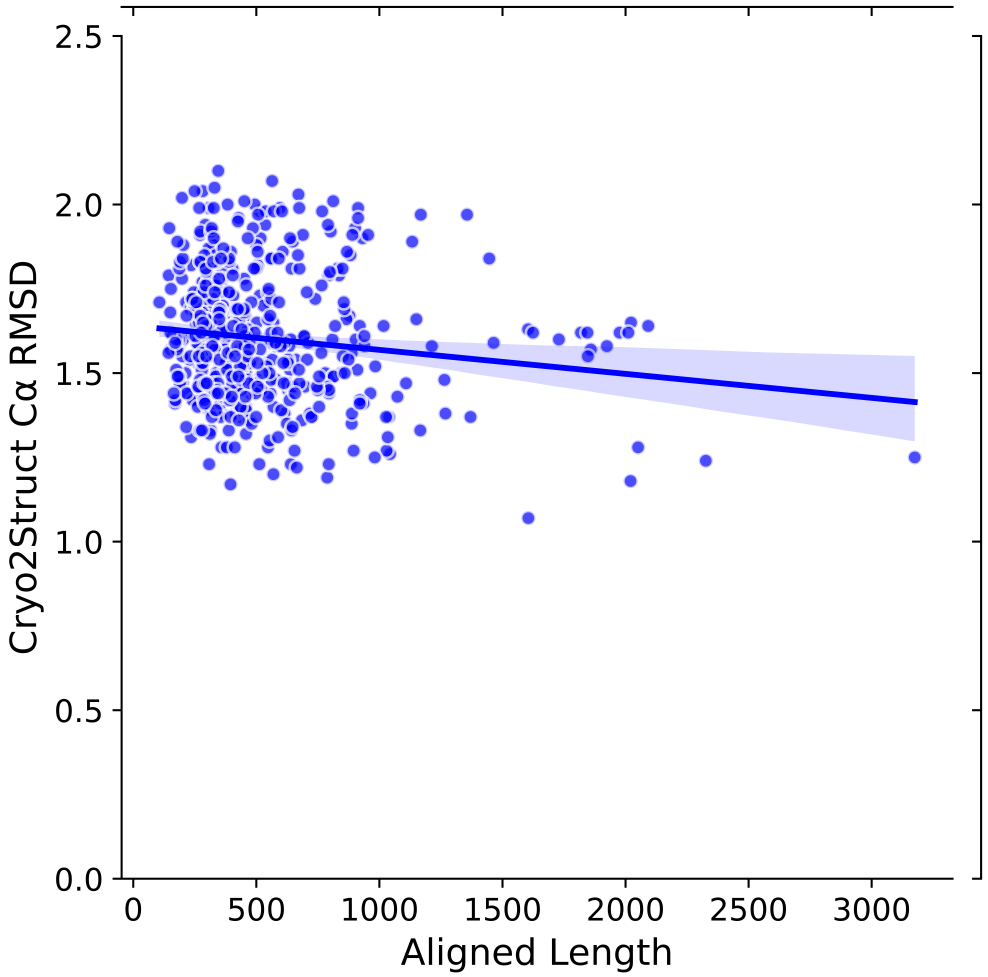


Figure S2: **RMSD versus the length of the aligned regions of the atomic models built for 500 test cryo-EM maps.** The models were aligned with the true structures by US-align. The solid line depicts linear regression line, and the colored area represents a 95% confidence interval. The regression equation: $y = -0.0001x + 1.6401$; the correlation: -0.134. The average RMSD of the models is 1.60 Å. The average aligned length is 532.51 where as the average length of true structure is 1837.43. Cryo2Struct models have about 29% aligned length.

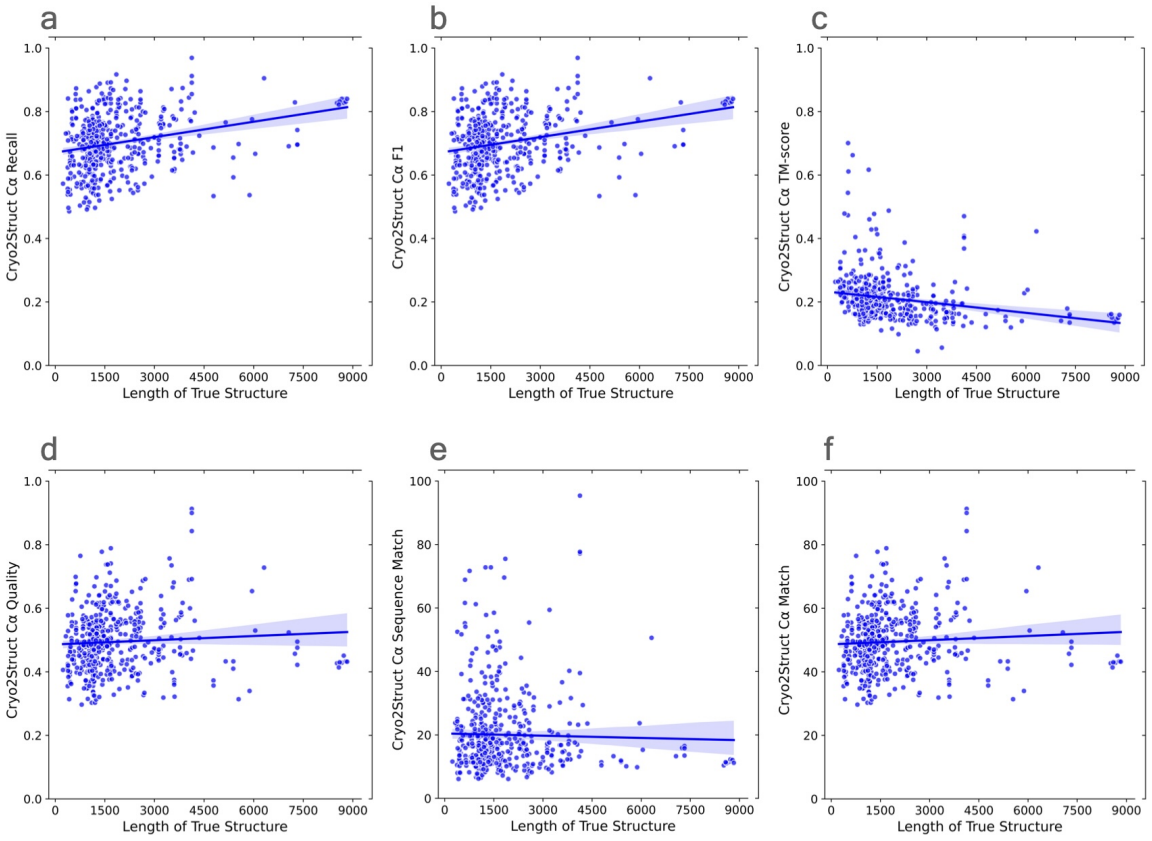


Figure S3: **The quality scores of atomic models built for the 500 cryo-EM maps in the new test dataset versus (VS) the length of the true structures.** The solid lines depicts linear regression lines, and the colored area represents a 95% confidence interval. (**a**) The Cα recall VS length of true structure; the regression equation: $0.0000x + 0.6712$; Pearson's correlation: 0.259. (**b**) The F1 score VS length of true structure; the regression equation: $0.0000x + 0.6714$; the correlation: 0.258. (**c**) The normalized TM-score VS length of true structure; the regression equation: $-0.0000x + 0.2328$; the correlation: -0.214. (**d**) The Cα quality score VS length of true structure; the regression equation: $0.0000x + 0.4863$; the correlation: 0.066. (**e**) The Cα sequence match score VS length of true structure; the regression equation: $-0.0002x + 20.4579$; the correlation: -0.025. (**f**) The Cα match score VS length of true structure; the regression equation: $0.0004x + 48.6615$; the correlation: 0.065.

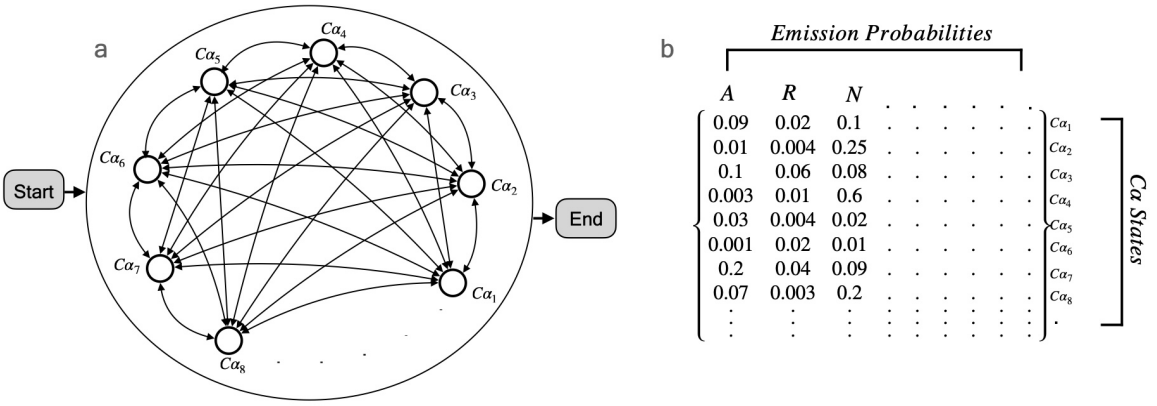


Figure S4: **A Hidden Markov Model (HMM) used for aligning protein sequences with predicted Cα atoms (voxels) to generate protein backbone traces. (a)** The states of the fully connected HMM. A hidden path can start from or end at any Cα state. It is worth noting that there is no gap state in the HMM and therefore every amino acid in a protein sequence can be aligned to one Cα atom. **(b)** The emission probabilities of the hidden Cα states are the normalized geometric mean of the predicted amino acid type probability and the background (prior) probability for 20 amino acids in the nature, referred to by their abbreviation.

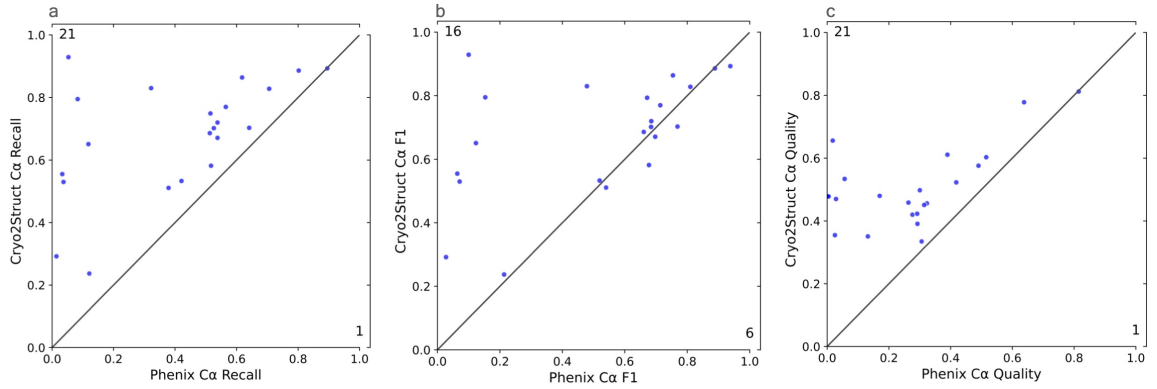


Figure S5: **The comparative analysis of atomic models built for 22 cryo-EM density maps in the redundancy-reduced standard test dataset by Cryo2Struct and Phenix in terms of recall, F1 score, and quality score of Cα atoms.** The proteins of the density maps have $\leq 25\%$ sequence identity with the protein in the training and validation datasets. In the panel of each evaluation metric, the score of the model built by Cryo2Struct for each map is plotted against that by Phenix for the same map. A dot above the 45 degree line indicates that Cryo2Struct has higher score than Phenix for the map. The number in the top-left corner represents the total number of maps on which Cryo2Struct has higher scores, while the number in the bottom-right corner denotes the total number of maps on which Phenix has higher scores. **(a)** The recall of Cα. **(b)** The F1 score of Cα. **(c)** The Cα quality score.

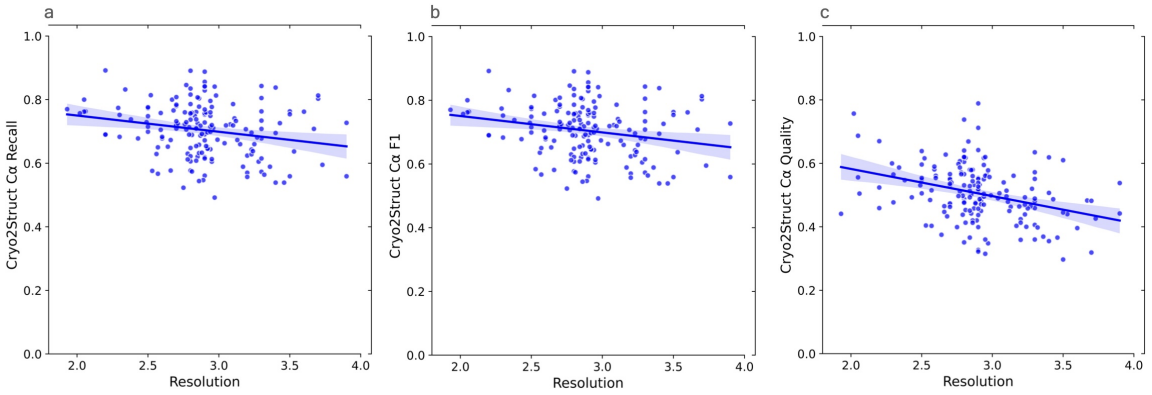


Figure S6: **The scores of atomic models built by Cryo2Struct for 169 test cryo-EM maps in the redundancy-reduced new test dataset plotted against the resolution of the maps.** The proteins of the density maps have $\leq 25\%$ sequence identity with the protein in the training and vaidation datasets. The solid lines depict linear regression lines, and the colored area represents a 95% confidence interval. (**a**) The Cα recall versus resolution; the regression equation: $-0.0511x + 0.8521$; Pearson's correlation: -0.217. (**b**) The F1 score versus resolution; the regression equation: $-0.0515x + 0.8536$; the correlation: -0.219. (**c**) The quality score versus resolution; the regression equation: $-0.0856x + 0.7537$; the correlation: -0.344.

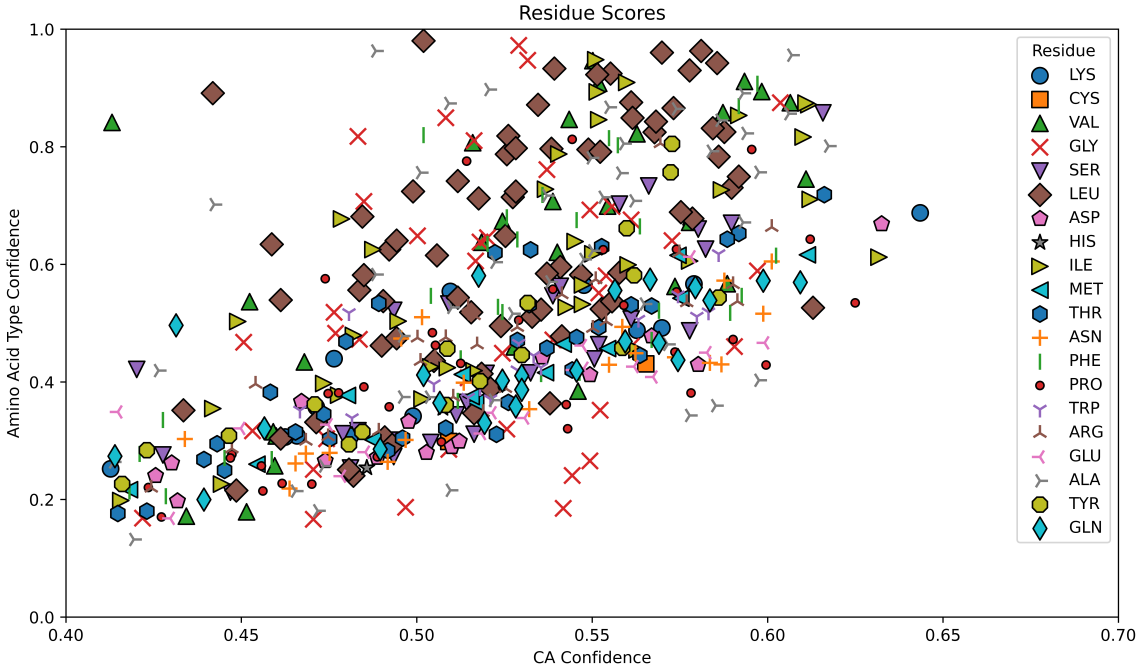


Figure S7: **The residue-wise confidence scores provided by Cryo2Struct pertaining to the modeled structure for the cryo-EM density map with the EMD ID: 15789 (PDB ID: 8B0N, released on 2023-07-12, and resolution of 2.67 Å).** The x-axis represents the confidence scores of predicted Cα atoms. The y-axis denotes the confidence scores associated with the amino acid types for the Cα atoms. The different shapes in the plot denote different amino acid types. The average Cα confidence score is 0.53, while the average confidence score for amino acid types is 0.511. The total number of modeled residues is 510. There is a clear positive correlation between the two kinds of confidence scores.

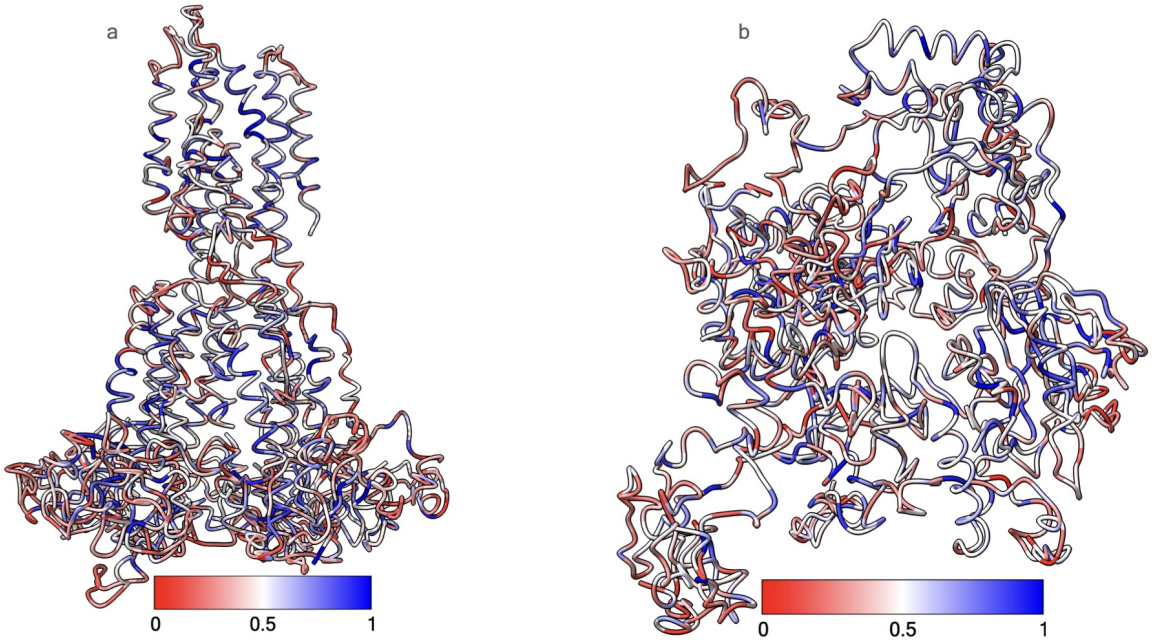


Figure S8: **The residue-wise amino acid type confidence scores mapped to the modeled structure and visualized using a color spectrum. (a)** Cryo2Struct modeled structure for the cryo-EM density map with the EMD ID: 41624 (PDB ID: 8TUL, released on 2023-09-13, resolution of 2.8 Å). **(b)** Cryo2Struct modeled structure for the cryo-EM density map with the EMD ID: 34402 (PDB ID: 8GZR, released on 2023-08-02, and resolution of 2.8 Å). Both **(a)** and **(b)** have less than 25% sequence identity with the proteins in the dataset used to train the deep learning model.

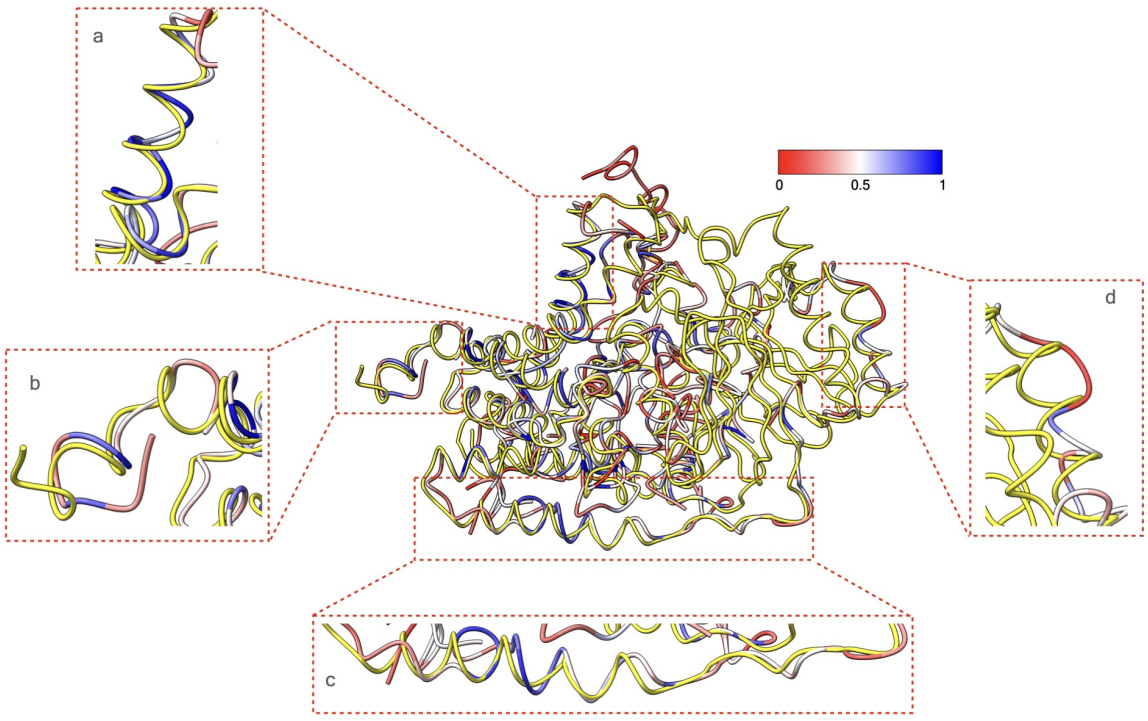


Figure S9: **An in-depth analysis of residue-wise amino acid type confidence scores, mapped onto the Cryo2Struct modeled structure and visualized through a color spectrum, for EMD ID: 15789.** The modeled structure has less than 25% sequence identity with the proteins in the dataset used to train the deep learning model. The known PDB structure (PDB ID: 8B0N) is depicted in yellow color. **(a)** A segment of the well modeled region with high confidence scores, particularly within helical motifs. **(b)** A mixed region of different quality exhibiting different confidence scores. **(c)** An extended segment of the modeled structure with varying confidence levels, ranging from high to low, compared to the known structure. **(d)** Low confidence scores are observed in the regions where the modeled helix substantially deviates from the known PDB structure, indicating uncertainty.

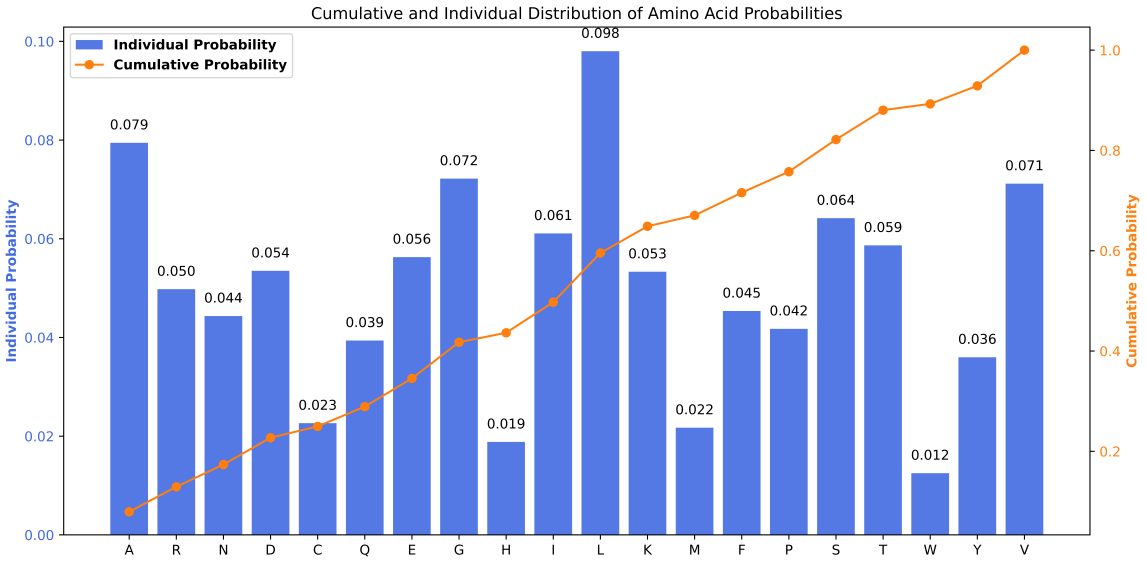


Figure S10: **The bar plot visualizing the individual probability (frequency) of each amino acid type in the training dataset.** Complementing this, the cumulative distribution function (CDF) is presented on the secondary y-axis (orange), elucidating the probability distribution of amino acid types in the dataset, summing up to 1. This visualization offers a comprehensive depiction of prior amino acid probabilities, which are combined with the probabilities of amino acid types predicted by Cryo2Struct to construct the emission probabilities of amino acid types in the HMM.

Chapter 5

ATOMIC STRUCTURE MODELING FROM CRYO-EM USING MULTI-MODAL DEEP LEARNING AND ALPHAFOLD3

5.1 Abstract

Cryo-electron microscopy (cryo-EM) has revolutionized structural biology by enabling near-atomic resolution visualization of protein structures. However, accurately modeling 3D atomic structures from cryo-EM density maps remains challenging, particularly for multi-chain complexes. Here, we present an automated pipeline that integrates multi-modal deep learning and advanced structure prediction techniques to improve model accuracy. Our approach leverages a deep learning model that combines sequence-based features from a Protein Language Model with cryo-EM density maps, enabling a richer feature representation across modalities. The deep learning-predicted voxels are utilized to build a Hidden Markov Model (HMM) and a tailored Viterbi algorithm is used to align sequences to generate an initial protein backbone structures. These backbone models then serve as templates for AlphaFold3, which refines the structures for improved accuracy. Our approach combines cryo-EM data with AlphaFold3 predictions, helping to refine and improve AlphaFold3's predicted structures. By integrating both methods, we can generate more accurate and reliable atomic models, particularly for proteins with complex conformations.

5.2 Introduction

Determining the three-dimensional (3D) atomic structures of macromolecules, such as protein complexes and assemblies, is important for understanding biological functions at the molecular level.

The spatial arrangement of atoms within these macromolecules provides key insights into protein structure, function, and interactions, guiding research in structural biology and rational drug discovery. In recent years, cryo-electron microscopy (cryo-EM) has emerged as a powerful experimental technique for visualizing large protein structures at near-atomic resolution. However, accurately building atomic structures from high-resolution cryo-EM density maps remains a challenging task due to missing electron values and local low resolution in some regions of cryo-EM maps which makes interpretation difficult.

Several computational approaches, including Phenix [29], DeepTracer [56], DeepMainmast [92], Cryo2Struct [72], and ModelAngelo [58], provide automated solutions for modeling atomic structures within cryo-EM density maps [72, 59]. Among them, Phenix [29] is a widely used tool that utilizes classical molecular optimization techniques to build atomic protein structures. DeepTracer [56] provides a deep learning-based web platform for automated structure building, while ModelAngelo [58] integrates cryo-EM data, amino acid sequences, and prior knowledge of protein geometries to refine structural models and assign residue identities. More recently, DeepMainmast [92] has combined AlphaFold2 [34] predictions with density-guided tracing protocols, significantly improving atomic model building from cryo-EM maps. These advancements have demonstrated that incorporating AlphaFold-predicted structures improves the reliability and accuracy of cryo-EM-based atomic structure determination.

Building on this momentum, we introduce a versatile workflow that integrates cryo-EM density data with AlphaFold3 [15] predictions, further refining and improving the accuracy of AlphaFold3-generated structures. By leveraging cryo-EM templates in the AlphaFold3 framework, our approach allows the generation of more accurate atomic models, particularly for proteins with flexible conformations and regions of low-resolution density. This integration improves the structure prediction capabilities, facilitating the modeling of complex biomolecular assemblies and expanding the potential applications of cryo-EM in structural biology and rational drug discovery.

5.3 Method

We developed a multi-modal deep learning model, called Cryo2Struct2, that integrates embeddings from the Protein Language Model (ESM) [16] with cryo-EM density maps. Cryo2Struct2 takes a 3D cryo-EM density map along with its associated amino acid sequence as inputs. To incorporate sequence information, we used the Evolutionary Scale Modeling (ESM) model [16], a pretrained language model with 3 billion parameters, to generate sequence embeddings of size 2560. These

embeddings were then incorporated into the density maps for training, validation and inference.

In our early work which is described in previous chapter, Cryo2Struct [72], we trained two separate models to predict atom types and amino acid types from cryo-EM density maps. In this work, for Cryo2Struct2, we designed a unified model with a shared transformer encoder to extract features from the density map, followed by task-specific decoders for atom and amino acid type predictions. The models were trained for volumetric segmentation tasks.

The predicted probabilities for atom and amino acid types from the deep learning model were then used to construct a Hidden Markov Model (HMM) [80], which was processed by a modified Viterbi algorithm, as introduced in Cryo2Struct [72], to align predicted voxels and build a 3D atomic protein backbone structure. We generated two atomic structures using this approach by adjusting parameters, particularly in selecting and clustering predicted Cα atoms. These atomic structures serve as template structures for AlphaFold3, encouraging it to predict protein structures that align with the cryo-EM density map while leveraging the state-of-the art prediction capabilities of AlphaFold3.

Our approach aims to integrate cryo-EM-based structure modeling with sequence-based structure prediction, employing a late fusion strategy where the structural models build from cryo-EM is incorporated as a template for AlphaFold3. This allows AlphaFold3 to refine the structure while maintaining consistency with experimental density data, ultimately improving the accuracy of atomic model predictions.

5.3.1 Model Architecture

The deep learning model is based on the 3D SegFormer [100] based architecture designed to integrate cryo-EM density map data with ESM embeddings for both amino-acid and atom-type prediction. The architecture consists of an encoder-decoder blocks, where the encoder extracts hierarchical representations of the cryo-EM density map, and the task-specific decoder separately predicts atom types and amino acid types in each voxel of cryo-EM density map.

The architecture of our deep learning module is shown in Figure 5.1. The encoder of the model consists of multiple Transformer [63] blocks designed to process volumetric cryo-EM density map. To fit the cryo-EM data within the memory constraints of commodity GPUs, the full 3D cryo-EM density map is divided into sub-cubes of size $32 \times 32 \times 32$, as also described in Cryo2StructData preparation chapter 3. The input density map, represented as a single-channel tensor of size $32 \times 32 \times 32 \times 1$, where the last dimension corresponds to the electron density value for each voxel in

cryo-EM density map, is processed through a series of convolution layers with varying kernel sizes and strides to generate multi-scale feature representations. We utilize the feed-forward multi layer perception (MLP) layer to transform the sequence embeddings generated from Protein Language Model (ESM) [16] from 2560 to the embedding dimension of the multi-scale feature representations and add those features to the multi-scale feature representations. This helps us to integrate the ESM embeddings with the features extracted from density map, ensuring that the sequence-level information is added in the model.

Each transformer block utilizes an efficient self-attention mechanism to efficiently capturing long-range patterns in the cryo-EM density maps. The standard self-attention mechanism [63] computes attention weights and has a computational complexity of $O(n^2)$, however with the efficient self-attention mechanism introduced in [101, 102], the computational complexity is reduced from $O(n^2)$ to $O(n^2/r)$. The reduction parameter r is set as 4,2,1,1 in the four stages of the encoder. The Transformer block also contains a 3D Mix FFN as described in [101] which allows for the automatic learning of positional cues and eliminates the need for fixed encoding for better performance. Finally, the encoder outputs four feature maps, (c_1, c_2, c_3, c_4) which are used as an inputs to the decoders.

We utilize two separate decoder blocks in the model, each designed for a specific prediction task, atom-type decoder and amino acid-type decoder. The atom-type decoder processes the features (c_1, c_2, c_3, c_4) from the encoder to predict atom types into four different classes, namely, Cα, N, C, and no presence of atoms. The amino acid-type decoder predicts amino acid types into 21 different classes by utilizing the same set of encoded features (c_1, c_2, c_3, c_4). The atom-type decoder directly predicts atomic labels from the encoded features, where as the amino acid type decoder benefits from additional features. Specifically, the amino acid type decoder incorporates the atom-type features generated from the atom-type decoder as an auxiliary feature, allowing the model to use atom-type information to improve amino acid classification accuracy, as shown in Figure 5.1.

Both atom and amino acid-type decoder follow a similar architecture, they first transform the multi-scale feature maps (c_1, c_2, c_3, c_4) from the encoder using MLP layers that project each feature layer into common embedding space. The transformed feature maps are then upsampled to a common spatial size before being fused through a convolutional layer. This fused representation undergoes dropout regularization, followed by a final convolutional layer that generates the class probability maps.

Finally, the predicted outputs from both decoders are upsampled by a factor of 4 using trilinear interpolation to get the original input size of $32 \times 32 \times 32 \times C$, where C represents output channels i.e., 4 for the backbone atom type classification (Cα, N, C, and the absence of an atom) and 21 for

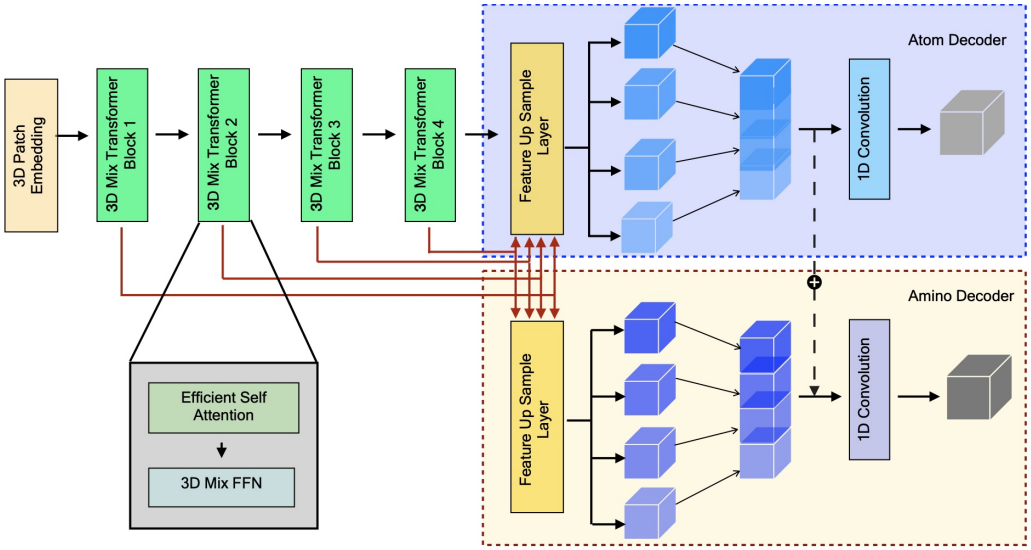


Figure 5.1: Model Overview: The architecture processes a sub-cube of a 3D cryo-EM density map using 3D Mix Transformer blocks to extract hierarchical spatial features. The Atom Decoder (blue background) classifies voxels into four atomic categories, while the Amino Decoder (yellow background) classifies voxels into 21 classes for amino acid prediction, using features from atom decoder.

the amino acid type classification (20 standard amino acids and no/unknown amino acid).

5.3.2 Training and Validation

We utilized the Cryo2StructData [88] dataset as described in chapter 3, which includes cryo-EM density map with the resolution in the range of [1.0-4.0 Å], to train and validate the deep learning model. The cryo-EM density maps in the dataset were released till 27 March 2023. The dataset is split according to a 90% and 10% ratio into training and validation datasets. The training dataset and validation dataset has 6652, and 740 cryo-EM density maps, respectively. Cryo2StructData [88] also provides the label maps for the cryo-EM density maps where every voxels in the density maps are labeled for both atom-type and amino acid types.

The model was trained on $32 \times 32 \times 32$ sub-grids extracted from full cryo-EM density maps, considering only sub-grids containing at least one nonzero voxel. Training was performed using a batch size of 1000 and optimized with the Adam optimizer, which integrates parameters from the shared encoder and both task-specific decoders. We employed a weighted cross-entropy loss function (Equation 5.1) to address class imbalance. The model was trained using four GPUs, each with 80 GB of memory, with a learning rate of 10^{-4}.

$$\mathcal{L}(x, y) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C} w_c \cdot y_{n,c} \cdot \log\left(\frac{\exp(x_{n,c})}{\sum_{i=1}^{C}\exp(x_{n,i})}\right) \tag{5.1}$$

where, $\mathcal{L}(x, y)$ represents the weighted cross-entropy loss. N is the number of samples in the minibatch. C is the number of classes. w_c is the weight for class c computed using Formula 5.2. $x_{n,c}$ is the logit for class c in sample n, and $y_{n,c}$ is a binary indicator (0 or 1) of whether class c is the correct classification for sample n. ω_c in Formula 5.2 represents the weight assigned to class c, n_c is the number of samples in class c, and $\sum_{k=0}^{\text{classes}} n_k$ is the total number of samples across all classes. The model with the lowest validation loss is selected as the trained model for inference stage.

$$\omega_c = 1 - \frac{n_c}{\sum_{k=0}^{\text{classes}} n_k} \tag{5.2}$$

5.3.3 Clustering predicted Cα voxels

The deep learning model predicts Cα atoms from cryo-EM density maps. Given the challenges of this prediction task, it is common for multiple spatially close voxels to be predicted as individual Cα atoms. To address this redundancy, we performed clustering of the predicted Cα atoms based on their proximity. Specifically, we applied two clustering thresholds: 2 Å and 3 Å. The 3 Å clustering threshold is more rigid, as the average distance between two Cα atoms is approximately 3.8 Å. By using a 3 Å threshold, the clustering helps to group voxels that are within 3 Å of each other, which assists in improving alignment and reducing incorrect connections between Cα atoms. The co-ordinate of the clustered Cα in each cluster, and the probabilities of the Cα amino acid-type for each cluster are averaged and are used for constructing the Hidden Markov Model (HMM).

5.3.4 HMM and Customized Viterbi

We utilized the same approach used by Cryo2Struct [72] as described in chapter 4 to align the protein amino acid sequence to the predicted and clustered Cα voxel co-ordinates. The transition probability for the HMM is constructed based on the distance between two predicted Cα voxels in the 3D space, calculated using the Euclidean distance formula. The distance is converted into a probability using the modified Gaussian probability density function (PDF) as shown in Equation 5.3 with a mean (μ) of 3.8047 Å and a standard deviation (σ) of 0.036 Å. Both μ and σ were estimated from the

distances between two adjacent Cα atoms in the known protein structures in the training dataset. Additionally, we introduce a fine-tune able scaling factor (Λ) that multiplies with (σ) to make the model adjustable. We set (Λ) to 10.

The emission probability matrix (δ) for each Cα state (voxel) is calculated from both its predicted amino acid type probability and the background (prior) probability of 20 amino acids in the nature. Specifically, the geometric mean of the two is calculated as $\sqrt{a \times b}$, where a corresponds to the predicted probability for each amino acid type, and b represents the background frequency of the amino acid type, that was precomputed from the true protein structures in the training dataset. The geometric means for 20 amino acid types are normalized by their sum as their final emission probability.

The initial probability for a Cα state is the probability that it generates the first amino acid of the protein sequence normalized by the sum of these probabilities of all the Cα states.

The customized Viterbi algorithm is used to find the most likely path in the HMM to generate a protein sequence with the maximum probability. For a multi-chain protein complex, the sequence of each chain is aligned with the HMM one by one. Once a chain is aligned, the states in the hidden path aligned with it are removed from the HMM before another chain is aligned. In the alignment process, it is ensured that any Cα state occurs at most once in one hidden state path. One distinct strength of this HMM-based alignment approach is that every amino acid of the protein is assigned to a Cα position as long as the number of the predicted Cα voxels is greater than or equal to the number of the amino acids of the protein.

$$f(x) = \frac{1}{\Lambda\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2(\Lambda\sigma)^2} \tag{5.3}$$

5.3.5 Templates for AlphaFold3

To improve the accuracy of structure prediction, we integrated template-based modeling into AlphaFold3 [15] using Cryo2Struct2-generated structures as templates. Our approach involves aligning query protein sequences with template sequences derived from cryo-EM-based structural predictions, in this case Cryo2Struct2. The alignment process extracts corresponding residue indices between the query and template sequences, ensuring a structurally meaningful template mapping. The extracted templates are then formatted into a JSON-compatible file for AlphaFold3, which includes multi-chain sequence entries, alignment indices, and embedded mmCIF template data. By incorporating these templates, AlphaFold3 can leverage prior structural information, improving alignment with

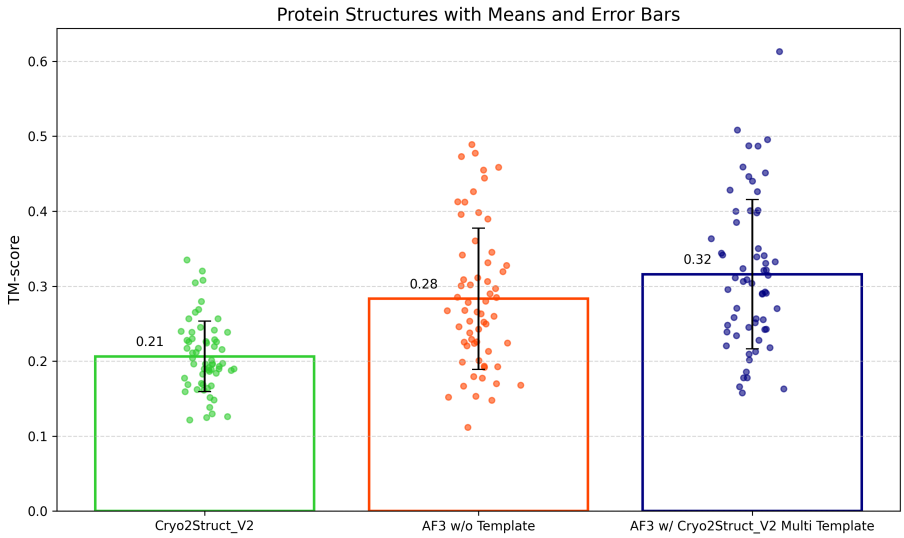


Figure 5.2: Evaluation of 61 protein structures modeled by each method using TM-Score. Each dot represents a structure predicted by the respective approach. The mean scores for structures modeled by Cryo2Struct2, AlphaFold3 without a template, and AlphaFold3 with multiple Cryo2Struct2 templates are 0.21, 0.28, and 0.32, respectively.

experimentally determined structures.

5.4 Results

The test dataset for atomic structure prediction was collected from the Electron Microscopy Data Bank (EMDB) [6] and consists of cryo-EM density maps deposited in the year 2024. These test data were not included in the training of either AlphaFold3 [15] or Cryo2StructData [88], ensuring an unbiased evaluation of model performance. The training data cut-off dates for AlphaFold3 [15] and Cryo2StructData [88] (chapter 3) were September 2021 and March 2023, respectively.

The test dataset consists of 61 cryo-EM density maps with an average resolution of 3.10 Å, ranging from 2.4 Å to 3.9 Å. The average number of residues per structure is 1,230.8, with a range of 374 to 3,245 residues. To assess structural similarity, we used the standard TM-score, which quantifies how well a predicted model aligns with its corresponding known structure. TM-scores were calculated using US-align [90], a protein complex structure comparison tool, with options enabled for aligning multi-chain oligomeric structures and all chains, as recommended for biological assembly alignment.

To ensure a fair comparison between models of varying lengths, the global TM-score was normalized using the length of the corresponding experimental structure. The TM-score ranges from 0 to 1, with 1 representing a exact structure match.

In terms of structural accuracy, as shown in Figure 5.2 the average TM-score for AlphaFold3

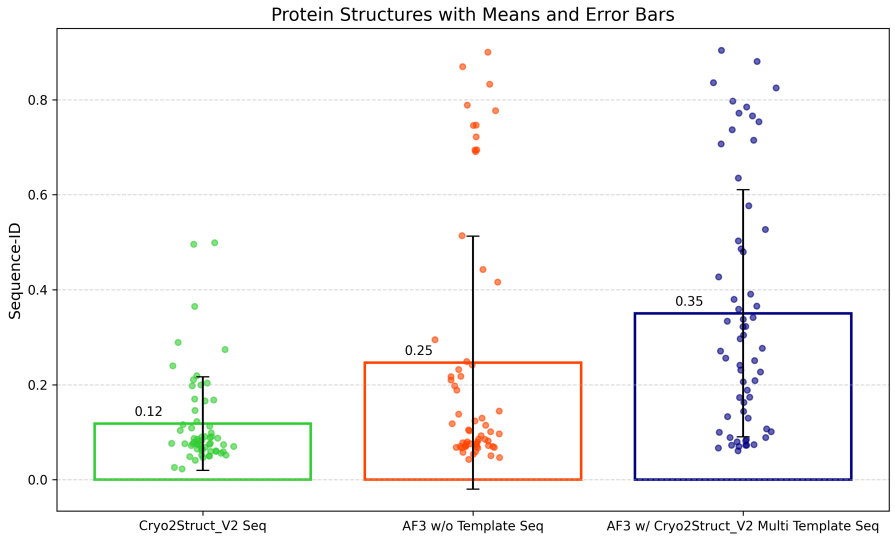


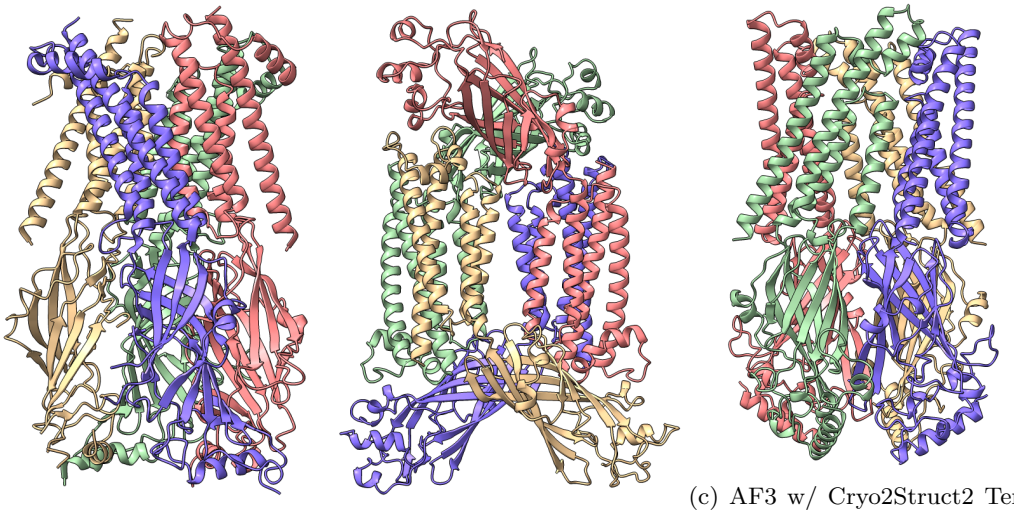
Figure 5.3: Evaluation of 61 protein structures modeled by each method using Sequence identity. Each dot represents a structure predicted by the respective approach. The mean scores for structures modeled by Cryo2Struct2, AlphaFold3 without a template, and AlphaFold3 with multiple Cryo2Struct2 templates are 0.12, 0.25, and 0.35, respectively.

using Cryo2Struct2 multi-template guidance is 0.32, whereas AlphaFold3 without template guidance achieves an average TM-score of 0.28. The average TM-score for structures modeled directly from Cryo2Struct2 alone is 0.21. These results indicate that integrating Cryo2Struct2-generated templates with AlphaFold3 improves structural accuracy, highlighting the benefit of incorporating cryo-EM-derived information for refining atomic structure predictions.

In terms of sequence accuracy, as shown in Figure 5.3, sequence identity is measured as the proportion of identical residues among the aligned residues. The average sequence identity (sequence-ID) for AlphaFold3 using Cryo2Struct2 multi-template guidance is 0.35, whereas AlphaFold3 without template guidance achieves an average sequence-ID of 0.25. For structures modeled directly from Cryo2Struct2 alone, the average sequence-ID is 0.12. These results indicate that incorporating Cryo2Struct2 templates into AlphaFold3 enhances sequence accuracy, further supporting the effectiveness of template-based refinement in cryo-EM-guided protein structure modeling.

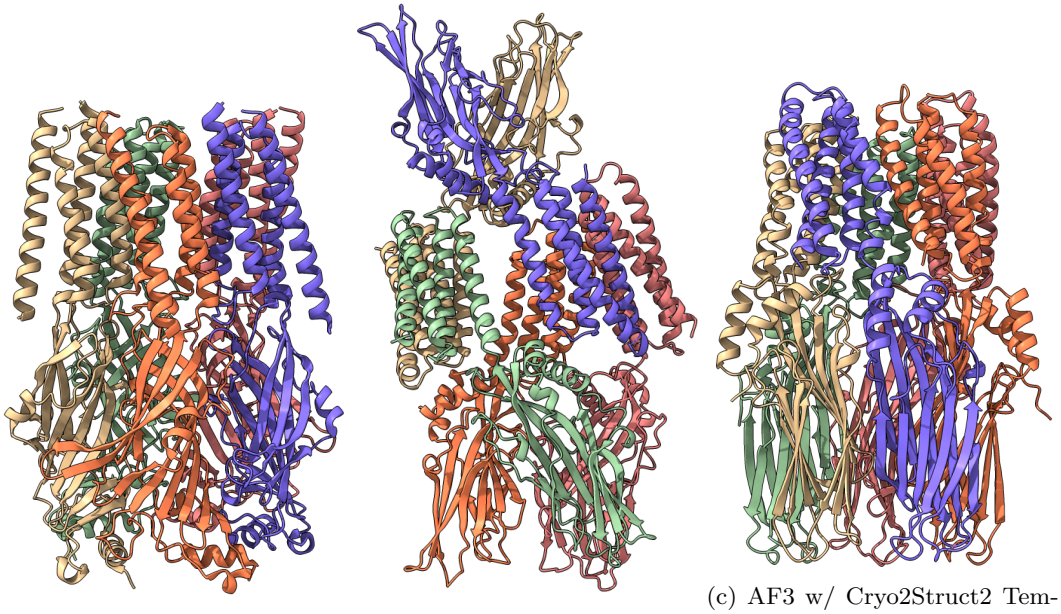
Figures 5.4, 5.5, and 5.6 showcase examples of modeled structures, providing detailed insights into their folds. Especially, in Figure 5.4, the structure predicted by AlphaFold3 without template guidance shows a lower match with the experimentally determined structure, with a TM-score of only 0.265. In contrast, when guided by Cryo2Struct2, the predicted structure aligns significantly better with the PDB-deposited structure, resulting in a higher TM-score of 0.398.

A similar trend is observed in Figure 5.5, where the AlphaFold3 prediction without template



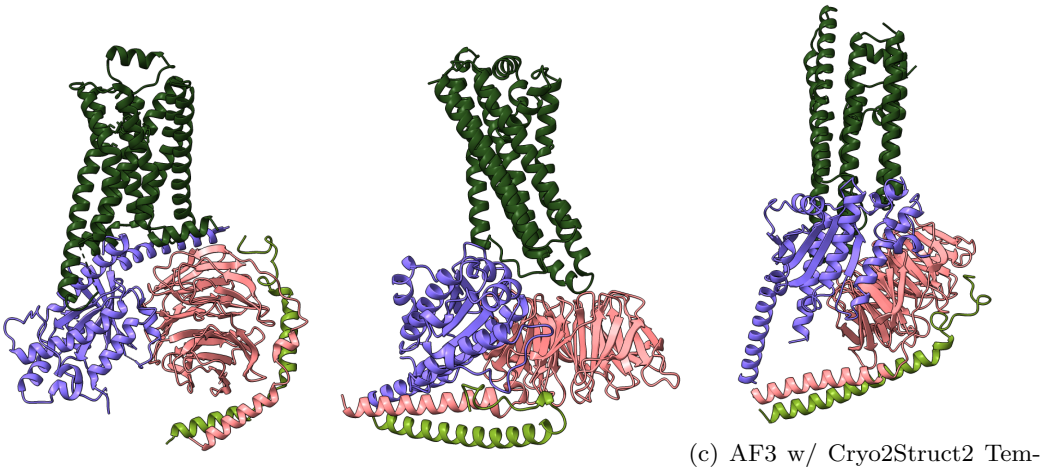
(a) PDB Structure (b) AF3 w/o Template (c) AF3 w/ Cryo2Struct2 Template

Figure 5.4: Example of modeled structures. **(a)** PDB-deposited structure (PDB Code: 8C1W). **(b)** AlphaFold3 prediction without templates has TM-Score of 0.265. **(c)** AlphaFold3 prediction using a Cryo2Struct2-generated template has TM-Score of 0.398.



(a) PDB Structure (b) AF3 w/o Template (c) AF3 w/ Cryo2Struct2 Template

Figure 5.5: Example of modeled structures. **(a)** PDB-deposited structure (PDB Code: 8BL8). **(b)** AlphaFold3 prediction without templates has TM-Score of 0.278. **(c)** AlphaFold3 prediction using a Cryo2Struct2-generated template has TM-Score of 0.400.



(a) PDB Structure (b) AF3 w/o Template (c) AF3 w/ Cryo2Struct2 Template

Figure 5.6: Example of modeled structures. **(a)** PDB-deposited structure (PDB Code: 8GE1). **(b)** AlphaFold3 prediction without templates has TM-Score of 0.412. **(c)** AlphaFold3 prediction using a Cryo2Struct2-generated template has TM-Score of 0.451.

guidance yields a TM-score of 0.278, indicating a lower structure match. However, the structure predicted using Cryo2Struct2 guidance demonstrates improved alignment with the reference structure, achieving a higher TM-score of 0.4. These examples highlight the effectiveness of integrating Cryo2Struct2 templates in correcting and improving structure accuracy.

5.5 Discussions

In this work, we demonstrated the effectiveness of using cryo-EM-based predicted atomic structures as templates for AlphaFold3 to refine and correct protein structures, ensuring they accurately represent the atomic details derived from the cryo-EM density maps. By leveraging only the templates generated from cryo-EM predictions, we were able to predict structures that closely correspond to the experimental density maps, which holds great potential for identifying novel proteins and understanding biological mechanisms relevant to drug discovery. AlphaFold3's [15] ability to refine structures is particularly valuable in regions where the cryo-EM-based method struggles, such as areas with low resolution and missing electron density values that typically complicate accurate structure modeling.

In the future, the approach outlined here can be extended to support the prediction of joint structures of complexes, including proteins, nucleic acids, small molecules, ions, and modified residues. By integrating cryo-EM data with AlphaFold3's advanced structure modeling capabilities, we have developed a versatile framework that enables the generation of complex structures, broadening the

scope of cryo-EM-based complex structure modeling.

Our experiments highlight that AlphaFold3 not only refines cryo-EM-based structures but also benefits from these cryo-EM predictions as templates, creating a mutually reinforcing feedback loop that improves both methods. Additionally, the alignment strategy implemented in Cryo2Struct [72] (chapter 4) has proven to be effective as a standalone tool, with potential for adaptation to other deep learning models. Future work will focus on further improving this alignment strategy to improve the quality and accuracy of generated atomic structural models.

Chapter 6

IMPROVING PROTEIN–LIGAND INTERACTION MODELING WITH CRYO-EM DATA, TEMPLATES, AND DEEP LEARNING IN 2021 LIGAND MODEL CHALLENGE

6.1 Abstract

Elucidating protein–ligand interaction is crucial for studying the function of proteins and compounds in an organism and critical for drug discovery and design. The problem of protein–ligand interaction is traditionally tackled by molecular docking and simulation, which is based on physical forces and statistical potentials and cannot effectively leverage cryo-EM data and existing protein structural information in the protein–ligand modeling process. In this work, we developed a deep learning bioinformatics pipeline (DeepProLigand) to predict protein–ligand interactions from cryo-EM density maps of proteins and ligands. DeepProLigand first uses a deep learning method to predict the structure of proteins from cryo-EM maps, which is averaged with a reference (template) structure of the proteins to produce a combined structure to add ligands. The ligands are then identified and added into the structure to generate a protein–ligand complex structure, which is further refined. The method based on the deep learning prediction and template-based modeling was blindly tested in the 2021 EMDataResource Ligand Challenge in fitting ligands to cryo-EM density maps. These results demonstrate that the deep learning bioinformatics approach is a promising direction for modeling protein–ligand interactions on cryo-EM data using prior structural information.

6.2 Introduction

Proteins are a building block of life and carry out many vital biological functions. Whether acting as an enzyme to accelerate the chemical reactions, or as regulatory molecules binding to other molecules to activate their functions, the detailed characterization of proteins and their interaction with their binding partners (e.g., the natural substrates or drugs as ligands) is of great importance. Protein-ligand interactions are necessary requirements for signal transduction, immune responses, and gene regulation in living organisms. The study of protein–ligand interactions is important in understanding the mechanisms of biological regulation and provides a theoretical basis for the design and discovery of new drugs. A fundamental objective of computational structural biology is to understand and model such molecular interactions of living systems in sufficient detail so that the behavior of the system can be predicted or modified as desired. In order to characterize the thermodynamic and kinetic behavior of components and their interactions of living organisms, an image of interacting molecules, such as protein-ligand complexes, at near atomic resolution is required to analyze and understand the physical and geometrical constrains of the molecules.

Cryo-EM, an acronym for the cryogenic electron microscopy technique [84], is a revolutionary technology that enables the determination of a 3D structure of macro-molecular complexes at atomic resolution. With the development of various techniques in the cryo-EM realm to generate high resolution maps, as seen in Figure 6.1, EMDataResource [6] has seen a surge in the deposition of cryo-EM derived protein density maps which elucidate the protein and ligand interactions in the molecules. The EMDataResource 2021 Ligand Model Challenge [103] was hosted to rigorously benchmark the current methods for generating models using cryo-EM density maps to improve the prediction and validation of protein and ligand interactions, and to identify the metrics which are most suitable for comparing the fit of atomic coordinate models into the cryo-EM maps.

One of the most popular approaches to modeling the protein–ligand complexes is the molecular docking [104, 105, 106, 107, 108], which uses physics- or statistical potential-based molecular simulations to generate protein–ligand complex models and a scoring function for estimation of their binding affinities to rank them. With the recent advancement in the field of deep learning, another most prominent approach to modeling protein–ligand complexes is deep learning-based methods. Deep learning-based methods predict protein–ligand binding sites [109, 110, 111, 112] using various neural network architectures such as convolution neural networks (CNN), long short-term memory networks (LSTM), and residual networks (ResNet). These methods primarily use three databases: BioLiP

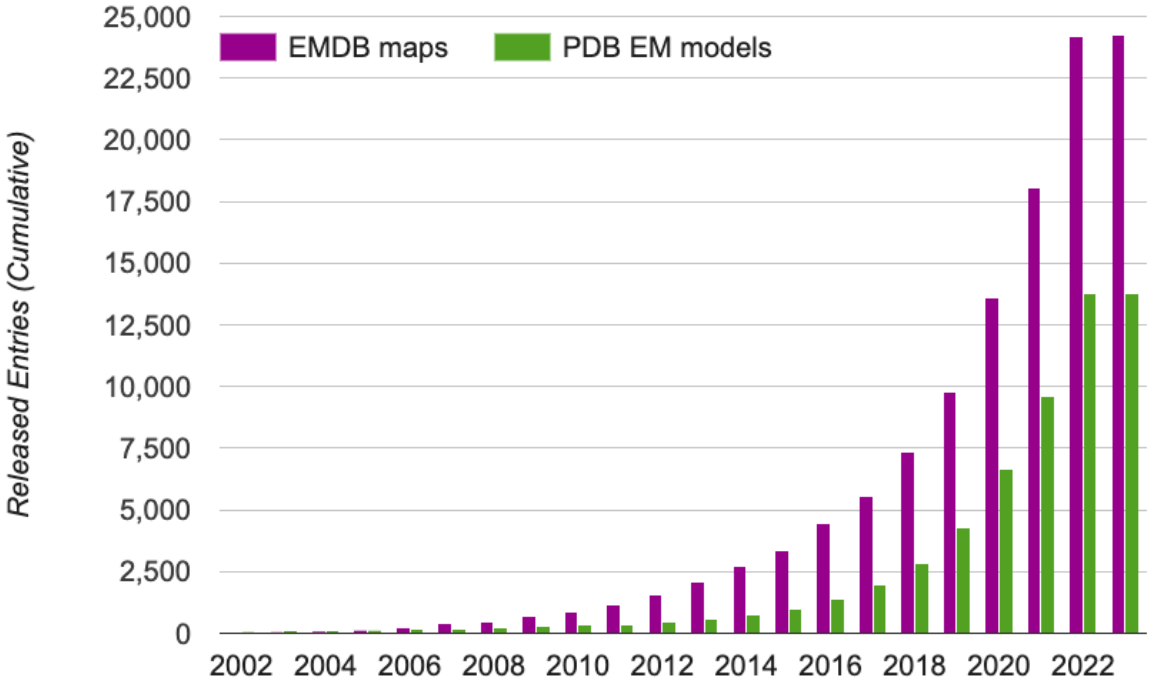


Figure 6.1: The growth of cryo-EM density maps and cryo-EM-derived protein structures. The statistics were obtained from EMDataResource [6], a unified data resource for 3-Dimension electron microscopy (3DEM) on 8 January 2023.

[113], ATPBind [114] and Sc-PDB [115] to train and validate their deep learning models before making binding-site predictions. Similarly, deep learning architectures, such as CNNs, graph neural networks, and attention mechanisms, are used for the prediction of protein–ligand binding affinity [116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128]. These methods mainly make use of two databases: PDBbind [129] and the CASF databases [130] for binding affinity predictions. More advanced methods such as Equibind [131]—an SE(3) [64] -equivariant geometric deep learning model for direct-shot prediction of receptor binding location and ligand's bound pose—and DIFFDOCK [132]—a diffusion generative model tailored to the task of molecular docking—have been developed recently. However, even with significant research efforts, despite some success, the protein–ligand interaction prediction problem still remains unsolved because existing methods cannot leverage vast structural data effectively.

Inspired by the success of AlphaFold [34], which uses a novel deep learning architecture to predict the protein structures using amino acid sequence data as an input, as well as the various deep learning-based protein–ligand modeling techniques, we adapted the deep learning-based approach for our work. In this work, we combined the deep learning-based protein structure prediction tool DeepTracer with template-based protein–ligand interaction prediction in order to determine the structures of

protein–ligand complexes for the 2021 Ligand Model Challenge that was held from 1 February to 1 April 2021. Based on the official results provided by the assessors of the challenge, our method performed best in fitting ligands to cryo-EM maps (measured across all targets), demonstrating the unique value of the deep learning bioinformatics approach for modeling protein-ligand interaction.

6.3 Methods

We attempted to solve the problem of protein–ligand interaction by using a set of bioinformatics methods, incorporating cryo-EM data and known structural information such as reference protein structures. In particular, we leveraged the recent advance of applying deep learning to directly predict the structure of proteins from high-resolution cryo-EM density maps; a succinct review of the methods can be found in Ref. [70] and chapter 2. To predict the bound conformation (3D atomic structure) of a protein–ligand complex, we utilized an existing deep learning-based tool as a key component of our model building pipeline (DeepProLigand). DeepProLigand predicts the 3D coordinates of protein structures using only a cryo-EM density map as an input. This protein structure model is a starting point for the downstream ligand positioning and model refinement tasks. The workflow illustrated in Figure 6.2 demonstrates our approach to generating the structure of a protein complex by incorporating a fully automatic deep learning-based method as its primary building block. The modeling pipeline of Figure 6.2 has three key steps described as follows:

6.3.1 Protein Complex Reconstruction from cryo-EM Density Maps and Reference Structures

Using DeepTracer [56], we first predicted the 3D backbone coordinates of the protein complex directly from a cryo-EM density map. DeepTracer uses a 3D U-Net architecture which is modified from the original 2D U-Net [43] architecture developed for biomedical image segmentation. The output from the DeepTracer block is a predicted 3D backbone coordinate structure that has the carbon, carbon alpha, nitrogen and oxygen atoms in the Protein Data Bank Format (PDB), which is standardized by wwPDB [21]. The predicted structure reflects the conformation of the protein in the ligand binding mode. Because the reference structure of the protein (prior structure without cryo-EM information) is also provided by 2021 Ligand Model Challenge organizers, we used the structural alignment to combine them to generate a posterior structure, conceptually similar to combining the prior probability and likelihood to generate a posterior probability in the Bayesian reasoning.

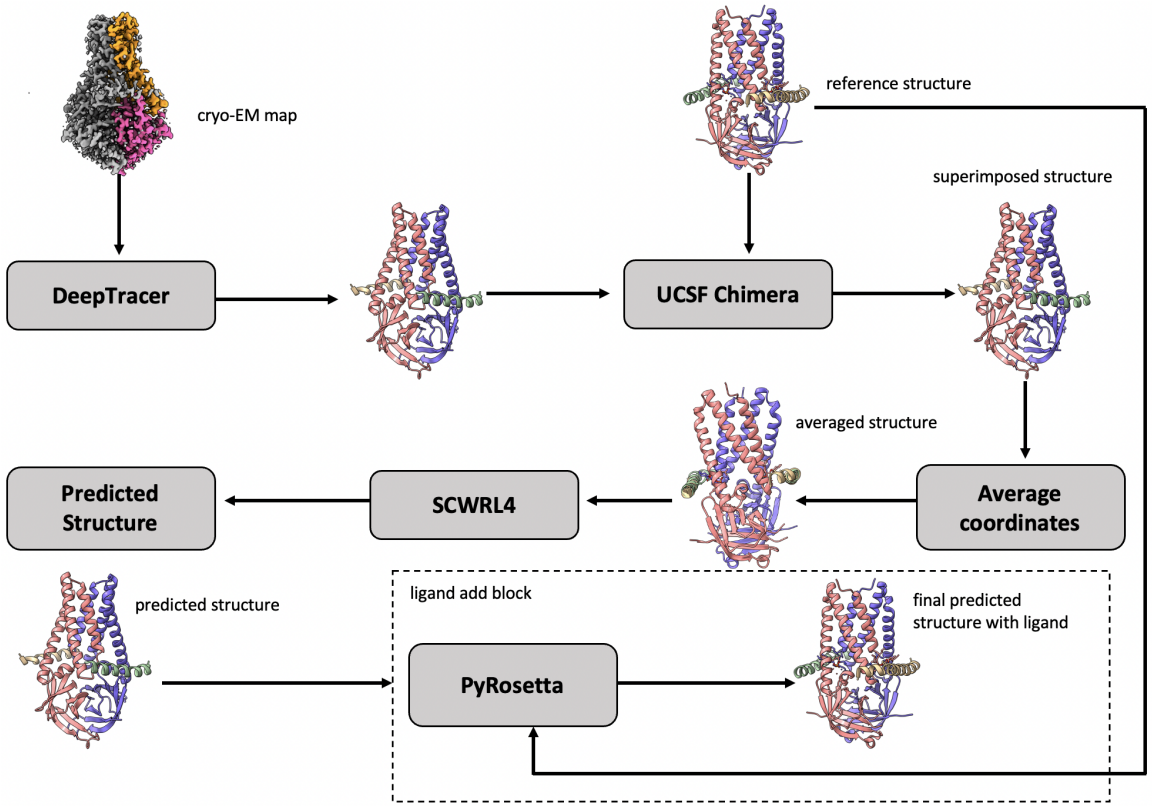


Figure 6.2: The workflow of DeepProLigand generating protein complex structure from cryo-EM map and reference structure. The cryo-EM map (EMD-22898) illustrated in the workflow is of a SARS-CoV-2 ORF3a ion channel in lipid nanodiscs [7]

Specifically, in order to combine the reference structure and the predicted structure together in terms of geometrical alignment, we utilized the UCSF Chimera's [9] matchmaker function to superimpose both structures together. Once the structures were superimposed, we saved the superimposed structure relative to reference structure into a new PDB file. The new PDB file then contained the atoms of both reference and predicted structures in the same geometrical space, allowing us to average the coordinates of the corresponding backbone atoms and utilize the reference structure's residue, and chain labeling for all the shared components between the two structures. The side chains were added on top of the combined backbone structures using the SCWRL4 [133] tool. Finally, a full-atom combined structure, consisting of multiple chains, was produced for the downstream processing. It is worth noting that our approach of generating protein complex structures was different from the traditional approach of fitting a reference structure into a cryo-EM density map.

Algorithm 1 depicts the pseudo code of averaging the backbone atoms' coordinates of reference and predicted structures. We started by initializing an empty PDB file named "average structure" that followed the guidelines of wwPDB [21]. For each residue of the reference structure, if the

Algorithm 1: Average predicted structure and reference structure

a) **Input:** threshold *(threshold can be modified per chain)*

b) Compute *distance* using Equation (6.1)

c) Initialize: $x_{avg} = 0,\ y_{avg} = 0,\ z_{avg} = 0$

d) If $distance < threshold$

   (a) $x_{avg} = (x_r + x_p)/2$
   (b) $y_{avg} = (y_r + y_p)/2$
   (c) $z_{avg} = (z_r + z_p)/2$

e) Else

   (a) $x_{avg} = x_r$
   (b) $y_{avg} = y_r$
   (c) $z_{avg} = z_r$

distance between the reference structure's carbon alpha atom and any carbon alpha atom of predicted backbone structure in the 3D geometrical space was less than a threshold value, then all the backbone atoms' coordinates of the residue in the particular reference structure were averaged with the predicted structure's corresponding residue and saved in the average structure PDB file; otherwise, the coordinates of the residue in reference structure were simply saved in the average structure PDB file. We refer to the arithmetic mean, the sum of collection of numbers divided by the count of numbers, as "average" throughout the paper. The default distance threshold value we used is 1 Angstrom (Å); however, the threshold value for each chain can be modified as desired.

$$distance = \sqrt{(x_r - x_p)^2 + (y_r - y_p)^2 + (z_r - z_p)^2}, \tag{6.1}$$

$$threshold = 1\ Angstrom. \tag{6.2}$$

Here, let x_r, y_r, and z_r be the coordinates for each carbon alpha residue of the reference structure and x_p, y_p, and z_p be the coordinates for each carbon alpha residue of the predicted structure. With this notation, we followed Algorithm 1 and generated new coordinates x_{avg}, y_{avg}, and x_{avg} which are the average coordinates of both reference and predicted structures. After computing the distance using Equation (6.1), we used a threshold value as shown in Equation (6.2) for Target 202 and Target 203 of the 2021 EMDataResource Ligand Challenge. For Target 201 of the challenge, since most of the chains were turned into coils, we used a threshold value of 0.3 Angstrom (Å) for chain C and 0.5 Angstrom (Å) for all other chains. The averaged coordinate structure was saved into a standard

Table 6.1: Number of residues averaged for target T201: EMD 7770.

Target T201 : EMD 7770			
Chain ID	**Total Residues**	**Averaged Residues**	**% of Residues Averaged**
Chain A	1021	845	82.8
Chain B	1021	845	82.8
Chain C	1021	461	45.2
Chain D	1021	852	83.4
Average % across chains			73.55

Table 6.2: Number of residues averaged for target T202: EMD 30210.

Target T202 : EMD 30210			
Chain ID	**Total Residues**	**Averaged Residues**	**% of Residues Averaged**
Chain A	834	762	91.4
Chain B	114	105	92.1
Chain C	63	48	76.2
Average % across chains			86.6

PDB file format. Tables 6.1–6.3 show the number of residues averaged per chain for each target. Target T201 has 73.55% residues averaged, target T202 has 86.60% residues averaged, and finally target T203 has 57.70% residues averaged.

After the backbone atoms were computed using Algorithm 1, we utilized SCWRL4 [133] to add the side-chain conformation into the protein structure. The deep learning-based method utilized to predict the backbone atoms had a high impact on determining the side-chains conformation as well, because high side chain accuracy is often achieved when the backbone prediction is accurate, as also demonstrated by AlphaFold [34].

Table 6.3: Number of residues averaged for target T203: EMD 22898.

Target T203 : EMD 22898			
Chain ID	**Total Residues**	**Averaged Residues**	**% of Residues Averaged**
Chain A	193	184	95.3
Chain B	193	0	0
Chain C	31	20	64.5
Chain D	31	22	71.0
Average % across chains			57.7

6.3.2 Template-Based Prediction of Protein-Ligand Interaction

After the protein structure that can accommodate ligands was generated using Algorithm 1, we utilized PyRosetta [134] to identify ligands and add them into the predicted structure by using the reference structure as a template, as depicted in Algorithm 2. The reference structure contains the ligands' atomic coordinates. Since PyRosetta is a residue based tool, when a pose is created, all the atoms in a structure including ligand atoms are indexed by residue indices. Following Algorithm 2, we let *res* be each residue in the reference structure that we checked for whether it was a ligand. PyRosetta's *is_ligand* function works by comparing the ligand to a chemical component dictionary and returns a bool value (i.e, either True for ligand or False for non ligand) for each residue.

Algorithm 2: Identify ligands and include them into average structure.

a) **Require:** pyrosetta

b) Initialize *pyrosetta*

c) $pose\_ref = pose\_from\_pdb(reference\ structure)$

d) If $pose\_ref.residue(res\_id).is\_ligand() ==$ True

   (a) **with** $open(\text{"}average\_structure.pdb\text{"}, \text{"}a\text{"})$ as file:

   (b) $file.write(residue)$

e) Else

   (a) *do nothing*

6.3.3 Refinement of Protein-Ligand Complex Model

After the prediction of the protein–ligand complex structure using the approach outlined above, we further refined the predicted complex structure using Rosetta FastRelax. Relax does not perform extensive refinement and only searches the local low-energy backbone and side-chain conformations near the starting conformations by implementing rounds of packing and minimizing, with repulsive weight in the scoring function gradually increasing from a low value to a normal value. The scoring function we used was *ref2015_cst.wts*, which is a default score function, repeated five times. Finally, after the refinement of the protein complex, we used UCSF Chimera's *Fit in Map* function to perform a rigid body optimization of the refined model. The 3D structure was rotated and aligned so that it fit to the density map. This refinement step was optional. During the blind experiment of the 2021 EMDataResource Ligand Challenge, we submitted both an unrefined model and a refined model for each target.

6.3.4 Target cryo-EM Density Maps of 2021 Ligand Challenge

We blindly tested the protein–ligand modeling pipeline DeepProLigand on three targets that were released as 2021 EMDataResource Ligand Challenge targets from February to April 2021. The next section elaborates the three targets and the experimental setting used for each target.

Target 201: Escherichia coli Beta-galactosidase

The β-Galactosidase [135] target with atomic resolution of 1.9 Angstrom (Å) contains protein Beta-galactosidase, magnesium ion, sodium ion, water and 2-phenylethyl 1-thio-beta-D-galactopyranoside (PTQ) as a ligand. The EMDB ID of the target in EMDataResource is EMD-7770. We predicted the 3D structure of the complex using the workflow of DeepProLigand, as highlighted in Figure 6.2 and, during averaging of the structure, we initialized a 0.3 Angstrom (Å) distance threshold for chain C and a 0.5 Angstrom (Å) distance threshold for all other chains of the complex by re-initializing the threshold value of Equation (6.2). The reason for threshold of 0.3 Å in chain C was because most of the chains were turned into coils/turns with a threshold of 0.5 Å. The ligand PTQ was appended using Algorithm 2. Figure 6.3 shows the map–model overlay of cryo-EM density map EMD-7770 and our reconstructed protein structure model.

Target 202: SARS-CoV-2 RNA-Dependent RNA Polymerase

The nsp12-nsp7-nsp8 complex bound to the template-primer RNA and triphosphate form of Remdesivir(RTP) [136] target with an atomic resolution of 2.5 Angstrom (Å) contains RNA-directed RNA polymerase, Non-Structural Protein 8, Non-Structural Protein 7, Primer, Templete, ZINC ION, PYROPHOSPHATE 2-, MAGNESIUM ION, water, and [(2 R,3 S,4 R,5 R)-5-(4-azanylpyrrolo[2,1-f][1,2,4]triazin-7-y)-5-cyano-3,4-bis(oxidanyl)oxolan-2-yl]methyl dihydrogen phosphate as aligand. The EMDB ID of the target in EMDataResource is EMD-30210. We predicted the 3D structure of the complex using the workflow of DeepProLigand as highlighted in Figure 6.2 and, during averaging of the structure, we used a 1 Angstrom (Å) distance threshold for all chains of the complex. The ligand F86 (remdesivir, covalent inhibitor) was appended using Algorithm 2. Figure 6.4 shows the map–model overlay of cryo-EM density map EM-30210 and our reconstructed protein structure model.

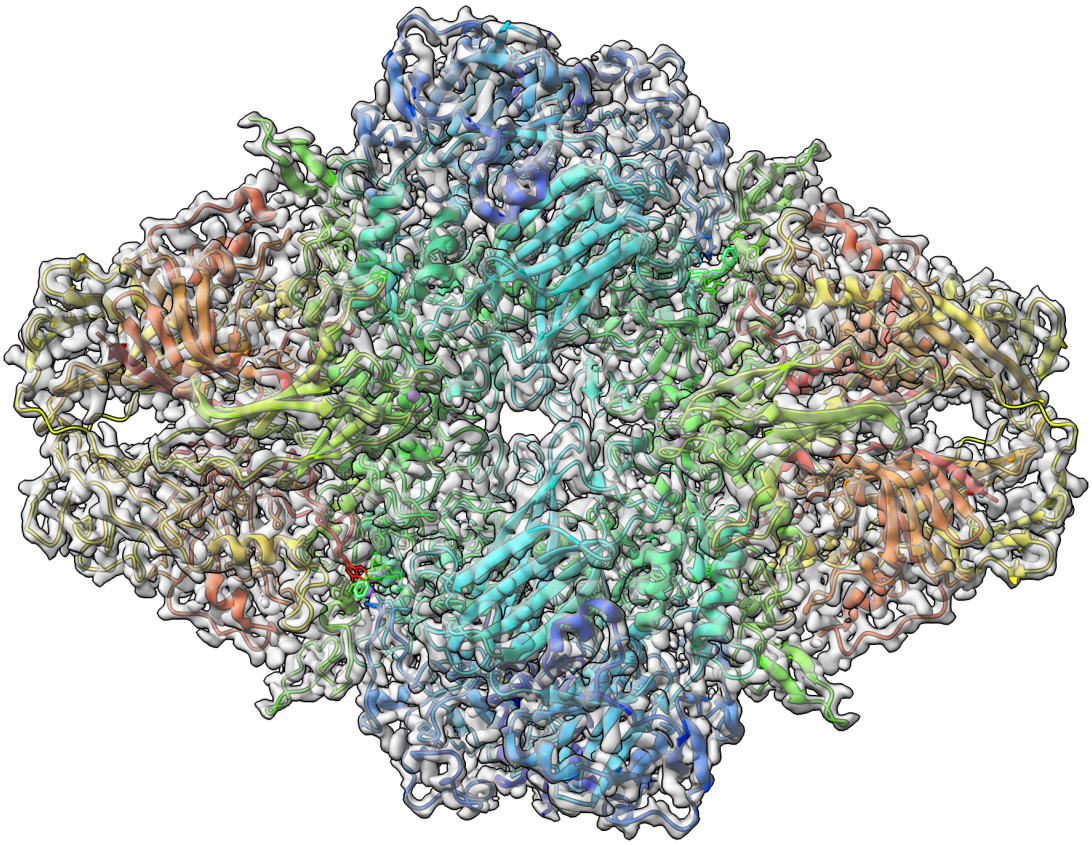


Figure 6.3: Target 201 (EMD-7770) map-model overlay at the recommended contour 0.52 (3.3 σ) with T0201EM0004_1 (ours).

Target 203: SARS-CoV-2 Protein 3a in Lipid Nanodiscs

The SARS-CoV-2 3a ion channel in lipid nanodiscs [7] target with atomic resolution of 2.08 Angstrom (Å) contains ORF3a protein, Apolipoprotein A-I, water and 1,2-Dioleoyl-sn-glycero-3-phosphoethanolamine as a ligand. The EMDB ID of the target in EMDataResource is EMD-22898. We predicted the 3D structure of the complex using the workflow of DeepProLigand as highlighted in Figure 6.2 and, during averaging of the structure, we used a 1 Angstrom (Å) distance threshold for all chains of the complex. The ligand PEE was appended using Algorithm 2. Figure 6.5 shows the map–model overlay of cryo-EM density map EMD-22898 and our reconstructed protein structure model.

6.4 Results

The analysis of the models in this section is based on the official results provided by the organizers of the 2021 Ligand Model Challenge. The fit to a map for a ligand was assessed by the Q-score

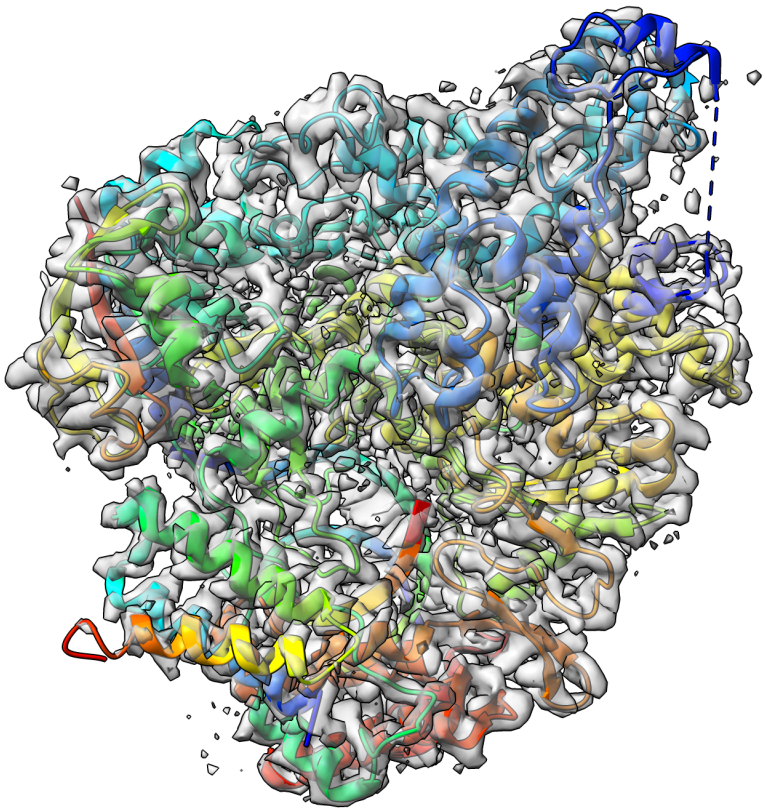


Figure 6.4: Target 202 (EMD-30210) map-model overlay at the recommended contour 0.058 (4.3 σ) with T0202EM004_1 (ours).

[137] and the Z-scores. The Q-score measures how similar map values around an atom are to a Gaussian-like function which we would see if the atom were well resolved. The Q-score was calculated as a correlation between two vectors: u, which contained map values at points around the atom, and v, which contained values obtained from the reference Gaussian.

We used the Q-score to compare the map-to-model fit for all the models that were submitted to the challenge. Table 6.4 shows the Q-score of the ligand for all the models submitted for Target 201; our model is highlighted in bold for scrutiny. Figure 6.6 shows the ligand (PTQ)'s binding pose and orientation in our best predicted model, T0201EM004_1. Ligand PTQ bound to all four chains of Target 201, resulting in four binding sites for the ligand. We visualized three binding locations for the ligand with its binding pose and orientations in Figure 6.6. Table 6.5 shows the Q-score of the ligand for Target 202 and, similar to Target 201, our model is highlighted in bold for scrutiny. Figure 6.7 shows ligand (F86)'s binding pose and orientation in our best predicted model, T0201EM004_1. Ligand F86 bound to only one location in Target 202. We visualized the binding location for the ligand with its binding pose and orientations in Figure 6.7.

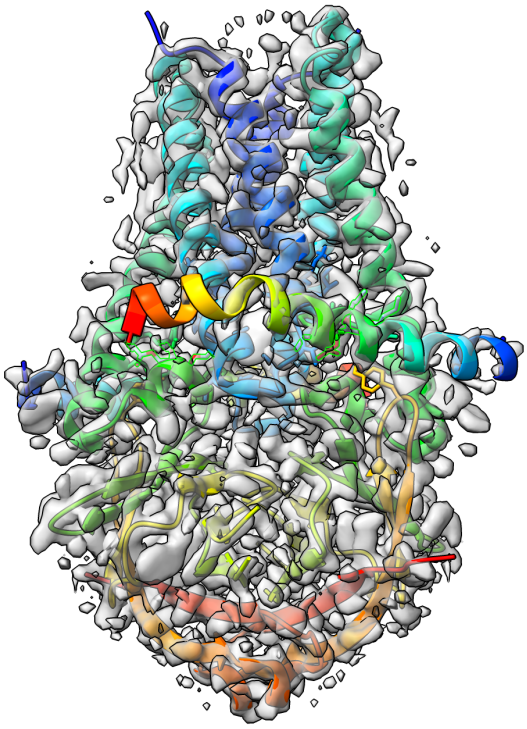


Figure 6.5: Target 203
(EMD-22898) map-model overlay at the recommended contour 0.7 (10.3 σ) with T0203EM004_1 (ours).

Table 6.6 shows the Q-score of the ligand for Target 203. Similar to Target 201 and 202, our model is highlighted in bold for scrutiny in Table 6.6. Figure 6.8 shows the ligand (PEE)'s binding pose and orientation in our best predicted model, T0201EM004_1. Ligand PEE bound to two locations in Target 203. We have visualized the binding locations for the ligand with its binding pose and orientation in Figure 6.8.

Figure 6.9 shows cumulative Z-scores on Q-scores of 17 groups participating in the 2021 Ligand Model Challenge. Our DeepProLigand predictor (EM004) performs best overall on all three targets. Specifically, our protein–ligand model was ranked first for Target 202, second for Target 203, and in the middle for Target 201 as shown in the Z-scores on Q-scores in Figure 6.9. The Q-scores of our best model for the three targets are shown in Table 6.7. Even though there are too few targets to draw a definite conclusion, the good results indicate that the deep learning structure prediction in conjunction with the template reference structure is able to build a good protein structure framework to accommodate ligands, and the template-based protein–ligand prediction can assemble the ligands with the protein structure well for some targets. Incorporating a deep learning approach in modeling

Table 6.4: Evaluation of Target 201: Escherichia coli beta-galactosidase on Q-score for all the models submitted in the 2021 Ligand Model Challenge

Team Name	PTQ (Ligand)
T0201EM014_1	0.82
T0201EM005_2	0.81
T0201EM012_1	0.81
T0201EM006_1	0.79
T0201EM009_1	0.78
T0201EM015_1	0.77
T0201EM005_1	0.73
T0201EM002_2	0.72
T0201EM004_1	**0.71**
T0201EM003_1	0.71
T0201EM003_2	0.71
T0201EM010_1	0.69
T0201EM011_1	0.64
T0201EM001_2	0.64
T0201EM013_1	0.63
T0201EM001_3	0.62
T0201EM002_3	0.60
T0201EM003_3	0.58
T0201EM002_1	0.55
T0201EM001_1	0.33
T0201EM007_1	0.31
T0201EM008_1	-

Note: Table is sorted in descending order using ligand: PTQ score. "-" means, we were unable to calculate the score of the model. Our best model is highlighted in bold.

enables us to predict the protein structure directly from the cryo-EM map within minutes, making the approach highly useful in terms of both prediction accuracy and time.

We also noticed the limitation of our approach in terms of the geometric quality of the atoms in the predicted protein structure, however. Particularly, there were some atom–atom clashes in the models, which may be caused by the violations of some geometric constraints of atom–atom distances in the protein structure predicted by the deep learning, as well as in the averaging of the coordinates of the predicted structure and the reference structure. The violations of geometric and stereochemical restraints were not fixed by the current refinement protocol in the prediction pipeline. The refinement protocol even introduced some new clashes into the model. AlphaFold [34] demonstrated that the well-trained sophisticated deep learning architecture can accurately capture the geometric restraints of atoms and bonds in protein structures by predicting high-quality protein structures of atomic resolution that are highly similar to natural protein structures; this means more advanced deep learning architectures can be developed to predict high-quality protein structures compatible with the geometric and stereochemical restraints of proteins from cryo-EM density maps

Table 6.5: Evaluation of Target 202: SARS-CoV-2 RNA-dependent RNA polymerase on Q-score for all the models submitted to the 2021 Ligand Model Challenge.

Team Name	F86 (Ligand)
T0202EM004_1	**0.74**
T0202EM009_1	0.71
T0202EM006_1	0.69
T0202EM005_1	0.68
T0202EM012_1	0.68
T0202EM002_2	0.68
T0202EM003_2	0.68
T0202EM010_1	0.67
T0202EM003_1	0.67
T0202EM008_1	0.63
T0202EM001_1	0.60
T0202EM001_2	0.59
T0202EM013_1	0.59
T0202EM007_1	0.57
T0202EM011_1	0.56
T0202EM002_1	0.52

Note: Table is sorted in descending order using ligand: F86 score. Our best model is highlighted in bold.

and reference structures.

6.5 Conclusion and Future Works

In this work, we demonstrate that the deep learning prediction of protein structures from cryo-EM maps can generate good protein structures for constructing protein–ligand complexes and the template-based protein–ligand interaction prediction can fit ligands well into the predicted protein structures according to the outstanding performance of our protein–ligand modeling pipeline. It is also worth noting that our method was fully automatic and did not involve any manual tweaking of the models to improve the scores. As discussed before, the current protein–ligand prediction pipeline cannot resolve some violations of some geometric and stereochemical restraints of atoms in protein structures. We plan to soon develop advanced end-to-end deep learning architectures, similar to some components in AlphaFold, to better predict better protein structures from cryo-EM maps and reference structures. Moreover, we plan to design 3D-equivariant deep learning architectures like the SE(3)-equivariant Transformer network [17, 64] to tackle the problem of geometric constraints which are not addressed by current methods. Finally, an end-to-end direct deep learning prediction of the structure of protein–ligand complexes from cryo-EM density maps, reference structures and ligand information to fully automate all the steps of the entire pipeline in this work will be pursued. We believe the application of a deep learning approach to the prediction of 3D structures of protein–ligand

Table 6.6: Evaluation of Target 203: SARS-CoV-2 ORF3a putative ion channel in nanodisc on Q-score for all the models submitted in the 2021 Ligand Model Challenge.

Team Name	PEE (Ligand)
T0203EM0016_1	0.77
T0203EM004_1	**0.76**
T0203EM0012_1	0.75
T0203EM005_1	0.74
T0203EM0010_1	0.73
T0203EM003_1	0.73
T0203EM003_2	0.70
T0203EM0011_1	0.72
T0203EM002_2	0.72
T0203EM009_1	0.71
T0203EM002_1	0.70
T0203EM006_1	0.70
T0203EM002_3	0.69
T0203EM0014_1	0.67
T0203EM001_2	0.66
T0203EM001_3	0.63
T0203EM008_1	0.63
T0203EM001_1	0.60
T0203EM007_1	0.51

Note: Table is sorted in descending order using ligand: PEE score. Our best model is highlighted in bold.

Table 6.7: Q-score of our best model (EM004_1) for three targets.

Target Name	Ligand
Target 201	0.71 (PTQ)
Target 202	0.74 (F86)
Target 203	0.76 (PEE)

Note: Among two models submitted for each target in the challenge, our best model's id is EM004_1 across all the targets. EM004_1 model is the non refined model.

complexes leveraging cryo-EM and other related data is a promising avenue with which to accelerate the advancement of the study of protein–ligand interaction [66, 49]. With the proliferation of cryo-EM maps being deposited in the EMDataResource database, the use of deep learning-based methods can help to determine the structure of the protein–ligand complexes rapidly and ultimately help to expedite the drug discovery process.

6.6 Sequence Based Modeling: An Approach for Predicting Protein Structure

DeepProLigand uses DeepTracer[56] to predict the protein structure as its main component because it uses both the cryoEM density map and the amino acid sequence. We conducted another study to

predict the protein structure using AlphaFold [34]. While AlphaFold predicts the atomic coordinates of most proteins with remarkable accuracy, in this study, AlphaFold struggled to predict the atomic coordinates that fit locally into the density map per residue. Figures 6.10 and 6.11 shows the comparison of structures predicted by AlphaFold and DeepProLigand with the PDB deposited structure.

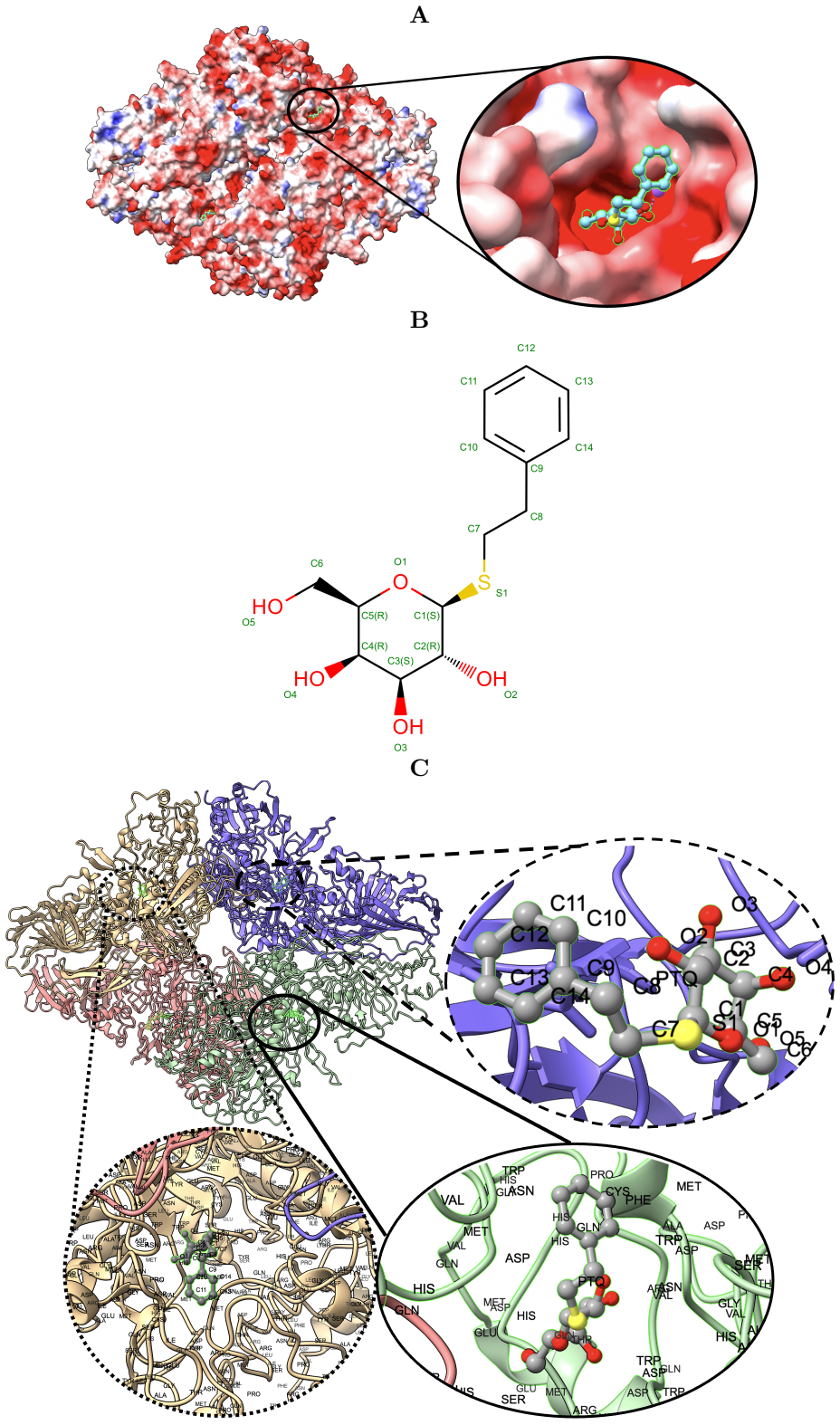


Figure 6.6: Target 201 . (**A**) T0201EM004_1 (ours) docked by Target 201 (EMD-7770) and visualized with electrostatic potential surface generated in UCSF Chimera. (**B**) Ligand PTQ, image extracted from Protein Data Bank (PDB). (**C**) Protein–ligand interactions in T0201EM004_1 (ours) model. Chains are colored differently (chain A: blue, chain B: pink, chain C: green and chain D: golden). The ligand is labeled with its atom names as well as the ligand name (PTQ). For chain D: golden and chain C: green, we have labeled the chain residue names for understanding protein–ligand interaction better.

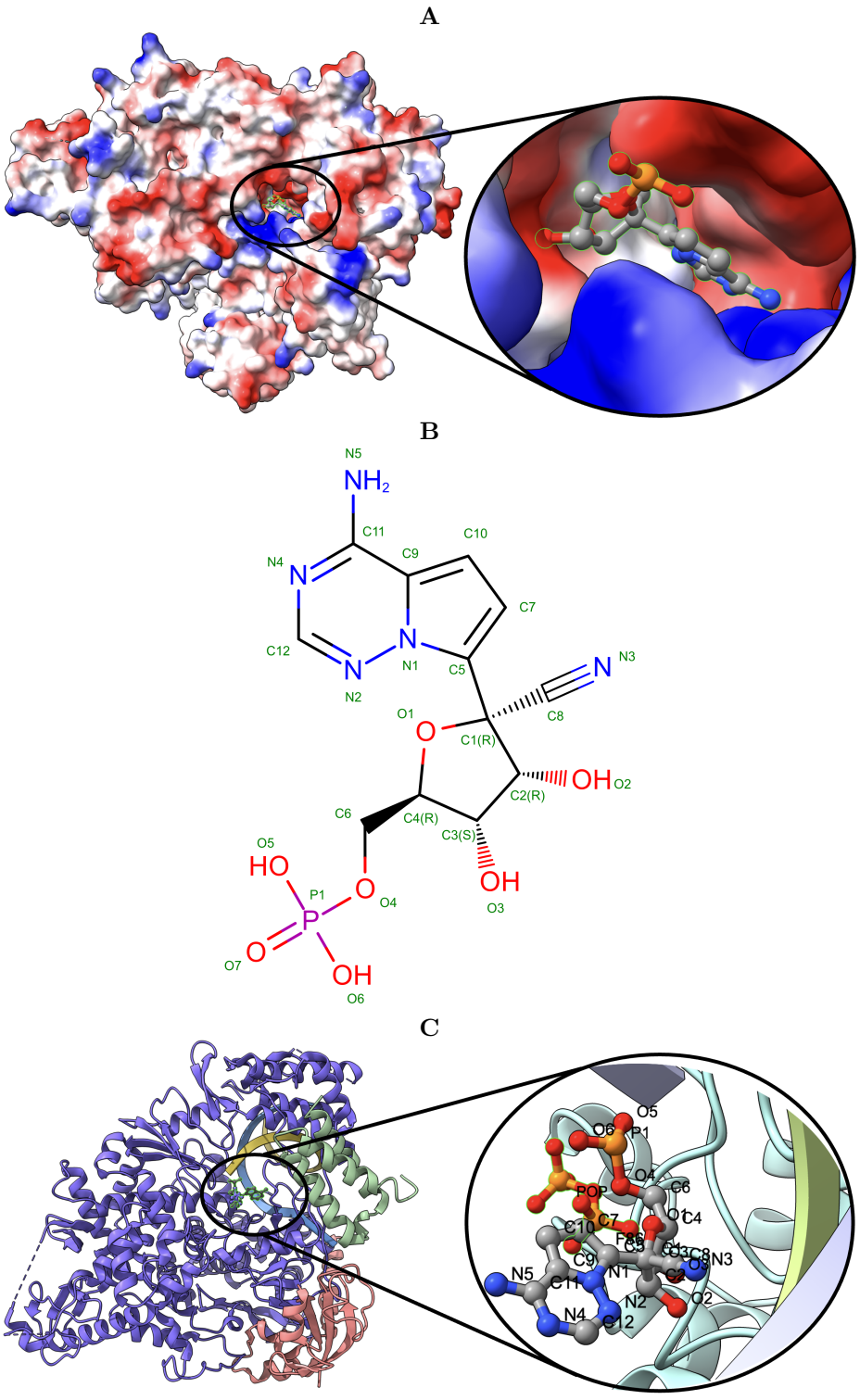


Figure 6.7: Target 202. (**A**) T0202EM0004_1 (ours) docked by Target 202 (EMD-30210) and visualized with electrostatic potential surface generated in UCSF Chimera. (**B**) Ligand F86, image extracted from Protein Data Bank (PDB). (**C**) Protein–ligand interactions in T0202EM004_1 (ours) model. Chains are colored differently (chain A: blue, chain B: orange, chain C: green, chain P: yellow, and chain T: teal). The ligand is labeled with its atom names as well as the ligand name (F86).

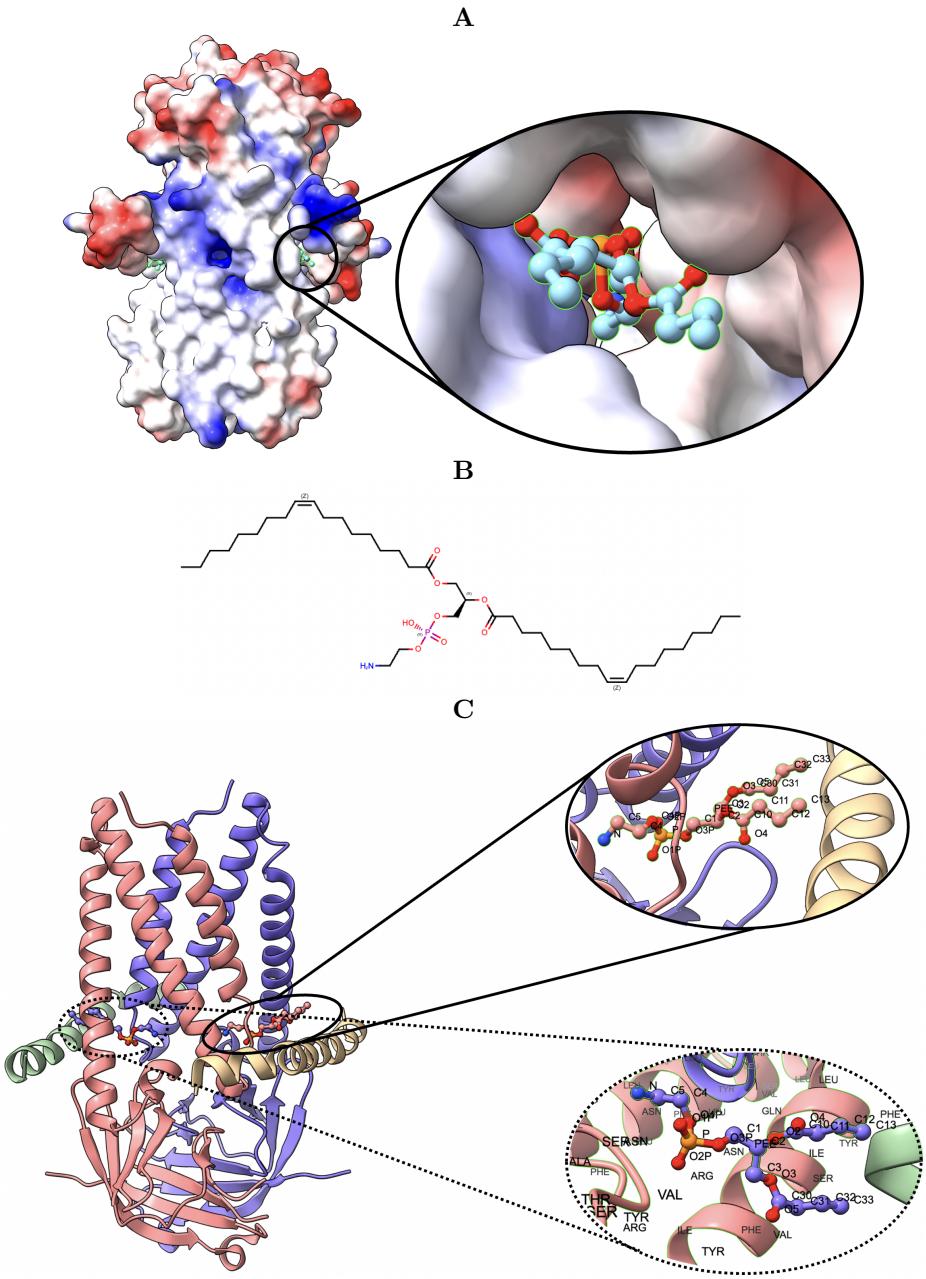


Figure 6.8: Target 203. (**A**) T0203EM0004_1 (ours) docked by Target 203 (EMD-22898) and visualized with electrostatic potential surface generated in UCSF Chimera. (**B**) Ligand PEE, image extracted from Protein Data Bank (PDB). (**C**) Protein–ligand interactions in T0203EM0004_1 (our) model. Chains are colored differently (chain A: blue, chain B: pink, chain C: green, and chain D: golden). The ligand are labeled with their atom names as well as the ligand's name (PEE).

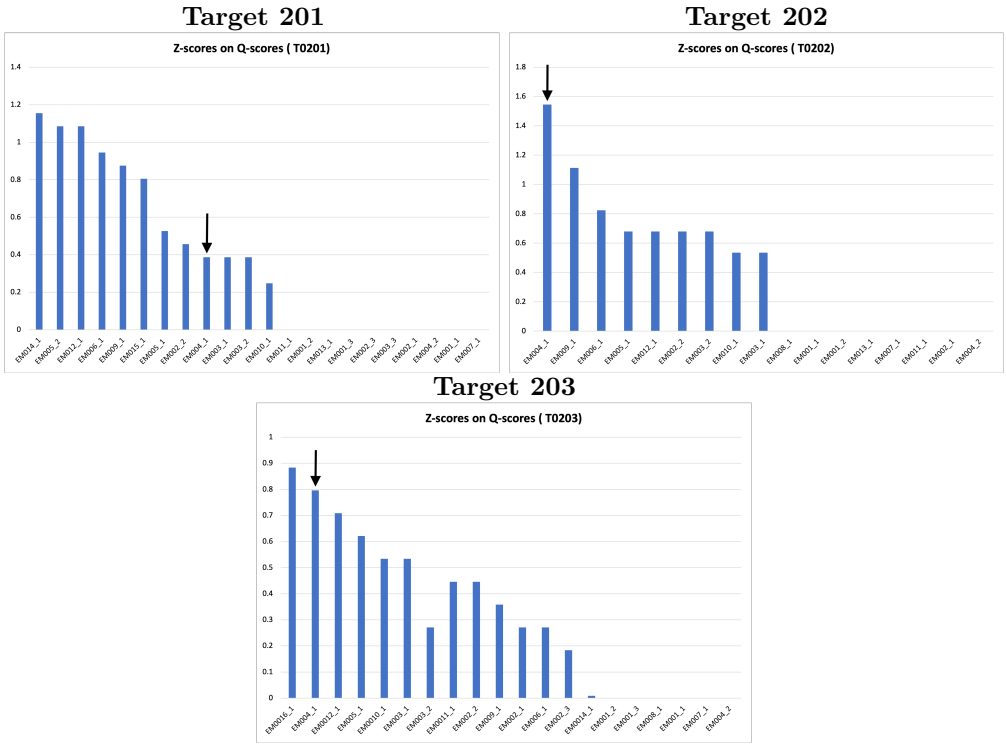
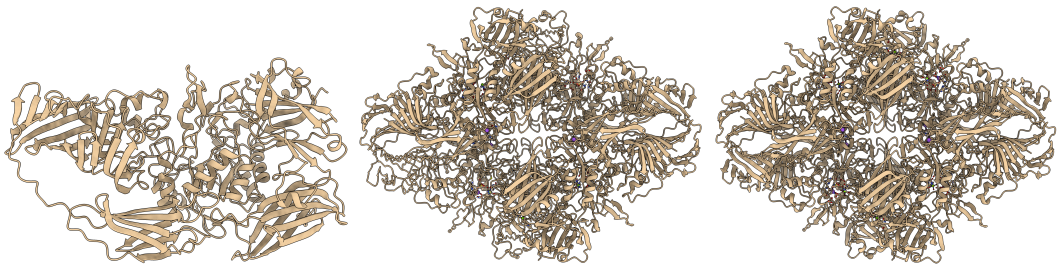
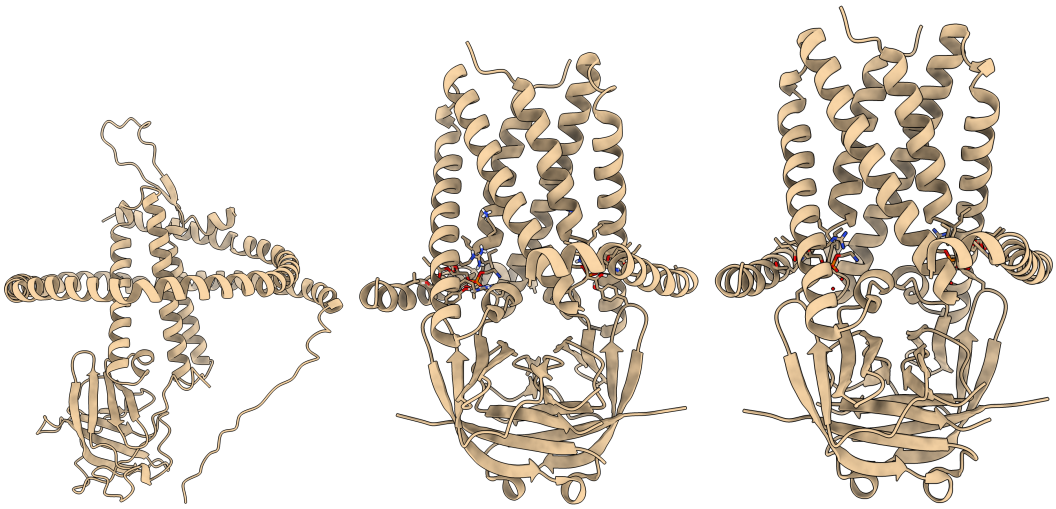


Figure 6.9: Z-scores on Q-scores for ligand of all the models submitted to 2021 Ligand Model Challenge. The pointed arrow represents our model.



(a) (b) (c)

Figure 6.10: The target T201 is of EMD: 7770. (a) AlphaFold Predicted Structure. (b) DeepProLigand Predicted Structure. (c) PDB Deposited Structure with PDB ID: 6CVM.



(a) (b) (c)

Figure 6.11: The target T203 is of EMD: 22898. (a) AlphaFold Predicted Structure. (b) DeepProLigand Predicted Structure. (c) PDB Deposited Structure with PDB ID: 7KJR.

Chapter 7

A LABELED DATASET FOR AI-BASED CRYO-EM MAP ENHANCEMENT

7.1 Abstract

Cryo-electron microscopy (cryo-EM) has transformed structural biology by enabling near-atomic resolution imaging of macromolecular complexes. However, cryo-EM density maps suffer from intrinsic noise arising from structural sources, shot noise, and digital recording, which complicates accurate atomic structure building. While various denoising methods exist, we lack the standardized datasets for benchmarking artificial intelligence (AI) approaches. Here, we present an open-source dataset for cryo-EM density map denoising comprising 650 high-resolution (1-4 Å) experimental maps paired with three types of generated label maps: regression maps capturing idealized density distributions, binary classification maps distinguishing structural elements from background, and atom-type classification maps. Each map is standardized to 1 Å voxel size and validated through Fourier Shell Correlation analysis, demonstrating substantial resolution improvements in label maps compared to experimental maps. This resource bridges the gap between structural biology and artificial intelligence communities, enabling researchers to develop and benchmark innovative denoising methods.

7.2 Background & Summary

Recent advances in cryo-electron microscopy (cryo-EM) have revolutionized structural biology by enabling the visualization of macromolecular complexes at near-atomic resolution [20]. Cryo-EM density maps are the three-dimensional volume reconstructions derived from two-dimensional images captured by electron microscopy. They have become invaluable tools for understanding complex biological structures. However, the experimental process inherently introduces artifacts that affect

image quality and subsequent interpretability of the cryo-EM density maps.

The generation of cryo-EM density maps involves the rapid exposure of flash-frozen aqueous samples to electron beams in vacuum. This process inherently produces raw images with low signal-to-noise ratios (SNR), significantly complicating the accurate interpretation of structural details. The resulting three-dimensional reconstructions consequently suffer from noise-related challenges that obstruct precise and accurate determination of atomic arrangements within the density maps. Given that the ultimate goal of cryo-EM single particle analysis is to obtain accurate atomic structures of protein complexes, these noise-related limitations present substantial obstacles to the structure building process.

Noise in cryo-EM data manifests at three distinct stages: First, *structural noise* arises from the surrounding ice matrix and often a superimposed thin carbon film. This background structure varies from one molecule to the next and is therefore inherently irreproducible. Conceptually, *structural noise* also encompasses any conformational variations within the molecule itself that are not consistently reproduced across samples. Second, *shot noise* results from the quantum nature of electron radiation, introducing statistical variations in electron detection. Third, *digital noise* emerges during the recording and digitization process - whether from photographic granularity and microdensitometer digitization noise in traditional methods, or from readout noise in modern direct electron detectors. While *shot* and *digital* noise appear as random, dust-like distributions without specific patterns (background noise), *structural noise* exhibits defined shapes with stronger density signals that can particularly confound interpretation [138].

The combined effect of these noise sources pass on through every stage of the structural determination pipeline, from initial data collection to final 3D reconstruction of cryo-EM density maps. This noise accumulation significantly hampers the global and local resolution and interpretability of the resulting density maps, making accurate atomic model building challenging. In particular, the noise can obscure important features such as side-chain densities and ligand binding sites [66, 59], potentially leading to misinterpretation of structural and functional characteristics [65].

Denoising cryo-EM density maps aims to mitigate these experimental artifacts, significantly enhancing map interpretability by clarifying the three-dimensional arrangement of atoms. This improved clarity is essential for understanding the functional properties of imaged biomolecules. Moreover, noise reduction substantially facilitates downstream atomic structure modeling tasks, benefiting both manual interpretation and automated approaches such as Phenix [29], MAINMAST [26], Cryo2Struct [72] (discussed in chapter 4), DeepTracer [56], DeepMainmast [92], and ModelAngelo [58]. Recent advances in computational methods have shown promise in addressing these challenges,

with deep learning approaches demonstrating particular potential for distinguishing signal from noise in complex density distributions [70, 139, 140, 141].

While various methods have been developed for denoising cryo-EM density maps, the field lacks standardized, and open-source datasets for advancement of artificial intelligence-based denoising approaches. This deficiency is particularly problematic given the rapid proliferation of AI-based tools in structural biology, as it limits rigorous comparison between methods and hinders benchmarking of new approaches. Furthermore, the absence of standardized test cases makes it difficult to assess how denoising algorithms perform across diverse structural classes and resolution ranges.

As cryo-EM technology continues its rapid expansion in structural biology, with applications extending to increasingly complex systems such as membrane proteins, large macromolecular assemblies, and heterogeneous samples, there is a need for benchmark datasets that bridge disciplinary boundaries and enable AI practitioners to develop innovative methods for enhancing map quality and interpretability. The dataset presented in this manuscript addresses this need by providing a comprehensive and standardized resource for the development and evaluation of cryo-EM denoising algorithms. By preparing this dataset, we aim to drive methodological innovation at the intersection of artificial intelligence and structural biology. Our goal is to accelerate progress in cryo-EM density map interpretation and, ultimately, to advance biomedical research through improved macromolecular structure determination from cryo-EM.

7.3 Methods

7.3.1 Related works

Recently, several AI-based methods have been developed to enhance the quality and interpretability of cryo-EM density maps, including DeepEMhancer [141], EMReady [139], DeepTracer-Denoising [142], and CryoTEN [140]. A key aspect of training these models is the generation of high-quality label maps. Existing approaches include creating simulated cryo-EM density maps using tools like *pdb2mrc* from the EMAN2 package [60], *pdb2vol* from the Situs package [61], or methods based on Rosetta [143]. These simulated maps, derived from known atomic structures, serve as idealized, noise-free ground truth representations of protein density. Alternatively, some models are trained on label maps generated with LocScale by leveraging atomic models to refine experimental cryo-EM maps [143]. In this work, we introduce an innovative approach for generating high-quality label maps which, when paired with experimental cryo-EM maps, are used to train AI-based models for denoising

and enhancing cryo-EM density maps. Once trained, these models can refine noisy experimental maps, improving their clarity and structural interpretability by reducing artifacts and noise.

7.3.2 Data Acquisition and Preprocessing

Dataset Curation

We curated a dataset of high-resolution cryo-EM density maps for single-particle proteins from the Electron Microscopy Data Bank [6] (EMDB), selecting those with resolutions between 1 and 4 Å. The corresponding atomic biological assembly structures were retrieved from the Protein Data Bank [21] (PDB) and used to generate label maps. We refer to the deposited maps obtained from EMDB as experimental cryo-EM density maps, which contain noise and artifacts that must be identified and removed for effective denoising. To ensure data quality, we applied the following filtering criteria: (1) Removal of maps without a corresponding atomic biological assembly structure in the PDB. (2) Removal of maps missing a resolution value determined by the Fourier Shell Correlation (FSC) 0.143 cut-off score. (3) Removal of redundant cryo-EM density maps associated with the same atomic biological assembly structure.

The FSC was computed using the *phenix.mtriage* function from Phenix software suite [29] by providing the experimental cryo-EM density map and its associated PDB structure as input. After filtering, we obtained a final dataset of 650 cryo-EM density maps, which we used to generate labels for training, validation, and testing of deep learning models for cryo-EM density map denoising.

Experimental Map Standardization

Experimental cryo-EM density maps in EMDB exhibit significant variations in density values (e.g., [-2.32, 3.91] and [-0.553, 0.762]) and voxel sizes (e.g., ranging from 0.7 Å to 1.6 Å) due to differences in microscope models, electron doses, detectors, and imaging conditions. Since our approach aims to refine cryo-EM density maps at the voxel level through voxel-wise classification or regression models, we standardized all experimental maps to ensure consistency. We standardized the voxel size of all experimental cryo-EM density maps to 1 Å using the resampling function in UCSF ChimeraX [9] . This preprocessing step enables the model to learn meaningful patterns across different cryo-EM density maps, ultimately improving its robustness and accuracy in denoising and structure enhancement.

7.3.3 Label Map Generation Workflow

We generated three types of label maps to train AI-based models for cryo-EM density map enhancement: a regression map capturing density values representing an ideal, noise-free cryo-EM map, and two classification label maps that provide complementary information to improve model learning during training.

Simulated Map Generation

In the first stage of label map generation, we created clean, noise-free cryo-EM density maps from atomic biological PDB structures using the *pdb2vol* utility from the Situs package [61]. This real-space convolution software generates simulated volumetric maps from input atomic structures. The simulated maps were generated with a voxel size of 1 Å, ensuring consistency with the standardized experimental cryo-EM density maps.

Label Map Generation

While simulated maps provide an idealized, artifact-free representation of cryo-EM density, they often lack precise voxel-wise alignment with their corresponding experimental maps. This alignment is important for computing accurate voxel-wise errors during model optimization. To address this, we processed the simulated maps to ensure they match the size, voxel spacing, and voxel-wise alignment of the experimental maps. We created three empty mask maps (meaning, the voxel values are filled with zero), each matching the dimensions of the experimental cryo-EM density map, to store label information. Using atomic coordinates from the PDB structure, we converted Euclidean coordinates into voxel grid indices using the Formula 7.1, ensuring proper mapping between the experimental and simulated maps. To address the challenges in label generation, we created three distinct types of label maps:

- **Regression Label Map:** This map contains continuous density values derived from the simulated cryo-EM density map. At each voxel position corresponding to an atom in the PDB structure, we assign the density value from the simulated map, providing a noise-free target for regression-based learning.
- **Classification Label Map:** This binary classification map distinguishes structural elements from background. We assign a value of 1 to voxels corresponding to atomic positions, while background voxels remain at 0.

- **Atom Type Classification Label Map:** This multi-class map differentiates between atom types, enabling the model to learn chemical specificity. We assign distinct values to voxels based on the corresponding atom type: Cα (1), Cβ (2), carbonyl carbon (3), oxygen (4), and nitrogen (5).

A critical consideration in our label generation approach is the treatment of neighboring voxels. Due to the inherent discrepancy between continuous atomic coordinates and discrete voxel grid positions, simply labeling the exact voxel corresponding to each atom would result in sparse and potentially imprecise labels. Furthermore, electron density in cryo-EM maps extends beyond the exact atomic positions, forming a cloud-like distribution that represents the probability of electron occurrence.

$$i = \lceil \left(\frac{\lfloor (z - origin_z) }{voxel_z} \right) ; j = \lceil \left(\frac{\lfloor (y - origin_y) }{voxel_y} \right) ; k = \lceil \left(\frac{\lfloor (x - origin_x) }{voxel_x} \right) \tag{7.1}$$

To address these issues and improve the structural context, we extended the labeled regions by identifying neighboring voxels within a 6 Å radius of each atomic coordinate. This radius was selected based on analysis of electron density distribution in high-resolution cryo-EM maps. This is necessary to enhance interpretability by providing meaningful structural context beyond atomic locations and to minimize precision loss when converting atomic coordinates (continuous space) into voxel grid indices (discrete space) as shown in Formula 7.1. The approach for handling these neighboring voxels varies by label map type:

- For the **Regression Label Map**, neighboring voxels are assigned corresponding values from the simulated cryo-EM density map, creating a continuous gradient of density values that mimics the electron cloud surrounding each atom.

- For the **Classification Label Map**, neighboring voxels are assigned a distinct value of 2, differentiating them from both the central atomic positions (value 1) and the background (value 0). This three-class approach allows the model to learn regions between atoms and background.

- For the **Atom Type Classification Label Map**, we maintain the original atomic type labeling only for voxels corresponding directly to atom positions, as the chemical identity of neighboring voxels cannot be unambiguously determined.

By providing complementary information across different label types, we enable deep learning models to learn diverse aspects of molecular structure, ultimately leading to more robust and accurate

enhancement of experimental cryo-EM density maps. These label maps can be used individually or in combination during model training, allowing for flexible architecture design and task-specific optimization strategies.

7.4 Data Records

Our dataset consists of the following components, curated to support the development and validation of deep learning models for cryo-EM density map enhancement:

Experimental Cryo-EM Density Maps

We compiled 650 high-resolution (1-4 Å) experimental maps from EMDB, standardized to 1 Å voxel size. These maps represent diverse structural classes including soluble proteins, membrane proteins, protein-nucleic acid complexes, and macromolecular assemblies. Each map is stored in MRC format with associated metadata including resolution and PDB accession codes.

Atomic Biological Assembly Structures

For each experimental map, we obtained the corresponding biological assembly atomic structure from PDB. These complete assemblies ensure full occupancy of the experimental density maps and serve as the foundation for both label generation and quality validation.

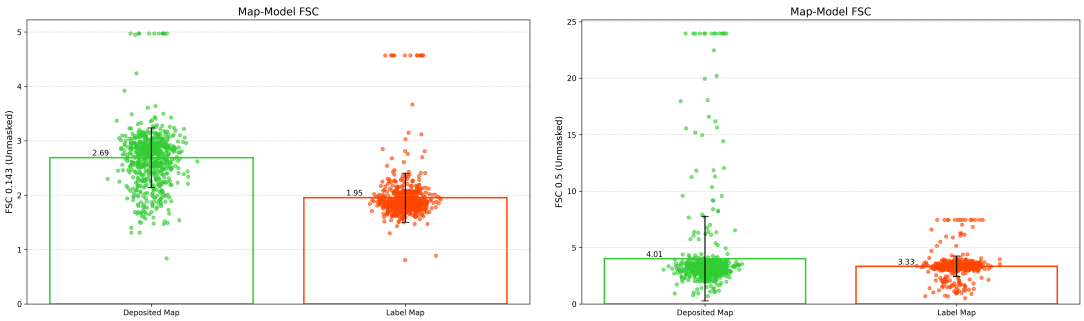
Simulated Cryo-EM Density Maps

Using the *pdb2vol* utility from Situs, we generated idealized, noise-free density maps from the atomic structures. These simulated maps serve as intermediates in our label generation process and are created with parameters matched to their experimental counterparts.

Label Maps

Based on the atomic structures and simulated maps, we generated three types of label maps:

- Regression Label Maps: Contain continuous density values derived from simulated maps, spatially aligned to the experimental maps. These maps represent idealized electron density distributions and serve as primary training targets.
- Classification Label Maps: Differentiate protein structure from background using a three-value system: atomic positions (1), neighboring regions within 6 Å (2), and background (0).



(a) (b)

Figure 7.1: (a) The mean Fourier Shell Correlation (FSC) 0.143 unmasked for deposited map and regression label map is 2.69 Å and 1.95 Å, respectively. (b) The mean FSC 0.5 unmasked for deposited map and regression label map is 4.01 Å and 3.33 Å, respectively.

- Atom Type Classification Label Maps: Label chemical information by assigning values based on atom type: Cα (1), Cβ (2), carbonyl carbon (3), oxygen (4), and nitrogen (5).

All data records maintain consistent dimensions and spatial alignment with their corresponding experimental maps, ensuring proper voxel-to-voxel correspondence for training deep learning models in supervised learning manner. The dataset is organized by PDB identifiers, with supplementary metadata files providing resolution information and quality metrics.

7.5 Technical Validation

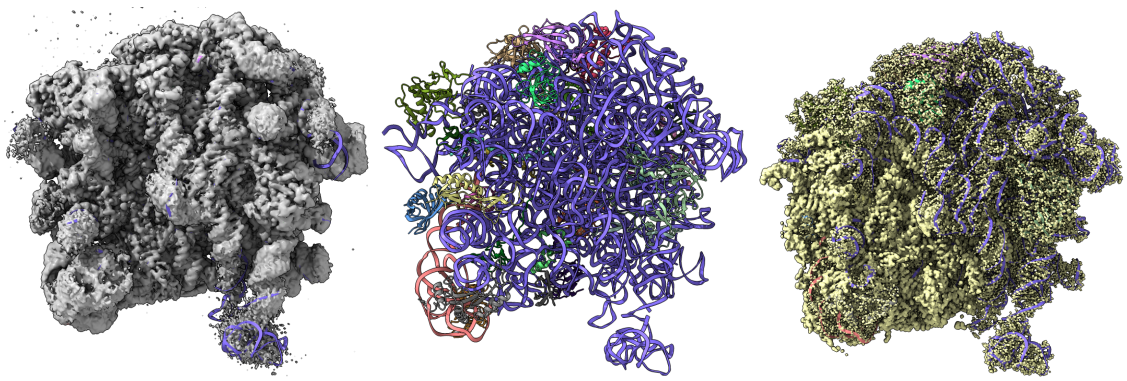
To objectively validate the quality of our regression label maps compared to the experimental maps, we used Fourier Shell Correlation (FSC), the gold standard for resolution assessment in cryo-EM. FSC quantifies the normalized cross-correlation between two 3D volumes as a function of spatial frequency, providing an objective measure of similarity between maps and their corresponding atomic models. For each map in our dataset (n = 650), we computed FSC using the *phenix.mtriage* tool from the Phenix software suite [29]. We analyzed both the standard FSC 0.143 criterion (widely accepted as the resolution threshold for cryo-EM maps) and the more stringent FSC 0.5 criterion (indicating higher confidence in map-model agreement), using unmasked analyses to evaluate global map quality.

Figure 7.1 presents a comprehensive scatter plot comparison of resolution metrics between the deposited experimental maps and our regression label maps. As shown in Figure 7.1 (a), the FSC 0.143 unmasked resolution values for the regression label maps (orange, mean = 1.95 Å) are better than those for the deposited experimental maps (green, mean = 2.69 Å). This represents a substantial

27.5% improvement in resolution. The clustering of the orange points around the mean indicates consistent quality enhancement across the dataset, with notably fewer outliers compared to the experimental maps. Similarly, Figure 7.1 (b) illustrates the comparison using the more stringent FSC 0.5 criterion. The mean unmasked FSC 0.5 resolution improved from 4.01 Å for experimental maps to 3.33 Å for regression label maps, demonstrating a 16.9% enhancement. The distribution pattern shows that regression label maps (orange) maintain more consistent quality at this higher confidence threshold, with experimental maps (green) showing variability.

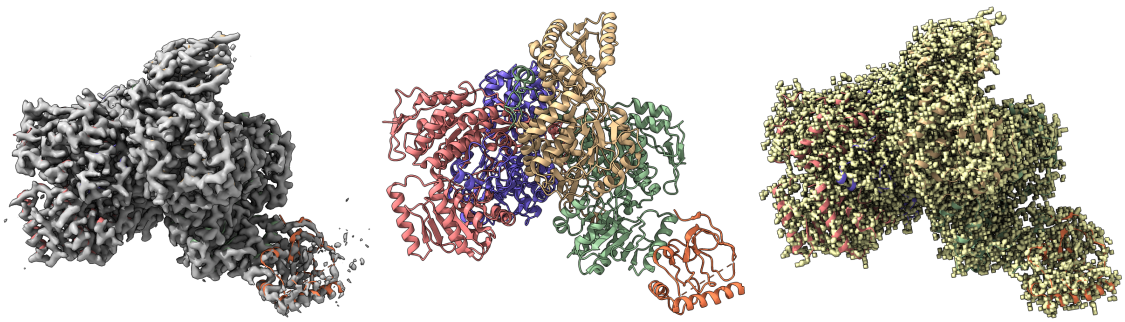
The box plots in both panels (Figure 7.1(a) and (b)) highlight the statistical significance of these improvements, with minimal overlap between the distributions. Notably, the upper quartile bound for the regression label maps falls below the mean of the experimental maps in both FSC metrics, emphasizing the consistent better resolution of the label maps.

Figure 7.2, 7.3, 7.4, 7.5, 7.6, 7.7 shows the experimental cryo-EM density map and the regression labeled cryo-EM density map for visual interpretation. These results provide evidence that our regression label maps offer substantially improved structural representation compared to the original experimental maps. The enhanced resolution and consistency make them ideal targets for training deep learning models aimed at denoising and refining experimental cryo-EM density maps. By learning to transform noisy experimental data toward these higher-quality representations, our models can effectively enhance structural interpretability and in turn more accurate *de novo* atomic structure modeling.



(a) (b) (c)

Figure 7.2: (a) Overlay of the deposited experimental cryo-EM density map (EMD-11900) in grey with dimensions of $308 \times 308 \times 308$ and a voxel size of $1 \times 1 \times 1$ Å, visualized at the recommended contour level of 0.0037 (1.1 σ), along with its corresponding biological atomic structure (PDB Code: 7ASM). The Fourier Shell Correlation (FSC) at 0.5 (unmasked) is 2.43 Å. (b) The atomic structure of the protein (PDB Code: 7ASM). (c) The regression label map in yellow with dimensions of $308 \times 308 \times 308$ and a voxel size of $1 \times 1 \times 1$ Å, overlaid with the known biological atomic structure (PDB Code: 7ASM), with an FSC at 0.5 (unmasked) of 1.27 Å.



(a) (b) (c)

Figure 7.3: (a) Overlay of the deposited experimental cryo-EM density map (EMD-33113) in grey with dimensions of $283 \times 283 \times 283$ and a voxel size of $1 \times 1 \times 1$ Å, visualized at the recommended contour level of 0.0178 (4.9 σ), along with its corresponding biological atomic structure (PDB Code: 7XC6). The Fourier Shell Correlation (FSC) at 0.5 (unmasked) is 4.16 Å. (b) The atomic structure of the protein (PDB Code: 7XC6). (c) The regression label map in yellow with dimensions of $283 \times 283 \times 283$ and a voxel size of $1 \times 1 \times 1$ Å, overlaid with the known biological atomic structure (PDB Code: 7XC6), with an FSC at 0.5 (unmasked) of 3.24 Å.

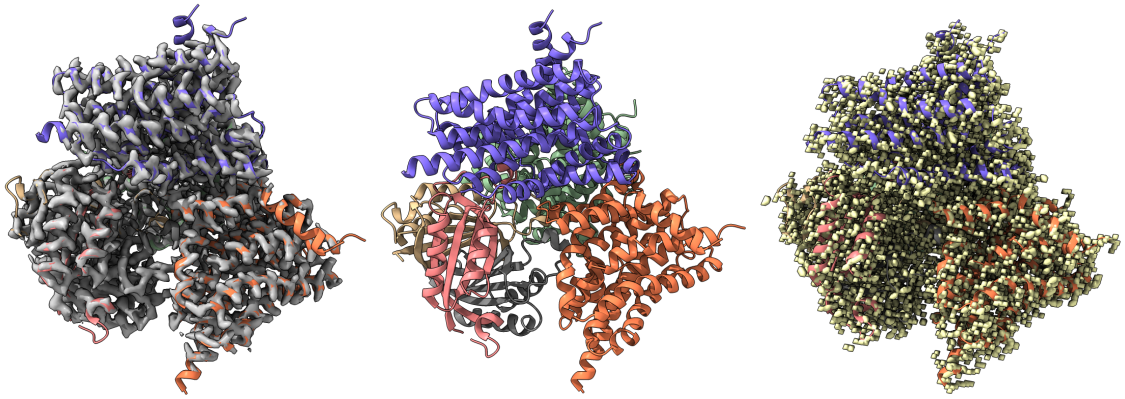
7.6 Usage Notes

The dataset provided in this work is specifically designed for training and evaluating deep learning models aimed at improving the clarity and interpretability of cryo-EM density maps. By enhancing cryo-EM map quality, these models, in turn, facilitate more accurate 3D atomic structure modeling from experimental cryo-EM density maps [70].

The dataset supports the development of various deep learning models for denoising cryo-EM density maps. It can be used for a supervised denoising approach by leveraging pairs of experimental cryo-EM density maps and regression labels to remove noise while preserving structural features. The classification labels allow for semantic segmentation, enabling models to distinguish protein structural components from the background. Additionally, the atom-type classification labels can be used to develop models capable of identifying specific atoms in the cryo-EM density maps.

For optimal results when training deep learning models with this dataset, we recommend that users split the dataset based on resolution, as described in Cryo2StructData [88] (chapter 3), rather than using a random split. To improve memory efficiency, an alternative approach is to generate 3D sub-cubes (e.g., $64 \times 64 \times 64$) from the cryo-EM maps instead of processing entire volumes at once. We also encourage users to consider a multi-task learning approach, training on multiple label types to enhance feature learning and generalization similar to Cryo2Struct2 (chapter 5).

By following these guidelines, researchers can effectively leverage this dataset to develop robust deep learning models for cryo-EM map enhancement, ultimately advancing structural biology research



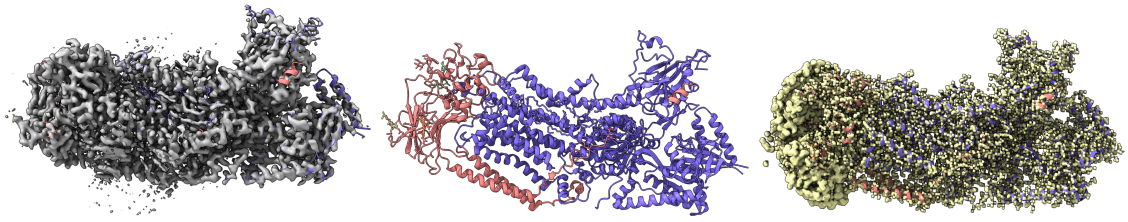
(a) (b) (c)

Figure 7.4: (a) Overlay of the deposited experimental cryo-EM density map (EMD-31135) in grey with dimensions of $258 \times 258 \times 258$ and a voxel size of $1 \times 1 \times 1$ Å, visualized at the recommended contour level of 0.05 (11.5 σ), along with its corresponding biological atomic structure (PDB Code: 7EGK). The Fourier Shell Correlation (FSC) at 0.5 (unmasked) is 7.93 Å. (b) The atomic structure of the protein (PDB Code: 7EGK). (c) The regression label map in yellow with dimensions of $258 \times 258 \times 258$ and a voxel size of $1 \times 1 \times 1$ Å, overlaid with the known biological atomic structure (PDB Code: 7EGK) with an FSC at 0.5 (unmasked) of 3.23 Å.

through improved map interpretation and atomic modeling.

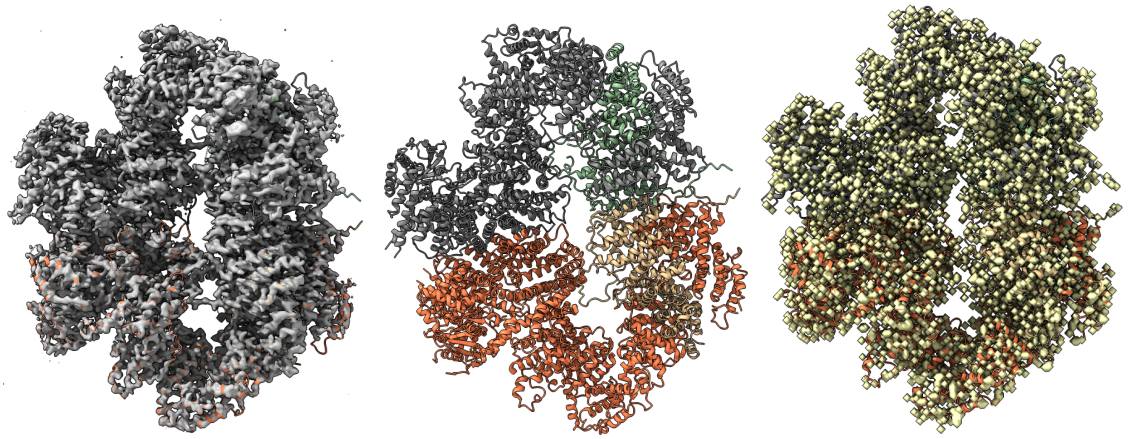
7.7 Code Availability

The source code and the instructions to reproduce the dataset is provided in the GitHub repository accessible at https://github.com/BioinfoMachineLearning/denoisecryodata.git. To keep the generated data files permanent, we published all data to the Harvard Dataverse (https://doi.org/10.7910/DVN/CI0J2B), an online data management and sharing platform with a permanent Digital Object Identifier number.



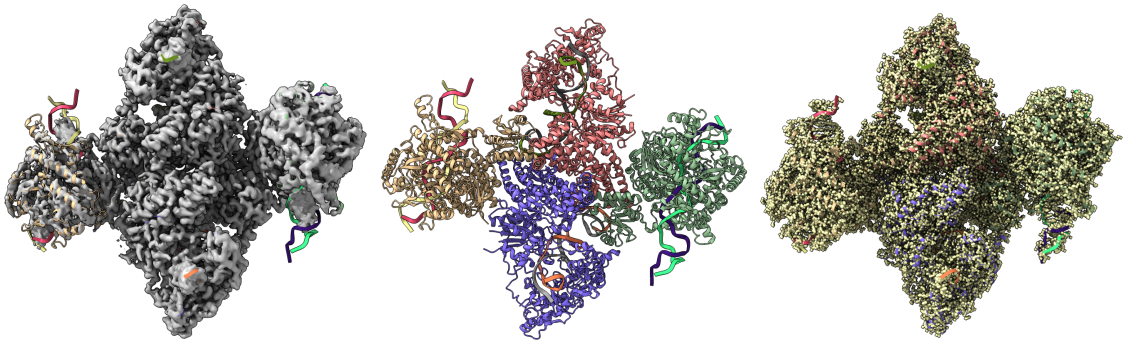
(a) (b) (c)

Figure 7.5: (a) Overlay of the deposited experimental cryo-EM density map (EMD-23075) in grey with dimensions of $231 \times 231 \times 231$ and a voxel size of $1 \times 1 \times 1$ Å, visualized at the recommended contour level of 0.018 (4.3 σ), along with its corresponding atomic structure (PDB Code: 7KYC). The Fourier Shell Correlation (FSC) at 0.5 (unmasked) is 2.86 Å. (b) The atomic structure of the protein (PDB Code: 7KYC). (c) The regression label map in yellow with dimensions of $231 \times 231 \times 231$ and a voxel size of $1 \times 1 \times 1$ Å, overlaid with the known biological atomic structure (PDB Code: 7KYC) with an FSC at 0.5 (unmasked) of 1.56 Å.



(a) (b) (c)

Figure 7.6: (a) Overlay of the deposited experimental cryo-EM density map (EMD-11055) in grey with dimensions of $347 \times 347 \times 347$ and a voxel size of $1 \times 1 \times 1$ Å, visualized at the recommended contour level of 1.6 (4.9 σ), along with its corresponding biological atomic structure (PDB Code: 6Z2W). The Fourier Shell Correlation (FSC) at 0.5 (unmasked) is 6.32 Å. (b) The atomic structure of the protein (PDB Code: 6Z2W). (c) The regression label map in yellow with dimensions of $347 \times 347 \times 347$ and a voxel size of $1 \times 1 \times 1$ Å, overlaid with the known biological atomic structure (PDB Code: 6Z2W) with an FSC at 0.5 (unmasked) of 3.32 Å.



(a) (b) (c)

Figure 7.7: (a) Overlay of the deposited experimental cryo-EM density map (EMD-23461) in grey with dimensions of $526 \times 526 \times 526$ and a voxel size of $1 \times 1 \times 1$ Å, visualized at the recommended contour level of 0.18 (8.2 σ), along with its corresponding biological atomic structure (PDB Code: 7LO5). The Fourier Shell Correlation (FSC) at 0.5 (unmasked) is 6.52 Å. (b) The atomic structure of the protein (PDB Code: 7LO5). (c) The regression label map in yellow with dimensions of $526 \times 526 \times 526$ and a voxel size of $1 \times 1 \times 1$ Å, overlaid with the known biological atomic structure (PDB Code: 7LO5) with an FSC at 0.5 (unmasked) of 3.5 Å.

Chapter 8

UTILIZING THE DEVELOPED TOOLS

8.1 Usage Instructions for Cryo2StructData

Cryo2StructData is a dataset for AI-based modeling of protein structures from cryo-EM density maps. The source codes for generating this dataset are included in the GitHub repository for users to reproduce the process of generating data or create customized datasets.

8.1.1 Dataset Download

To keep the data files of Cryo2StructData permanent, we published all the generated data to the Harvard Dataverse accessible at https://dataverse.harvard.edu/dataverse/Cryo2StructData. Harvard Dataverse is an online data management and sharing platform with a permanent Digital Object Identifier number for each dataset.

The Cryo2StructData Dataverse comprises the Full Cryo2StructData (https://doi.org/10.7910/DVN/FCDG0W) along with its associated trained deep transformer model and data split (https://doi.org/10.7910/DVN/SXNYRE). Similarly, within the Cryo2StructData Dataverse, you will find the Small Subsample (https://doi.org/10.7910/DVN/CGUENL) of the complete Cryo2StructData companied by its respective trained deep transformer model and data splits (https://doi.org/10.7910/DVN/DTV4JF). Finally, the test dataset can be access here: https://doi.org/10.7910/DVN/2GSSC9. The Full and Small Subsample datasets are used for training and validating the deep learning models, whereas the test dataset, which is not included in training or validation, is used to assess the model's performance.

8.1.2 Dataset Access and Structure

The dataset can be accessed using the above dataset download link. The protein structures and cryo-EM density maps can be visualized using tools such as UCSF ChimeraX [9]. The dataset follows the format described below. Please refer to the subsequent section for descriptions of these data files.

Dataset Structure

```
cryo2struct
|-- metadata.csv
|-- splits.csv
|-- root
    |-- full_dataset
        |-- EMD_0
           |-- 0004
              |-- emd_0004.map
              |-- emd_normalized_map.mrc
              |-- atom_emd_normalized_map.mrc
              |-- atom_ca_emd_normalized_map.mrc
              |-- amino_emd_normalized_map.mrc
              |-- sec_struc_emd_normalized_map.mrc
              |-- 6giq.pdb
              |-- 6giq_helix.pdb
              |-- 6giq_coil.pdb
              |-- 6giq_strand.pdb
              |-- 6giq.fasta
              |-- 6giq_all_chain_combined.fasta
              |-- atomic.fasta
              |-- dealign_clustal_input.fasta
              |-- dealign_clustal_output.fasta
           |-- 0031
              |-- emd_0031.map
              |-- ...
        |-- EMD_1
```

```
    |-- 11150
      |-- emd_11150.map
      |-- ...
   |-- 10040
      |-- emd_10040.map
      |-- ...
|-- EMD_2
   |-- 20060
      |-- emd_20060.map
      |-- ...
|-- EMD_3
   |-- 3099
     |-- emd_3099.map
     |-- ...
|-- ...
```

Metadata Information

In the Cryo2StructData Dataverse, an Excel sheet named `metadata.csv` contains relevant information for each cryo-EM density map in the Cryo2StructData dataset. Each row of the sheet includes:

- The EMD ID of the density map.
- The corresponding PDB code.
- The density map resolution.
- The structure determination method.
- The software used for determining the density map.
- The title and journal of the article describing the density maps.

Full Dataset Directory Structure

The full Cryo2StructData directory (https://doi.org/10.7910/DVN/FCDG0W) contains 10 subdirectories, corresponding to 10 folds of the curated training density maps (`EMD_0`, ..., `EMD_9`). As shown in the example data format, each sub-directory for a cryo-EM density map contains the following files:

- `emd_0004.map`: Deposited cryo-EM density map in EMDB [6] (EMD ID: 0004).
- `emd_normalized_map.mrc`: Normalized cryo-EM density map.
- `atom_emd_normalized_map.mrc`: Atom-labeled cryo-EM density map. Users can generate this labeled map using the provided scripts.
- `atom_ca_emd_normalized_map.mrc`: Carbon-alpha (Cα) atom-labeled cryo-EM density map. Users can generate this labeled map using the provided scripts.
- `amino_emd_normalized_map.mrc`: Amino acid-labeled cryo-EM density map. Users can generate this labeled map using the provided scripts.
- `sec_struc_emd_normalized_map.mrc`: Secondary structure-labeled cryo-EM density map. Users can generate this labeled map using the provided scripts.
- `6giq.pdb`: PDB file of the cryo-EM density. Downloaded from PDB [21].
- `6giq_helix.pdb`: Extracted helices from the PDB file.
- `6giq_coil.pdb`: Extracted coils from the PDB file.
- `6giq_strand.pdb`: Extracted strands from the PDB file.
- `6giq.fasta`: Original protein sequence in FASTA format.
- `6giq_all_chain_combined.fasta`: Combined sequence of all chains in FASTA format.
- `atomic.fasta`: Sequence extracted from the PDB structure in FASTA format.
- `dealign_clustal_input.fasta`: Input file for Clustal Omega [78] tool.
- `dealign_clustal_output.fasta`: Alignment between the original sequence and the sequence extracted from the protein structure, generated using Clustal Omega.

8.1.3 Code Repository

The codes are available in the GitHub repository accessible here :
https://github.com/BioinfoMachineLearning/cryo2struct.

8.1.4 Programs to Generate the Dataset

The following instructions provide a brief overview of the dataset generation process. More detailed instructions can be found in the `README.md` file in the GitHub repository.

Setup Environment

To generate the dataset, please set up the environment using Anaconda. Run the below to set up a working `conda environment`: (*cryo2struct.yml* is available in GitHub repository):

```
conda env create -f cryo2struct.yml
conda activate cryo2struct
```

Additionally, the pipeline requires UCSF ChimeraX to resample the density map and extract secondary structures from the `.pdb` file in non-GUI mode. We used ChimeraX 1.4-1 on a CentOS 8 system. To extract the data, run the command below from the preprocessing directory. In the bash script, include the names of the density maps you want to download from the EMDB website. The Python scripts download PDB and FASTA files from the RCSB website.

```
bash fetch_EMDB_maps.sh
python3 get_pdb_from_rcsb.py
python3 get_fasta_from_rcsb.py
```

Run Resample Map Program

```
python3 get_resample_map.py <absolute input path> <chimera_path>
```

The absolute input path is the directory where cryo-EM density maps are present (e.g., path to `EMD_0`). `chimera_path` is optional. If the UCSF ChimeraX path differs from `/usr/bin/chimerax`, please enter the correct path.

Run Normalize Map Program

```
python3 get_normalize_map.py <absolute input path>
```

Normalization requires the resampled map.

Run Labeling Programs

```
python3 get_atoms_label.py <absolute input path>
python3 get_amino_labels.py <absolute input path>
python3 get_secondary_pdb.py <absolute input path>
python3 get_sec_stru_coil_label.py <absolute input path>
python3 get_sec_stru_helix_label.py <absolute input path>
```

```
python3 get_sec_stru_strand_label.py <absolute input path>
```

Run these programs in sequence as they modify the same MRC file.

Extract Sequence from PDB structure and Align with FASTA Sequence

```
python3 merge_chains_fasta.py <absolute input path>
python3 get_pdb_seq.py <absolute input path>
python3 make_input_run_clustal.py <absolute input path>
```

Ensure the programs are run in sequence.

Grid Division of Density Maps

```
python3 grid_division.py <absolute input path>
```

A new directory will be created for grid division results, which will be used in training and inference.

Run All at Once

To run the entire data preparation pipeline at once:

```
bash run_data_preparation.bash ../example
```

In the above example `../example` is the absolute input path.

8.2 Usage Instructions for Cryo2Struct

Cryo2Struct is a fully automated ab initio cryo-EM structure modeling method that first employs a 3D transformer-based model to identify atoms and amino acid types in cryo-EM density maps. It then utilizes an innovative Hidden Markov Model (HMM) to connect predicted atoms, building the backbone structures of proteins.

8.2.1 Code Repository

The codes are available in the GitHub repository accessible here :
https://github.com/jianlin-cheng/Cryo2Struct.

Programs for Atomic Structure Modeling using Cryo2Struct

The following instructions provide a brief overview of atomic structure modeling using Cryo2Struct. More detailed instructions can be found in the `README.md` file in the GitHub repository.

Clone Cryo2Struct Repository

```
git clone https://github.com/jianlin-cheng/Cryo2Struct.git
cd ./Cryo2Struct
```

Setup Environment

Please set up the environment using Anaconda. Run the below to set up a working `conda environment`: (*cryo2struct.yml* is available in GitHub repository):

```
conda env create -f cryo2struct.yml
conda activate cryo2struct
```

Input: Cryo-EM Density Map and Sequence

First, you need to prepare your own data or use our provided example data. The directory should be organized as follows:

```
cryo2struct
|-- input
    |-- 34610
```

```
        |-- emd_34610.map
        |-- 8hb0.fasta
```

The `emd_34610.map` is the density map with EMD ID: 34610, downloaded from the EMDB website [6]. The `8hb0.fasta` is the corresponding sequence file. The first step is to make the input cryo-EM map ready for Cryo2Struct. We ran UCSF ChimeraX in non-GUI mode to resample the density map to 1 Angstrom. Please install UCSF ChimeraX to preprocess the map. We used ChimeraX 1.4-1 on a CentOS 8 system. Once ChimeraX is installed, run the following command:

```
bash preprocess/run_data_preparation.bash input/
```

In the above example, `input/` is the absolute input path where the maps are located. After data preprocessing, the directory structure for this example looks like this:

```
cryo2struct
|-- input
    |-- 34610
        |-- emd_34610.map
        |-- emd_normalized_map.mrc
        |-- 8hb0.fasta
```

Running Cryo2Struct

The deep learning model requires trained atom and amino acid type models. The trained models are available in the Cryo2Struct Harvard Dataverse. Use the following commands to download the trained models:

```
cd models
wget –O amino_acid_type.ckpt
https://dataverse.harvard.edu/api/access/datafile/8076563
wget –O atom_type.ckpt
https://dataverse.harvard.edu/api/access/datafile/8076564
cd ..
```

The organization of the downloaded models should look like this:

```
cryo2struct
|--input
```

```
    |-- 34610
        |-- emd_34610.map
        |-- emd_normalized_map.mrc
        |-- 8hb0.fasta
|-- models
    |-- amino_acid_type.ckpt
    |-- atom_type.ckpt
    |-- aa_regression_model.pkl
    |-- ca_regression_model.pkl
```

Update the configurations in the `config/arguments.yml` file, especially the input data directory, trained model checkpoint path, and density map name. By default, the program runs inference on the CPU. Running the inference program on the GPU speeds up prediction. To enable GPU processing, modify `infer_run_on` in the configuration file to `gpu` and provide the GPU device ID on `infer_on_gpu` (e.g., `0`).:

Compile Modified Viterbi Algorithm

The Hidden Markov Model-guided carbon-alpha alignment programs are available in the `viterbi/` directory of GitHub repository. The alignment algorithm is written in C++ and needs to be compiled. Use the following command to compile:

```
cd viterbi
g++ -fPIC -shared -o viterbi.so viterbi.cpp -O3
cd ..
```

The HMM alignment program runs on the CPU and is optimized with the `-O3` flag. We tested this compilation on CentOS 7, 8, and AlmaLinux OS 8.8 and 8.9.

Running the Inference Program

Finally, run the following to build the atomic structure:

```
python3 cryo2struct.py --density_map_name 34610
```

Output: Modeled Atomic Structure

The output model is saved in the density map's directory. To visualize the structure, use UCSF ChimeraX. It took 9.19 minutes to model the structure for cryo-EM density map 34610.

Confidence Scores

Cryo2Struct provides a per-residue estimation of confidence within the range of [0, 1] for both carbon-alpha and amino acid type predictions. The score is mapped in the predicted atomic structure. To enable the color spectrum (confidence score) in UCSF ChimeraX, navigate to `Tools > Depiction > Render by Attribute` and select `'bfactor'` as the attribute.

8.3 Usage Instructions for Cryo2Struct2

Cryo2Struct2 is a fully automated method for modeling 3D atomic structures from cryo-EM density maps, building on its predecessor, Cryo2Struct. It employs a multi-task deep learning model that integrates sequence-based features from a Protein Language Model (ESM) [16] with cryo-EM density maps, merging feature representation across modalities. The predicted voxels are then used to construct a Hidden Markov Model (HMM), followed by a customized Viterbi algorithm to align sequences and generate initial protein backbone structures. These backbone models are used as templates for AlphaFold3 [15], which further refines the structures for improved accuracy.

8.3.1 Code Repository

The codes are available in the GitHub repository accessible here :
https://github.com/BioinfoMachineLearning/Cryo2Strut2.

Programs for Atomic Structure Modeling using Cryo2Struct2

The following instructions provide a brief overview of atomic structure modeling using Cryo2Struct2. More detailed instructions can be found in the `README.md` file in the GitHub repository.

Clone Cryo2Struct2 Repository

```
git clone https://github.com/BioinfoMachineLearning/Cryo2Strut2.git
cd ./Cryo2Struct2
```

Setup Environment

Please set up the environment using Anaconda. Run the below to set up a working `conda environment`: (*cryo2struct.yml* is available in GitHub repository):

```
conda env create -f cryo2struct2.yml
conda activate cryo2struct2
```

Input: Cryo-EM Density Map and Sequence

First, you need to prepare your own data or use our provided example data. The directory should be organized as follows:

```
cryo2struct
|-- input
    |-- 34610
        |-- emd_34610.map
        |-- 8hb0.fasta
```

The `emd_34610.map` is the density map with EMD ID: 34610, downloaded from the EMDB website. The `8hb0.fasta` is the corresponding sequence file. The first step is to make the input cryo-EM map ready for Cryo2Struct2. We ran UCSF ChimeraX in non-GUI mode to resample the density map to 1 Angstrom. Please install UCSF ChimeraX to preprocess the map. We used ChimeraX 1.4-1 on a CentOS 8 system. Once ChimeraX is installed, run the following command:

```
bash preprocess/run_data_preparation.bash input/
```

In the above example, `input/` is the absolute input path where the maps are located. After preprocessing, the directory structure for this example looks like this:

```
cryo2struct
|-- input
    |-- 34610
        |-- emd_34610.map
        |-- emd_normalized_map.mrc
        |-- 8hb0.fasta
```

Set Up ESM

Cryo2Struct2 generates features from Evolutionary Scale Modeling (ESM): Pretrained language models for proteins [16]. Set up ESM in your system following the instruction provided in https://github.com/facebookresearch/esm. The esm.pretrained model we used is `esm2_t36_3B_UR50D()`.

Running Cryo2Struct2

The deep learning model requires trained atom and amino acid type models. The trained models are available in the Cryo2Struct2 Harvard Dataverse. Use the following commands to download the trained models:

```
cd models
wget –O amino_acid_type.ckpt
```

```
https://dataverse.harvard.edu/api/access/datafile/10888677
wget –O atom_type.ckpt
https://dataverse.harvard.edu/api/access/datafile/10888678
cd ..
```

The organization of the downloaded models should look like this:

```
cryo2struct
|--input
    |-- 34610
        |-- emd_34610.map
        |-- emd_normalized_map.mrc
        |-- 8hb0.fasta
|-- models
    |-- amino_acid_type.ckpt
    |-- atom_type.ckpt
    |-- aa_regression_model.pkl
    |-- ca_regression_model.pkl
```

Update the configurations in the `config/arguments.yml` file, especially the input data directory, trained model checkpoint path, and density map name. By default, the program runs inference on the CPU. Running the inference program on the GPU speeds up prediction. To enable GPU processing, modify `infer_run_on` in the configuration file to `gpu` and provide the GPU device ID on `infer_on_gpu` (e.g., `0`).:

Compile Modified Viterbi Algorithm

The Hidden Markov Model-guided carbon-alpha alignment programs are available in the `viterbi/` directory. The alignment algorithm is written in C++ and needs to be compiled. Use the following command:

```
cd viterbi
g++ –fPIC –shared –o viterbi.so viterbi.cpp –O3
cd ..
```

The HMM alignment program runs on the CPU and is optimized with the `-O3` flag. We tested this compilation on CentOS 7, 8, and AlmaLinux OS 8.8 and 8.9.

Running the Inference Program

Finally, run the following command to model the atomic structure:

```
python3 cryo2struct2.py --density_map_name 34610
```

Output: Modeled Atomic Structure

The output model is saved in the density map's directory. To visualize the structure, use UCSF ChimeraX [9].

Integrating Cryo2Struct2 Models as Templates for AlphaFold3

The models generated by Cryo2Struct2 are used as templates for AlphaFold3. Use the provided script `prepare_script_af3_multichain_multi_template.py` to generate `.json` files that will be used as input to run AlphaFold3.

Set up AlphaFold3

Request AlphaFold3 parameters and follow the instructions to set up AlphaFold3 from here : https://github.com/google-deepmind/alphafold3.

Run AlphaFold3

Use the script `run_af3_docker_all.py` to run AlphaFold3 and to predict structures.

8.4 Usage Instructions for DeepProLigand

DeepProLigand is a deep learning bioinformatics pipeline developed for modeling protein-ligand interaction with cryo-EM data in 2021 Ligand Model Challenge.

8.4.1 Code Repository

The codes are available in the GitHub repository accessible here :
https://github.com/jianlin-cheng/DeepProLigand.

Programs for Protein-Ligand Modeling Using Cryo-EM Data

The following instructions provide a brief overview of protein-ligand modeling using cryo-EM data. More detailed instructions can be found in the `README.md` file in the GitHub repository.

Clone Cryo2Struct Repository

```
git clone https://github.com/jianlin-cheng/DeepProLigand.git
cd ./DeepProLigand
```

Setup Environment

Please set up the environment using Anaconda. Run the below to set up a working `conda environment`: (*environment.yml* is available in GitHub repository):

```
conda env create -f environment.yml
conda activate deep-pro-ligand
```

Running DeepProLigand

The pipeline makes use of DeepTracer, UCSF Chimera, and PyRosetta. Please run DeepTracer through the webpage and install USCF Chimera and PyRosetta through their websites:

DeepTracer is available at: https://deeptracer.uw.edu/home

UCSF Chimera is available at: https://www.cgl.ucsf.edu/chimera/download.html

PyRosetta is available at: https://www.pyrosetta.org/downloads

Stepwise Procedure:

a) Using the density map as an input, please run DeepTracer to generate the atomic backbone structure of the protein.

b) The output of DeepTracer and reference structure (deposited into PDB) for the density map needs to be superimposed using UCSF Chimera's matchmaker function. The matchmaker function can be enabled from UCSF Chimera application as : `Tools > Structure Comparision > Matchmaker`. Once the structures are superimposed, they need to be saved in a single PDB file. Please find the superimposed structures for all three targets in `/data/` directory of the GitHub repository.

c) The next steps requires PyRosetta to be installed into the system. To generate the average protein structure run as shown below for each density map:

```
python3 avg_7770.py
```

`7770` is the density map's name. For, `30210` density map, run `python3 avg_30210.py`. Similarly, for `22898` density map, run `python3 avg_22898.py`. The outputs are currently saved in `/data/averaged`, output directory can be changed from the program.

d) To generate the refined protein structure run the following:

```
python3 fastrelax.py
```

The outputs of fastrelax program are currently saved in the directory: `/data/refined/`.

8.5 Usage Instructions for Denoise Dataset

An open-source dataset for cryo-EM density map denoising comprising 650 high-resolution experimental maps paired with three types of generated label maps: regression maps capturing idealized density distributions, binary classification maps distinguishing structural elements from background, and atom-type classification maps.

8.5.1 Dataset Download

To keep the data files of denoise dataset permanent, we published all data to the Harvard Dataverse (https://doi.org/10.7910/DVN/CI0J2B), an online data management and sharing platform with a permanent Digital Object Identifier number for each dataset.

8.5.2 Dataset Access and Structure

The dataset can be accessed using the provided dataset download link. The protein structures and cryo-EM density maps can be visualized using tools such as UCSF ChimeraX [9]. The dataset follows the format described below. Please refer to the subsequent section for descriptions of these data files.

```
|-- 6bco
    |-- 6bco.mrc
    |-- 6bco.pdb1
    |-- 6bco_classification_situs.mrc
    |-- 6bco_classification_types_situs.mrc
    |-- 6bco_regression_situs.mrc
    |-- 6bco_situs_simulated.mrc

|-- 7ki6
    |-- 7ki6.mrc
    |-- 7ki6.pdb1
    |-- 7ki6_classification_situs.mrc
    |-- 7ki6_classification_types_situs.mrc
    |-- 7ki6_regression_situs.mrc
    |-- 7ki6_situs_simulated.mrc
```

```
|-- 7o3h
    |-- 7o3h.mrc
    |-- 7o3h.pdb1
    |-- 7o3h_classification_situs.mrc
    |-- 7o3h_classification_types_situs.mrc
    |-- .
    |-- .

.
.
```

As shown in the example data format above, each individual directory for the cryo-EM density map (directory name is the protein PDB Code) provides the following data files:

- `6bco.mrc`: The deposited experimental cryo-EM density map. The filename is the PDB code of the cryo-EM density map. This data is downloaded from EMDB [6].
- `6bco.pdb1`: The atomic biological assembly of the macromolecule downloaded from PDB [21].
- `6bco_situs_simulated.mrc`: The simulated cryo-EM density map of `6bco.pdb1` generated using the Situs package.
- `6bco_classification_situs.mrc`: The classification label map, neighboring voxels are assigned a value of 2, central atomic positions (value 1) and the background (value 0).
- `6bco_classification_types_situs.mrc`: The atom type classification label map, values to voxels based on the corresponding atom type: Cα (1), Cβ (2), carbonyl carbon (3), oxygen (4), and nitrogen (5).
- `6bco_regression_situs.mrc`: This map contains continuous density values derived from the simulated cryo-EM density map. At each voxel position corresponding to an atom in the PDB structure, we assign the density value from the simulated map, providing a noise-free target for regression-based learning.

8.5.3 Code Repository

The codes are available in the GitHub repository accessible here :

https://github.com/BioinfoMachineLearning/denoisecryodata.

8.5.4 Programs to Generate the Dataset

```
python3 generate_labels_all_conv.py
```

Evaluating Map Quality with Phenix mtriage

The below code runs `phenix.mtriage` to compute the FSC scores. The webpage of Phenix mtriage is available here: https://phenix-online.org/documentation/reference/mtriage.html

```
python3 evaluate_phenix_mtriage_exp.py
python3 evaluate_phenix_mtriage_reg.py
python3 evaluate_phenix_mtriage_sim.py
```

Programs to extract the information from Phenix mtriage output

The FSC scores reported in the paper are generated using Phenix mtriage. Use the below to extract information in `.csv` format

```
python3 parse_get_resolution_from_log_file_exp.py
python3 parse_get_resolution_from_log_file_reg.py
python3 parse_get_resolution_from_log_file_sim.py
```

BIBLIOGRAPHY

[1] N. Giri, L. Wang, and J. Cheng, "Cryo2StructData : Full Dataset," 2023. [Online]. Available: https://doi.org/10.7910/DVN/FCDG0W

[2] ——, "Cryo2StructData : Small Subsample Dataset," 2023. [Online]. Available: https://doi.org/10.7910/DVN/CGUENL

[3] ——, "Cryo2StructData : Trained Model and Data Splits (Full)," 2023. [Online]. Available: https://doi.org/10.7910/DVN/SXNYRE

[4] ——, "Cryo2StructData : Trained Model and Data Splits (Small Subset)," 2023. [Online]. Available: https://doi.org/10.7910/DVN/DTV4JF

[5] "How Cryo-Electron Microscopy Can Accelerate Drug Discovery," https://blog.delmic.com/cryo-electron-microscopy-for-drug-discovery, 2025.

[6] "Emdb—the electron microscopy data bank," *Nucleic acids research*, vol. 52, no. D1, pp. D456–D465, 2024.

[7] D. M. Kern, B. Sorum, S. S. Mali, C. M. Hoel, S. Sridharan, J. P. Remis, D. B. Toso, A. Kotecha, D. M. Bautista, and S. G. Brohawn, "Cryo-em structure of sars-cov-2 orf3a in lipid nanodiscs," *Nature structural & molecular biology*, vol. 28, no. 7, pp. 573–582, 2021.

[8] M. Gui, W. Song, H. Zhou, J. Xu, S. Chen, Y. Xiang, and X. Wang, "Cryo-electron microscopy structures of the sars-cov spike glycoprotein reveal a prerequisite conformational state for receptor binding," *Cell research*, vol. 27, no. 1, pp. 119–129, 2017.

[9] E. F. Pettersen, T. D. Goddard, C. C. Huang, E. C. Meng, G. S. Couch, T. I. Croll, J. H. Morris, and T. E. Ferrin, "Ucsf chimerax: Structure visualization for researchers, educators, and developers," *Protein science*, vol. 30, no. 1, pp. 70–82, 2021.

[10] P. Mostosi, H. Schindelin, P. Kollmannsberger, and A. Thorn, "Haruspex: a neural network for the automatic identification of oligonucleotides and protein secondary structure in cryo-electron

microscopy maps," *Angewandte Chemie International Edition*, vol. 59, no. 35, pp. 14 788–14 795, 2020.

[11] J. He and S.-Y. Huang, "Emnuss: a deep learning framework for secondary structure annotation in cryo-em maps," *Briefings in bioinformatics*, vol. 22, no. 6, p. bbab156, 2021.

[12] D. M. Kern, B. Sorum, S. S. Mali, C. M. Hoel, S. Sridharan, J. P. Remis, D. B. Toso, A. Kotecha, D. M. Bautista, and S. G. Brohawn, "Cryo-em structure of sars-cov-2 orf3a in lipid nanodiscs," *Nature structural & molecular biology*, vol. 28, no. 7, pp. 573–582, 2021.

[13] J. W. Saville, D. Mannar, X. Zhu, S. S. Srivastava, A. M. Berezuk, J.-P. Demers, S. Zhou, K. S. Tuttle, I. Sekirov, A. Kim *et al.*, "Structural and biochemical rationale for enhanced spike protein fitness in delta and kappa sars-cov-2 variants," *Nature communications*, vol. 13, no. 1, p. 742, 2022.

[14] S. Banerjee, A. Bartesaghi, A. Merk, P. Rao, S. L. Bulfer, Y. Yan, N. Green, B. Mroczkowski, R. J. Neitz, P. Wipf, V. Falconieri, R. J. Deshaies, J. L. S. Milne, D. Huryn, M. Arkin, and S. Subramaniam, "2.3 Å resolution cryo-em structure of human p97 and mechanism of allosteric inhibition," *Science*, vol. 351, no. 6275, pp. 871–875, 2016. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.aad7974

[15] J. Abramson, J. Adler, J. Dunger, R. Evans, T. Green, A. Pritzel, O. Ronneberger, L. Willmore, A. J. Ballard, J. Bambrick *et al.*, "Accurate structure prediction of biomolecular interactions with alphafold 3," *Nature*, pp. 1–3, 2024.

[16] Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, A. dos Santos Costa, M. Fazel-Zarandi, T. Sercu, S. Candido *et al.*, "Language models of protein sequences at the scale of evolution enable accurate structure prediction," *bioRxiv*, 2022.

[17] M. Baek, F. DiMaio, I. Anishchenko, J. Dauparas, S. Ovchinnikov, G. R. Lee, J. Wang, Q. Cong, L. N. Kinch, R. D. Schaeffer *et al.*, "Accurate prediction of protein structures and interactions using a three-track neural network," *Science*, vol. 373, no. 6557, pp. 871–876, 2021.

[18] A. Dhakal, R. Gyawali, L. Wang, and J. Cheng, "Artificial intelligence in cryo-em protein particle picking: recent advances and remaining challenges," *Briefings in Bioinformatics*, vol. 26, no. 1, p. bbaf011, 2025.

[19] A. Singer and F. J. Sigworth, "Computational methods for single-particle electron cryomicroscopy," *Annual review of biomedical data science*, vol. 3, no. 1, pp. 163–190, 2020.

[20] W. Kühlbrandt, "The resolution revolution," *Science*, vol. 343, no. 6178, pp. 1443–1444, 2014. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.1251652

[21] S. K. Burley, H. M. Berman, G. J. Kleywegt, J. L. Markley, H. Nakamura, and S. Velankar, "Protein data bank (pdb): the single global macromolecular structure archive," *Protein crystallography: methods and protocols*, pp. 627–641, 2017.

[22] S. Lindert, N. Alexander, N. Wötzel, M. Karakaş, P. L. Stewart, and J. Meiler, "Em-fold: de novo atomic-detail protein structure determination from medium-resolution density maps," *Structure*, vol. 20, no. 3, pp. 464–478, 2012.

[23] M. L. Baker, S. S. Abeysinghe, S. Schuh, R. A. Coleman, A. Abrams, M. P. Marsh, C. F. Hryc, T. Ruths, W. Chiu, and T. Ju, "Modeling protein structure at near atomic resolutions with gorgon," *Journal of structural biology*, vol. 174, no. 2, pp. 360–373, 2011.

[24] F. DiMaio, A. Leaver-Fay, P. Bradley, D. Baker, and I. André, "Modeling symmetric macromolecular structures in rosetta3," *PloS one*, vol. 6, no. 6, p. e20450, 2011.

[25] M. Chen, P. R. Baldwin, S. J. Ludtke, and M. L. Baker, "De novo modeling in cryo-em density maps with pathwalking," *Journal of structural biology*, vol. 196, no. 3, pp. 289–298, 2016.

[26] G. Terashi and D. Kihara, "De novo main-chain modeling for em maps using mainmast," *Nature communications*, vol. 9, no. 1, p. 1618, 2018.

[27] G. Terashi, Y. Kagaya, and D. Kihara, "Mainmastseg: automated map segmentation method for cryo-em density maps with symmetry," *Journal of chemical information and modeling*, vol. 60, no. 5, pp. 2634–2643, 2020.

[28] E. Alnabati, G. Terashi, and D. Kihara, "Protein structural modeling for electron microscopy maps using vesper and mainmast," *Current protocols*, vol. 2, no. 7, p. e494, 2022.

[29] D. Liebschner, P. V. Afonine, M. L. Baker, G. Bunkóczi, V. B. Chen, T. I. Croll, B. Hintze, L.-W. Hung, S. Jain, A. J. McCoy *et al.*, "Macromolecular structure determination using x-rays, neutrons and electrons: recent developments in phenix," *Biological Crystallography*, vol. 75, no. 10, pp. 861–877, 2019.

[30] J. G. Greener, S. M. Kandathil, L. Moffat, and D. T. Jones, "A guide to machine learning for biologists," *Nature reviews Molecular cell biology*, vol. 23, no. 1, pp. 40–55, 2022.

[31] L. Ma, M. Reisert, and H. Burkhardt, "Rennsh: A novel\alpha-helix identification approach for intermediate resolution electron density maps," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 9, no. 1, pp. 228–239, 2011.

[32] D. Si, S. Ji, K. A. Nasr, and J. He, "A machine learning approach for the identification of protein secondary structure elements from electron cryo-microscopy density maps," *Biopolymers*, vol. 97, no. 9, pp. 698–708, 2012.

[33] A. Esteva, K. Chou, S. Yeung, N. Naik, A. Madani, A. Mottaghi, Y. Liu, E. Topol, J. Dean, and R. Socher, "Deep learning-enabled medical computer vision. npj digital medicine, 4 (1), 5," *URL: https://doi. org/10*, vol. 1038, 2021.

[34] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko *et al.*, "Highly accurate protein structure prediction with alphafold," *nature*, vol. 596, no. 7873, pp. 583–589, 2021.

[35] D. Si, A. Nakamura, R. Tang, H. Guan, J. Hou, A. Firozi, R. Cao, K. Hippe, and M. Zhao, "Artificial intelligence advances for de novo molecular structure modeling in cryo-electron microscopy," *Wiley Interdisciplinary Reviews: Computational Molecular Science*, vol. 12, no. 2, p. e1542, 2022.

[36] H. Gupta, M. T. McCann, L. Donati, and M. Unser, "Cryogan: A new reconstruction paradigm for single-particle cryo-em via deep adversarial learning," *IEEE Transactions on Computational Imaging*, vol. 7, pp. 759–774, 2021.

[37] E. D. Zhong, T. Bepler, B. Berger, and J. H. Davis, "Cryodrgn: reconstruction of heterogeneous cryo-em structures using neural networks," *Nature methods*, vol. 18, no. 2, pp. 176–185, 2021.

[38] M. Chen and S. J. Ludtke, "Deep learning-based mixed-dimensional gaussian mixture model for characterizing variability in cryo-em," *Nature methods*, vol. 18, no. 8, pp. 930–936, 2021.

[39] H. Lei and Y. Yang, "Cdae: A cascade of denoising autoencoders for noise reduction in the clustering of single-particle cryo-em images," *Frontiers in genetics*, vol. 11, p. 627746, 2021.

[40] D. Kimanius, G. Zickert, T. Nakane, J. Adler, S. Lunz, C.-B. Schönlieb, O. Öktem, and S. H. Scheres, "Exploiting prior knowledge about biological macromolecules in cryo-em structure determination," *IUCrJ*, vol. 8, no. 1, pp. 60–75, 2021.

[41] A. Al-Azzawi, A. Ouadou, H. Max, Y. Duan, J. J. Tanner, and J. Cheng, "Deepcryopicker: fully automated deep neural network for single protein particle picking in cryo-em," *BMC bioinformatics*, vol. 21, pp. 1–38, 2020.

[42] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural computation*, vol. 29, no. 9, pp. 2352–2449, 2017.

[43] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer, 2015, pp. 234–241.

[44] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: A nested u-net architecture for medical image segmentation," in *Deep learning in medical image analysis and multimodal learning for clinical decision support: 4th international workshop, DLMIA 2018, and 8th international workshop, ML-CDS 2018, held in conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, proceedings 4*. Springer, 2018, pp. 3–11.

[45] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[46] A. Sherstinsky, "Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network," *Physica D: Nonlinear Phenomena*, vol. 404, p. 132306, 2020.

[47] D. Si, S. A. Moritz, J. Pfab, J. Hou, R. Cao, L. Wang, T. Wu, and J. Cheng, "Deep learning to predict protein backbone structure from high-resolution cryo-em density maps," *Scientific reports*, vol. 10, no. 1, p. 4282, 2020.

[48] P.-N. Li, S. H. de Oliveira, S. Wakatsuki, and H. van den Bedem, "Sequence-guided protein structure determination using graph convolutional and recurrent networks," in *2020 IEEE 20th international conference on bioinformatics and bioengineering (BIBE)*. IEEE, 2020, pp. 122–127.

[49] L. Chang, F. Wang, K. Connolly, H. Meng, Z. Su, V. Cvirkaite-Krupovic, M. Krupovic, E. H. Egelman, and D. Si, "Deeptracer-id: De novo protein identification from cryo-em maps," *Biophysical Journal*, vol. 121, no. 15, pp. 2840–2848, 2022.

[50] X. Zhang, B. Zhang, P. L. Freddolino, and Y. Zhang, "Cr-i-tasser: assemble protein structures from cryo-em density maps using deep convolutional neural networks," *Nature methods*, vol. 19, no. 2, pp. 195–204, 2022.

[51] J. He, P. Lin, J. Chen, H. Cao, and S.-Y. Huang, "Model building of protein complexes from intermediate-resolution cryo-em maps with deep learning-guided automatic assembly," *Nature Communications*, vol. 13, no. 1, p. 4066, 2022.

[52] S. R. Maddhuri Venkata Subramaniya, G. Terashi, and D. Kihara, "Protein secondary structure detection in intermediate-resolution cryo-em maps using deep learning," *Nature methods*, vol. 16, no. 9, pp. 911–917, 2019.

[53] M. Rozanov and H. J. Wolfson, "Aanchor: Cnn guided detection of anchor amino acids in high resolution cryo-em density maps," in *2018 IEEE international conference on bioinformatics and biomedicine (BIBM)*. IEEE, 2018, pp. 88–91.

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

[55] J. Yang, R. Yan, A. Roy, D. Xu, J. Poisson, and Y. Zhang, "The i-tasser suite: protein structure and function prediction," *Nature methods*, vol. 12, no. 1, pp. 7–8, 2015.

[56] J. Pfab, N. M. Phan, and D. Si, "Deeptracer for fast de novo cryo-em protein structure modeling and special studies on cov-related complexes," *Proceedings of the National Academy of Sciences*, vol. 118, no. 2, p. e2017525118, 2021.

[57] J. He, P. Lin, J. Chen, H. Cao, and S.-Y. Huang, "Model building of protein complexes from intermediate-resolution cryo-em maps with deep learning-guided automatic assembly," *Nature Communications*, vol. 13, no. 1, p. 4066, 2022.

[58] K. Jamali, L. Käll, R. Zhang, A. Brown, D. Kimanius, and S. H. Scheres, "Automated model building and protein identification in cryo-em maps," *Nature*, vol. 628, no. 8007, pp. 450–457, 2024.

[59] N. Giri and J. Cheng, "Improving protein–ligand interaction modeling with cryo-em data, templates, and deep learning in 2021 ligand model challenge," *Biomolecules*, vol. 13, no. 1, p. 132, 2023.

[60] G. Tang, L. Peng, P. R. Baldwin, D. S. Mann, W. Jiang, I. Rees, and S. J. Ludtke, "Eman2: an extensible image processing suite for electron microscopy," *Journal of structural biology*, vol. 157, no. 1, pp. 38–46, 2007.

[61] W. Wriggers, R. A. Milligan, and J. A. McCammon, "Situs: a package for docking crystal structures into low-resolution maps from electron microscopy," *Journal of structural biology*, vol. 125, no. 2-3, pp. 185–195, 1999.

[62] M. M. Bronstein, J. Bruna, T. Cohen, and P. Veličković, "Geometric deep learning: Grids, groups, graphs, geodesics, and gauges," *arXiv preprint arXiv:2104.13478*, 2021.

[63] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[64] F. Fuchs, D. Worrall, V. Fischer, and M. Welling, "Se (3)-transformers: 3d roto-translation equivariant attention networks," *Advances in neural information processing systems*, vol. 33, pp. 1970–1981, 2020.

[65] F. Boadu, H. Cao, and J. Cheng, "Combining protein sequences and structures with transformers and equivariant graph neural networks to predict protein function," *bioRxiv*, pp. 2023–01, 2023.

[66] A. Dhakal, C. McKay, J. J. Tanner, and J. Cheng, "Artificial intelligence in the prediction of protein–ligand interactions: recent advances and future directions," *Briefings in Bioinformatics*, vol. 23, no. 1, p. bbab476, 2022.

[67] X.-C. Bai, G. McMullan, and S. H. Scheres, "How cryo-em is revolutionizing structural biology," *Trends in biochemical sciences*, vol. 40, no. 1, pp. 49–57, 2015.

[68] A. Iudin, P. K. Korir, S. Somasundharam, S. Weyand, C. Cattavitello, N. Fonseca, O. Salih, G. Kleywegt, and A. Patwardhan, "EMPIAR: the Electron Microscopy Public Image Archive," *Nucleic Acids Research*, vol. 51, no. D1, pp. D1503–D1511, 11 2022. [Online]. Available: https://doi.org/10.1093/nar/gkac1062

[69] A. Dhakal, R. Gyawali, L. Wang, and J. Cheng, "A large expert-curated cryo-em image dataset for machine learning protein particle picking," *Scientific Data*, vol. 10, no. 1, p. 392, 2023.

[70] N. Giri, R. S. Roy, and J. Cheng, "Deep learning for reconstructing protein structures from cryo-em density maps: Recent advances and future directions," *Current Opinion in Structural Biology*, vol. 79, p. 102536, 2023.

[71] C. L. Lawson, A. Kryshtafovych, G. D. Pintilie, S. K. Burley, J. Černỳ, V. B. Chen, P. Emsley, A. Gobbi, A. Joachimiak, S. Noreng *et al.*, "Outcomes of the emdataresource cryo-em ligand modeling challenge," *Nature methods*, vol. 21, no. 7, pp. 1340–1348, 2024.

[72] N. Giri and J. Cheng, "De novo atomic protein structure modeling for cryoem density maps using 3d transformer and hmm," *Nature Communications*, vol. 15, no. 1, p. 5511, 2024.

[73] Y. Cheng, N. Grigorieff, P. A. Penczek, and T. Walz, "A primer to single-particle cryo-electron microscopy," *Cell*, vol. 161, no. 3, pp. 438–449, 2015.

[74] N. Giri, L. Wang, and J. Cheng, "Cryo2StructData Metadata," 2023. [Online]. Available: https://doi.org/10.7910/DVN/JMN60H

[75] A. Cheng, R. Henderson, D. Mastronarde, S. J. Ludtke, R. H. Schoenmakers, J. Short, R. Marabini, S. Dallakyan, D. Agard, and M. Winn, "Mrc2014: Extensions to the mrc format header for electron cryo-microscopy and tomography," *Journal of structural biology*, vol. 192, no. 2, pp. 146–150, 2015.

[76] N. Giri, L. Wang, and J. Cheng, "Cryo2StructData : Test Dataset," 2023. [Online]. Available: https://doi.org/10.7910/DVN/2GSSC9

[77] W. R. Pearson and D. J. Lipman, "Improved tools for biological sequence comparison." *Proceedings of the National Academy of Sciences*, vol. 85, no. 8, pp. 2444–2448, 1988.

[78] F. Sievers and D. G. Higgins, "Clustal omega," *Current protocols in bioinformatics*, vol. 48, no. 1, pp. 3–13, 2014.

[79] T. Burnley, C. M. Palmer, and M. Winn, "Recent developments in the *CCP-EM* software suite," *Acta Crystallographica Section D*, vol. 73, no. 6, pp. 469–477, Jun 2017. [Online]. Available: https://doi.org/10.1107/S2059798317007859

[80] L. Rabiner and B. Juang, "An introduction to hidden markov models," *ieee assp magazine*, vol. 3, no. 1, pp. 4–16, 1986.

[81] M. Gao, P. Lund-Andersen, A. Morehead, S. Mahmud, C. Chen, X. Chen, N. Giri, R. S. Roy, F. Quadir, T. C. Effler *et al.*, "High-performance deep learning toolbox for genome-scale prediction of protein structure and function," in *2021 IEEE/ACM Workshop on Machine Learning in High Performance Computing Environments (MLHPC)*. IEEE, 2021, pp. 46–57.

[82] W. Yin, C. Mao, X. Luan, D.-D. Shen, Q. Shen, H. Su, X. Wang, F. Zhou, W. Zhao, M. Gao, S. Chang, Y.-C. Xie, G. Tian, H.-W. Jiang, S.-C. Tao, J. Shen, Y. Jiang, H. Jiang, Y. Xu, S. Zhang, Y. Zhang, and H. E. Xu, "Structural basis for inhibition of the rna-dependent rna polymerase from sars-cov-2 by remdesivir," *Science*, vol. 368, no. 6498, pp. 1499–1504, 2020. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.abc1560

[83] E. Soltanikazemi, R. S. Roy, F. Quadir, N. Giri, A. Morehead, and J. Cheng, "Drlcomplex: Reconstruction of protein quaternary structures using deep reinforcement learning," *arXiv preprint arXiv:2205.13594*, 2022.

[84] D. Cressey and E. Callaway, "Cryo-electron microscopy wins chemistry nobel," *Nature*, vol. 550, no. 7675, 2017.

[85] C. Lawson, A. Kryshtafovych, G. Pintilie, S. Burley, J. Cerny, V. Chen, P. Emsley, A. Gobbi, A. Joachimiak, S. Noreng *et al.*, "Outcomes of the emdataresource cryo-em ligand modeling challenge," *Research Square*, pp. rs–3, 2024.

[86] A. Dhakal, R. Gyawali, L. Wang, and J. Cheng, "CryoTransformer: a transformer model for picking protein particles from cryo-EM micrographs," *Bioinformatics*, vol. 40, no. 3, p. btae109, 02 2024. [Online]. Available: https://doi.org/10.1093/bioinformatics/btae109

[87] M. C. DiIorio and A. W. Kulczyk, "Novel artificial intelligence-based approaches for ab initio structure determination and atomic model building for cryo-electron microscopy," *Micromachines*, vol. 14, no. 9, 2023. [Online]. Available: https://www.mdpi.com/2072-666X/14/9/1674

[88] N. Giri, L. Wang, and J. Cheng, "Cryo2structdata: A large labeled cryo-em density map dataset for ai-based modeling of protein structures," *Scientific Data*, vol. 11, no. 1, p. 458, 2024.

[89] Phenix.map_to_model, "A fully automatic method yielding initial models from high-resolution electron cryo-microscopy maps," https://phenix-online.org/phenix_data/terwilliger/map_to_model_2018/, 2018, [Online; accessed 17-Dec-2023].

[90] C. Zhang, M. Shine, A. M. Pyle, and Y. Zhang, "Us-align: universal structure alignments of proteins, nucleic acids, and macromolecular complexes," *Nature methods*, vol. 19, no. 9, pp. 1109–1115, 2022.

[91] M. Steinegger and J. Söding, "Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets," *Nature biotechnology*, vol. 35, no. 11, pp. 1026–1028, 2017.

[92] G. Terashi, X. Wang, D. Prasad, T. Nakamura, and D. Kihara, "Deepmainmast: integrated protocol of protein structure modeling for cryo-em with deep learning and structure prediction," *Nature Methods*, vol. 21, no. 1, pp. 122–131, 2024.

[93] P. V. Afonine, B. K. Poon, R. J. Read, O. V. Sobolev, T. C. Terwilliger, A. Urzhumtsev, and P. D. Adams, "Real-space refinement in phenix for cryo-em and crystallography," *Acta Crystallographica Section D: Structural Biology*, vol. 74, no. 6, pp. 531–544, 2018.

[94] A. Hatamizadeh, Y. Tang, V. Nath, D. Yang, A. Myronenko, B. Landman, H. R. Roth, and D. Xu, "Unetr: Transformers for 3d medical image segmentation," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2022, pp. 574–584.

[95] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[96] T. Dozat, "Incorporating nesterov momentum into adam," 2016.

[97] W. A. Falcon, "Pytorch lightning," *GitHub*, vol. 3, 2019.

[98] N. S. Detlefsen, J. Borovec, J. Schock, A. H. Jha, T. Koker, L. Di Liello, D. Stancl, C. Quan, M. Grechkin, and W. Falcon, "Torchmetrics-measuring reproducibility in pytorch," *Journal of Open Source Software*, vol. 7, no. 70, p. 4101, 2022.

[99] N. Giri and J. Cheng, "Cryo2Struct: De Novo Atomic Protein Structure Modeling for CryoEM Density Maps Using 3D Transformer and HMM," Jun. 2024. [Online]. Available: https://doi.org/10.5281/zenodo.11492584

[100] S. Perera, P. Navard, and A. Yilmaz, "Segformer3d: an efficient transformer for 3d medical image segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 4981–4988.

[101] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and efficient design for semantic segmentation with transformers," in *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34. Curran Associates, Inc., 2021, pp.

12 077–12 090. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2021/file/64f1f27bf1b4ec22924fd0acb550c235-Paper.pdf

[102] W. Wang, E. Xie, X. Li, D. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," *CoRR*, vol. abs/2102.12122, 2021. [Online]. Available: https://arxiv.org/abs/2102.12122

[103] EMDB, "2021 ligand model challenge," 2021, available online: https://challenges.emdataresource.org/?q=2021-model-challenge (accessed on 22 November 2022).

[104] J. Eberhardt, D. Santos-Martins, A. Tillack, and S. Forli, "Autodock vina 1.2.0: New docking methods, expanded force field, and python bindings," *J. Chem. Inf. Model.*, vol. 61, pp. 3891–3898, 2021.

[105] C. Morris and D. Della Corte, "Using molecular docking and molecular dynamics to investigate protein–ligand interactions," *Mod. Phys. Lett. B*, vol. 35, p. 2130002, 2021.

[106] J. Fan, A. Fu, and L. Zhang, "Progress in molecular docking," *Quant. Biol.*, vol. 7, pp. 83–89, 2019.

[107] S. Mansoor, S. Shahid, K. Ashiq, N. Alwadai, M. Javed, S. Iqbal, U. Fatima, S. Zaman, M. Sarwar, F. Alshammari, and et al., "Controlled growth of nanocomposite thin layer based on zn-doped mgo nanoparticles through sol-gel technique for biosensor applications," *Inorg. Chem. Commun.*, vol. 142, p. 109702, 2022.

[108] S. Shahid, E. Anam, J. Mohsin, M. Sana, I. Shahid, B. Eslam, M. Rami, H. Alsaab, N. Awwad, H. Ibrahium, and et al., "The anti-inflammatory and free radical scavenging activities of bio-inspired nano magnesium oxide," *Front. Mater.*, vol. 9, 2022.

[109] Y. Cui, Q. Dong, D. Hong, and X. Wang, "Predicting protein–ligand binding residues with deep convolutional neural networks," *BMC Bioinformatics*, vol. 20, pp. 1–12, 2019.

[110] C.-Q. Xia, X. Pan, and H.-B. Shen, "Protein–ligand binding residue prediction enhancement through hybrid deep heterogeneous learning of sequence and structure data," *Bioinformatics*, vol. 36, pp. 3018–3027, 2020.

[111] S. Mylonas, A. Axenopoulos, and P. Daras, "Deepsurf: A surface-based deep learning approach for the prediction of ligand binding sites on proteins," *Bioinformatics*, vol. 37, pp. 1681–1690, 2021.

[112] J. Kandel, H. Tayara, and K. Chong, "Puresnet: Prediction of protein–ligand binding sites using deep residual neural network," *J. Cheminformatics*, vol. 13, pp. 1–14, 2021.

[113] J. Yang, A. Roy, and Y. Zhang, "Biolip: A semi-manually curated database for biologically relevant ligand-protein interactions," *Nucleic Acids Res.*, vol. 41, pp. 1096–1103, 2013.

[114] J. Hu, Y. Li, Y. Zhang, and D.-J. Yu, "Atpbind: Accurate protein–atp binding site prediction by combining sequence-profiling and structure-based comparisons," *J. Chem. Inf. Model.*, vol. 58, pp. 501–510, 2018.

[115] J. Desaphy, G. Bret, D. Rognan, and E. Kellenberger, "Sc-pdb: A 3d-database of ligandable binding sites-10 years on," *Nucleic Acids Res.*, vol. 43, pp. D399–D404, 2015.

[116] H. Ashtawy and N. Mahapatra, "Bgn-score and bsn-score: Bagging and boosting based ensemble neural networks scoring functions for accurate binding affinity prediction of protein–ligand complexes," *BMC Bioinformatics*, vol. 16, pp. 1–12, 2015.

[117] J. Jiménez, M. Skalic, G. Martinez-Rosell, and G. De Fabritiis, "K deep: Protein–ligand absolute binding affinity prediction via 3d-convolutional neural networks," *J. Chem. Inf. Model.*, vol. 58, pp. 287–296, 2018.

[118] H. Öztürk, A. Özgür, and E. Ozkirimli, "Deepdta: Deep drug–target binding affinity prediction," *Bioinformatics*, vol. 34, pp. i821–i829, 2018.

[119] L. Zheng, J. Fan, and Y. Mu, "Onionnet: A multiple-layer intermolecular-contact-based convolutional neural network for protein–ligand binding affinity prediction," *ACS Omega*, vol. 4, pp. 15 956–15 965, 2019.

[120] F. Zhu, X. Zhang, J. Allen, D. Jones, and F. Lightstone, "Binding affinity prediction by pairwise function based on neural network," *J. Chem. Inf. Model.*, vol. 60, pp. 2766–2772, 2020.

[121] M. Rezaei, Y. Li, D. Wu, X. Li, and C. Li, "Deep learning in drug design: Protein–ligand binding affinity prediction," *IEEE/ACM Trans. Comput. Biol. Bioinform.*, vol. 19, pp. 407–417, 2020.

[122] D. Jones, H. Kim, X. Zhang, A. Zemla, G. Stevenson, W. Drew Bennett, D. Kirshner, S. Wong, F. Lightstone, and J. Allen, "Improved protein–ligand binding affinity prediction with structure-based deep fusion inference," *J. Chem. Inf. Model.*, vol. 61, pp. 1583–1592, 2021.

[123] Y. Kwon, W. Shin, J. Ko, and J. Lee, "Ak-score: Accurate protein–ligand binding affinity prediction using an ensemble of 3d-convolutional neural networks," *Int. J. Mol. Sci.*, vol. 21, p. 8424, 2020.

[124] D. Karlov, S. Sosnin, M. Fedorov, and P. Popov, "Graphdelta: Mpnn scoring function for the affinity prediction of protein–ligand complexes," *ACS Omega*, vol. 5, pp. 5150–5159, 2020.

[125] K. Wang, R. Zhou, Y. Li, and M. Li, "Deepdtaf: A deep learning method to predict protein–ligand binding affinity," *Brief. Bioinform.*, vol. 22, pp. 1–15, 2021.

[126] J. Azzopardi and J. Ebejer, "Ligityscore: Convolutional neural network for binding-affinity predictions," *Bioinformatics*, vol. 3, pp. 38–49, 2021.

[127] S. Seo, J. Choi, S. Park, and J. Ahn, "Binding affinity prediction for protein–ligand complex using deep attention mechanism based on intermolecular interactions," *BMC Bioinformatics*, vol. 22, p. 542, 2021.

[128] A. Ahmed, B. Mam, and R. Sowdhamini, "Deelig: A deep learning approach to predict protein–ligand binding affinity," *Bioinform. Biol. Insights*, vol. 15, pp. 1–9, 2021.

[129] R. Wang, X. Fang, Y. Lu, and et al., "The pdbbind database: Methodologies and updates," *J. Med. Chem.*, vol. 48, pp. 4111–4119, 2005.

[130] M. Su, Q. Yang, Y. Du, G. Feng, Z. Liu, Y. Li, and R. Wang, "Comparative assessment of scoring functions: The casf-2016 update," *J. Chem. Inf. Model.*, vol. 59, pp. 895–913, 2018.

[131] H. Stärk, O. Ganea, L. Pattanaik, R. Barzilay, and T. Jaakkola, "Equibind: Geometric deep learning for drug binding structure prediction," in *Proceedings of the International Conference on Machine Learning (ICML)*. PMLR, 2022, pp. 20 503–20 521.

[132] G. Corso, H. Stärk, B. Jing, R. Barzilay, and T. Jaakkola, "Diffdock: Diffusion steps, twists, and turns for molecular docking," *arXiv*, 2022.

[133] G. G. Krivov, M. V. Shapovalov, and R. L. Dunbrack Jr, "Improved prediction of protein side-chain conformations with scwrl4," *Proteins: Structure, Function, and Bioinformatics*, vol. 77, no. 4, pp. 778–795, 2009.

[134] S. Chaudhury, S. Lyskov, and J. J. Gray, "Pyrosetta: a script-based interface for implementing molecular modeling algorithms using rosetta," *Bioinformatics*, vol. 26, no. 5, pp. 689–691, 2010.

[135] A. Bartesaghi, C. Aguerrebere, V. Falconieri, S. Banerjee, L. Earl, X. Zhu, N. Grigorieff, J. Milne, G. Sapiro, X. Wu, and et al., "Atomic resolution cryo-em structure of β-galactosidase," *Structure*, vol. 26, pp. 848–856, 2018.

[136] W. Yin, C. Mao, X. Luan, D. Shen, Q. Shen, H. Su, X. Wang, F. Zhou, W. Zhao, M. Gao, and et al., "Structural basis for inhibition of the rna-dependent rna polymerase from sars-cov-2 by remdesivir," *Science*, vol. 368, pp. 1499–1504, 2020.

[137] G. Pintilie, K. Zhang, Z. Su, S. Li, M. F. Schmid, and W. Chiu, "Measurement of atom resolvability in cryo-em maps with q-scores," *Nature methods*, vol. 17, no. 3, pp. 328–334, 2020.

[138] W. T. Baxter, R. A. Grassucci, H. Gao, and J. Frank, "Determination of signal-to-noise ratios and spectral snrs in cryo-em low-dose imaging of molecules," *Journal of structural biology*, vol. 166, no. 2, pp. 126–132, 2009.

[139] J. He, T. Li, and S.-Y. Huang, "Improvement of cryo-em maps by simultaneous local and non-local deep learning," *Nature Communications*, vol. 14, no. 1, p. 3217, 2023.

[140] J. Selvaraj, L. Wang, and J. Cheng, "Cryoten: Efficiently enhancing cryo-em density maps using transformers," *bioRxiv*, 2024.

[141] R. Sanchez-Garcia, J. Gomez-Blanco, A. Cuervo, J. M. Carazo, C. O. S. Sorzano, and J. Vargas, "Deepemhancer: a deep learning solution for cryo-em volume post-processing," *Communications biology*, vol. 4, no. 1, p. 874, 2021.

[142] H. Guan and D. Si, "Deeptracer-denoising: Deep learning for 3d electron density map denoising," in *2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2022, pp. 2080–2087.

[143] F. DiMaio, M. D. Tyka, M. L. Baker, W. Chiu, and D. Baker, "Refinement of protein structures into low-resolution density maps using rosetta," *Journal of molecular biology*, vol. 392, no. 1, pp. 181–190, 2009.

VITA

Nabin Giri was born in Kathmandu, Nepal, and completed his undergraduate degree at Bangalore University, India, in 2014. He later earned his Master of Science in Computer Science from the University of Central Missouri, USA, in 2020. Before graduate school, he worked as a Software Engineer at Capgemini and as a System Engineer at Max International. He began his PhD in Computer Science at the University of Missouri in Spring 2021, completing a minor in Statistics along the way. During his PhD, he interned at BIOVIA and Amazon Science, gaining valuable industry research experience. His research focuses on leveraging deep learning to advance structural biology. In the future, he aspires to develop innovative computational methods applicable across various scientific fields.