AAnchor: CNN guided detection of anchor amino acids in high resolution cryo-EM density maps

Mark Rozanov
Blavatnik School of Computer Science,Tel Aviv University

Haim J. Wolfson
Blavatnik School of Computer Science,Tel Aviv University

September 9, 2019

1 Abstract

Motivation: The recent cryo-EM resolution revolution enables the development of algorithms for direct de-novo modeling of protein structures into cryo-EM density maps. Such anchor residues can be exploited in several local de-novo modeling tasks, such as the reliable positioning of secondary structures, loop modeling and general fragment based modeling.
Results: A deep learning based method was developed for the detection of high confidence anchor amino acid residues in such a map. In the experimental results we show the ability of the proposed procedure to locate and classify a significant number of amino acids in density maps of 3.1 Å (or better) resolution. Our performance analysis indicates that the main factor affecting the detection accuracy is the lack of sufficient experimental data for the training stage of the algorithm. Thus, our method is expected to improve significantly in the near future, due to the rapid increase in the release of novel high resolution cryo-EM maps.
Availability: A web application based on the method can be found at `https://bioinfo3d.cs.tau.ac.il/AAnchor`
Contact: markroza@tauex.tau.ac.il
Supplementary information: Supplementary data are available at *Bioinformatics* online.

2 Introduction

Atomic accuracy models of protein structures are an invaluable tool for the elucidation of protein function. The recent "resolution revolution" Kühlbrandt (2014) in cryo electron microscopy (cryoEM) has led to an ever increasing number of near atomic resolution density maps deposited in the EM databank EMDB Lawson et al. (2016). While in 2002 the best structure deposited was at $9\mathring{A}$ resolution, recently (Banerjee et al. (2016), Bartesaghi et al. (2015)) key structures have been resolved at resolution better than $2.5\mathring{A}$.

Most of the techniques for modeling protein structures into intermediate resolution ($5-10\ \mathring{A}$) maps are based on rigid fitting of template protein structures into these maps. Many of these methods (Dror et al. (2007), Jiang et al. (2001), Rusu & Wriggers (2012), Si et al. (2012), Yu & Bajaj (2008)) use secondary structures as anchors for their fitting procedure.

At resolutions better than $4-5\mathring{A}$, the goal is more ambitious and de novo modeling techniques are being exploited. Some of them are adaptations of the standard X-ray crystallography modeling methods, however these tend to be time consuming. Recently, several de-novo modeling techniques have been developed to deal specifically with cryoEM density maps-DiMaio & Chiu (2016). Pathwalking Chen et al. (2016) detects first pseudo-atom anchors and then applies the travelling salesperson (TSP) combinatorial optimization algorithm to detect a long enough path which should model the protein backbone. MAINMAST Terashi & Kihara (2018) detects a set of anchor points and calculates the backbone by applying a minimun spanning tree (MST) approach. A recently published method Wang et al. (2015) fits short sequence based structure fragment templates into the density map and applies a Monte Carlo simulated annealing procedure to detect a set of mutually compatible fragments. All of the above mentioned procedures require prior segmentation of the density map into its various protein chains.

Prior detection of reliable **amino acid anchors** in the density map, namely having knowledge of even a relatively small number of amino acids, whose identity and location has been established with high confidence can be used to guide the various de-novo modeling methods, as well as serve as a starting point for the development of novel methods. In particular, it could lead to the development of novel techniques, which do not require prior segmentation of the EM density map. While in high resolution maps (roughly, 3.5 $\mathring{A}$ or better), sidechains become visible and individual rotamers may be distinguished (DiMaio & Chiu (2016),Cassidy et al. (2018)), no automated method has been suggested to detect specific amino acids in a cryo-EM density map.

In this work we present a machine learning (ML) algorithm nicknamed **AAnchor** (amino-acid anchor) for the detection and localization of amino acids in a high resolution cryo EM density map. The two primary goals of this study are (i) to develop an automatic tool for the detection and classification of amino acids in a near atomic resolution cryo EM map; and (ii) to provide a quantitative analysis of the ability to detect a specific amino acid in an experimental cryo EM map.

The preliminary results of our algorithm are quite encouraging. For example, on the 3.1$\mathring{A}$ cryo EM map of lysenin pore Bokori-Brown et al. (2016), the algorithm localizes above one hundred amino acids of different types with confidence above 80%. These preliminary results also indicate that the quality of detection is best for amino acids, which have a small number of rotamers and a large enough training set. This last observation also explains why contrary to expectation the results for 2.9$\mathring{A}$ and 3.1 $\mathring{A}$ resolution maps are better then those for 2.2 $\mathring{A}$ maps. This is due to the limited training data set existing for the better resolution maps. Thus, with the rapid increase in high resolution cryoEM structures and the availability of larger trainng sets, the results of our machine learning based methodology are expected to improve significantly.

A server of the detection/prediction stage of the AAnchor algorithm is available at `http://bioinfo3d.cs.tau.ac.il/AAnchor/`.

3 Methods

3.1 The Task

Given an electron density map of a protein/macromolecular assembly, our task is to detect voxels in this map, which correspond to the location of the centers of mass of specific amino acids. The goal is to report only those amino acids, which have been detected with high confidence, which we call "anchors". The two main building blocks of our method are a **classification** CNN and a **detection** algorithm. The detection algorithm generates candidates for the classification CNN and filters out candidates of expected low accuracy.

3.2 A CNN for the classification task

Convolutional neural networks (CNN) were proposed by Yann LeCun in 1989 for zip code recognition (Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard & Jackel 1989). A CNN consists of alternating convolutional and pooling layers optionally followed by fully connected layers. The first and last layers are the input and output layer respectively, while the other layers are referred to as hidden layers.

Formally, a CNN of depth D is a composition of D parametrized functions $\{f_1, \cdots, f_D\}$, which maps an input vector $\mathbf{x}$ to an output vector $\mathbf{y}$:

$$\mathbf{y} = f(\mathbf{x}) = f_D(\mathbf{z}, w_D, b_D)\circ, \ldots, \circ f_1(\mathbf{x}, w_1, b_1), \quad (1)$$

where w_k and b_k are the weights and biases vectors for the function f_k. The functions f_k are the previously mentioned layers.

Given a set of labeled data pairs $\{(\mathbf{x}^i, \mathbf{y}^i)\}_{i=1}^M$, the training process of a CNN defined by (1) is a process of a numerical solution of the optimization problem:

$$\begin{array}{l} \text{Find } \{w_k, b_k\}_{k=1}^D \text{ which minimize:} \\ \frac{1}{M} \sum\limits_{i=1}^{M} d\left(f(\mathbf{x}_i), \mathbf{y}_i\right), \end{array} \tag{2}$$

where $d(\cdot, \cdot)$ is the loss function expressing a penalty for an incorrect classification. For a comprehensive discussion of CNNs the reader is referred to (Goodfellow et al. 2016).

3.3 Softmax CNN for the Amino Acids Classification

We formulate the amino acids classification problem as a multiclass classification of volume cubes of a cryo-EM map. For a cryo EM map $E \in \mathbb{R}^{Length \times Height \times Width}$, denote by $\mathbf{x} \in \mathbb{R}^{L \times L \times L}$ (L in our experiments was $11\mathring{A}$) a local volume cube of the map, which is examined to contain the center of mass of an amino acid (of a specific type) in the proximity of the local cube's center. Let $c \in \{1, \cdots, C\}$ denote the corresponding label/type of the volume cube $\mathbf{x}$, in which C is the total number of classes. In our case C=21, which represent the 20 amino acids as well as a local volume, which does not contain a center of mass of an amino acid close to its center.

We train a CNN network of depth D, such that

$$c = \operatorname*{argmax}_i \mathbf{p}\left(\mathbf{y}_D(\mathbf{x})\right), \tag{3}$$

where $\mathbf{p}\left(\mathbf{y}_D(\mathbf{x})\right) = f(\mathbf{x})$ is the output of the last layer of the CNN obtained by applying the softmax function to output $\mathbf{x}$ of the previous layer. The softmax function $\mathbf{p}(\mathbf{y}) = (p_1(\mathbf{y}), \cdots, p_C(\mathbf{y}))$ is used to transform the vector of C (arbitrary) real values to a vector, where each value is in the $(0, 1)$ interval and the sum of all the values is 1. It is defined as:

$$p_i(\mathbf{y}) = e^{y_i} / \sum_{k=1}^{C} e^{y_k}. \tag{4}$$

Since $\sum\limits_i p_i(\mathbf{y}) = 1$ and $0 \le p_i(\mathbf{y}) \le 1$, the output of the softmax function is often (informally) treated as probabilities of a cube $\mathbf{x}$ to have label i.

3.4 Network Architecture and Training Details

The detailed architecture of the applied softmax CNN is presented in Table 1. We use the rectified linear (ReLU) activition function which is known to provide the best learning rate in image classification tasks (Krizhevsky et al. 2012). Keras (Chollet & Others 2015) is used to implement and train the softmax CNN. The training time for one epoch was 3 minutes for one million samples on a server with 4 NVIDIA Titan black GPUs, each with 2880 cores.

3.5 Preparation of the Datasets for the Training Stage

We describe the datasets used for training of the CNN and their structure. The input to this stage are pairs of a cryoEM map with an atomic structure fitted to it. From each map we extract a set of volume cubes of size $L \times L \times L$ (L in our experiments is $11\mathring{A}$) centered at each amino acid of the map. Such a cube is labeled by the fitting amino acid type. In practice, we, usually, label 8 cubes with this label, namely, if the coordinates of the amino acid center of mass are not integer, we assign this label to all the nearby cubes with integer coordinates at their center. We normalize the density of a labeled cube so that the mean density is 0 and the standard deviation is 1. This is due to the fact that the average density of a cryoEM map varies between different regions. In addition to the amino-acid induced cubes, we sample a sufficient number of density cubes that do not represent centers of mass of amino acids and assign them the 21'st ("zero") label.

The labeled volume cubes from all the training maps of the dataset are fed into the CNN training procedure.

We have applied the above described procedure to several diverse datasets. Since there is not enough experimental data to perform proper training, we have used both available experimental datasets as well as datasets of simulated EM maps at the required resolutions. Simulated and experimental datasets were created for each of the resolution spans presented in Table 2.

The **simulated** dataset consists of structures from the Dunbrack Rotamers Library (Shapovalov & Dunbrack 2011) and cryo EM maps created using the UCSF Chimera (Pettersen et al. 2004) *molmap* command. Each map was created at a randomly selected resolution within a given resolution span.

The **experimental** dataset consists of publicly available cryo EM maps from the EMDataBank (Lawson et al. 2016) within the required resolution span together with aligned/fitted PDB structures.

A **data augmentation** procedure was employed to increase the dataset and reduce the overfitting effect. Each map was rotated at a random angle together with the corresponding fitted atomic-resolution model and the rotated map and model were added to the dataset. We performed 10 rotations for each protein map. Since a virus structure already consists of repetitions of small sub-structures at different poses, virus maps were excluded from the augmentation procedure.

3.6 Detection

The workflow of the proposed detection method is illustrated in Figure 2. In the preprocessing phase an input cryoEM map is resampled to a grid of $1\mathring{A} \times 1\mathring{A} \times 1\mathring{A}$ voxels. Using the sliding window approach we sample the volume cubes from the resampled map (see Figure). Cubes with average density value less than the average density of the map are filtered out. The remaining cubes density is normalized to 0 mean and standard deviation 1. For each cube $\mathbf{x}$ we obtain the predicted value from the three pretrained classification CNNs: $N_S(\mathbf{x})$, $N_E(\mathbf{x})$, $N_{ES}(\mathbf{x})$.

The predicted label and confidence are calculate by combination of the results of three CNNs. The combination method depends on the amino acid type (Table 5), and is one of the following: one of $N_S(\mathbf{x})$, $N_E(\mathbf{x})$, $N_{ES}(\mathbf{x})$, majority of $N_S(\mathbf{x})$,$N_E(\mathbf{x})$,$N_{ES}(\mathbf{x})$, mean value of $N_S(\mathbf{x})$,$N_E(\mathbf{x})$,$N_{ES}(\mathbf{x})$,or mean value of $N_S(\mathbf{x})$, $N_E(\mathbf{x})$. A cube centre is marked as an amino acid centre if the resulting confidence is above a threshold. The threshold value depends on the amino acid type and calculated in the training phase.

Table 1: Classification CNN architecture

Layer	Type	Filter Dimensions	Input/Output Dimensions
1	Input		$11 \times 11 \times 11 \times 1$
2	3D Conv	$3 \times 3 \times 3 \times 1 \times 50$	$9 \times 9 \times 9 \times 50$
3	3D Conv	$2 \times 2 \times 2 \times 50 \times 50$	$8 \times 8 \times 8 \times 50$
4	Max Pool	$2 \times 2 \times 2$	$4 \times 4 \times 4 \times 50$
5	Fully Connected	$4 \times 4 \times 4 \times 50 \times 100$	1×100
6	SoftMax	1×100	1×21

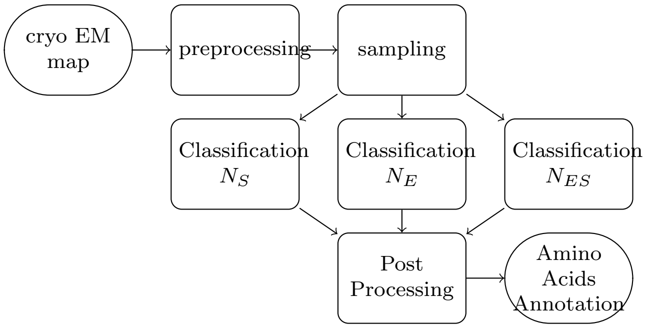


Figure 1: Workflow for the detection procedure

4 Experimental Results

4.1 The Classification Task

We start by presenting the results of the classification task. The classification task, unlike the detection task, assumes that an approximate location of an amino acid is given. Though the detection task is the more interesting one, analysis of the simpler classification task is key to understanding the algorithm behavior and improving its performance.

Once trained, the CNN is used to perform prediction on a validation dataset, which is disjoint from the training set. For each input cube with known label j, the softmax CNN produces "probabilities" p_k, $k = 0, \cdots, C$. The predicted label i the one giving maximal "probability", and the confidence is p_i.

4.1.1 Confusion Matrix and Reliability Curve

For an amino acid labeled j, we are interested in the following questions:

1. What are our chances to detect it correctly, and what are the chances of missing it with an amino acid i?

2. Does our confidence level p_j reflect the probability of the true detection? Namely, does a p_j fraction of all the inputs predicted as j represent correct predictions.

Commonly accepted tools for quantitative answers to these questions are the **confusion matrix** (Fawcett 2006) and the **reliability curve** (Guo et al. 2017).

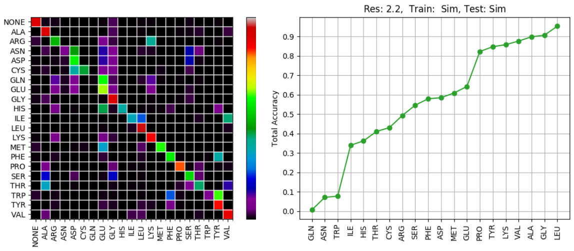
An entry $a_{i,j}$ of a **confusion matrix** A is defined as the ratio $a_{i,j} = \frac{T_j^j}{N_j}$, where N_j is the number of input cubes labeled j and T_j^i the number of inputs labeled j with predicted label i. An ideal classification algorithm will result in $a_{i,i} = 1$ and $a_{i,j} = 0$ if $i \neq j$. We also refer to $a_{i,i}$ as the **total accuracy** of label i.

It is desired that softmax CNN be **well calibrated** (Guo et al. 2017), i.e., the predicted confidence should reflect the ground truth probability. A **reliability curve** is a plot of ground truth accuracy prediction vs reported confidence. For a perfectly calibrated network the probability curve is an identity function. For the experimental results the ground truth accuracy is estimated by grouping predictions in interval bins according to their reported confidence (Guo et al. 2017).

4.1.2 2.2$\mathring{A}$ Resolution

Simulation Results We start by analyzing the results obtained on simulated data. Though it is commonly accepted that simulated data is not as reliable as real data, analysis of classification CNN performance on simulated data provides important insight on the real data behavior. Also, simulated data results provide an upper limit of what can be achieved with the selected CNN architecture.

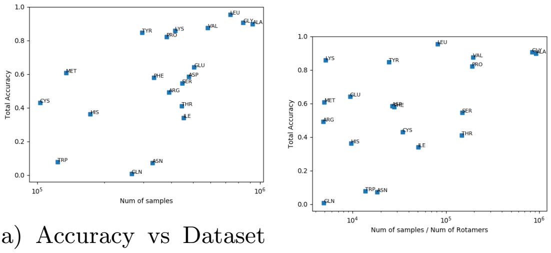
Figure 2a shows the confusion matrix obtained while the total accuracies are summarized in Figure 2b. The amino acids with the highest accuracy are: LEU, GLY, ALA, VAL, LYS, TYR, PRO. This indicates that the number of rotamers (Shapovalov & Dunbrack 2011) of an amino acid has a major effect on the accuracy. Since an amino acid with a large number of rotamers can exhibit numerous conformations, it is harder for a classification algorithm to adjust its parameters for the given amino acid. Except LYS, all of the above amino acids have a relatively small number of rotamers. Moreover, all the amino acids with a small number of rotamers have high classification accuracy, except Glutamic acid and Glutamine which are mutually mixed in classification. The size of the dataset is another dominant factor for the classification accuracy. In a Machine Learning approach a large training dataset enables robust parameter estimation and reduces the effect of overfitting. Figures 3a and 3b show the correlation between the training dataset size and the resulting accuracy.



(a) Confusion Matrix (b) Total Accuracy

Figure 2: Classification Results: Resolution 2.2$\mathring{A}$, Train: Simulated, Test : Simulated

Experimental Data Results We used a 2.2 cryo-EM map of β-galactosidase ((Bartesaghi et al. 2015). EMD-2984) to study the CNN performance. At the time of the research, only three additional cryoEM maps with resolution better than 2.3$\mathring{A}$ were available: EMD-8762 (Dong et al. 2017), EMD-8194 (Merk et al. 2016), and EMD-3295 (Banerjee et al. 2016). Even augmented , this is definitely not enough for proper training of the CNN. The confusion matrix and the total accuracies are shown in Figures 4a and 4b. The accuracy dependence on the dataset size is shown in Figures 7a and 7b. Despite significantly lower accuracies, the tendency observed on simulated data is preserved.



(a) Accuracy vs Dataset size

(b) Accuracy vs Dataset size normalized to the number of Rotamers

Figure 3: Effect of the Training Dataset Size on the Classification Accuracy: Resolution 2.2$\mathring{A}$, Train: Simulated, Test : Simulated

In order to improve the classification accuracy we tried two methods to overcome the lack of data:

1. We trained up to 6 independent CNNs and averaged the obtained probabilities.

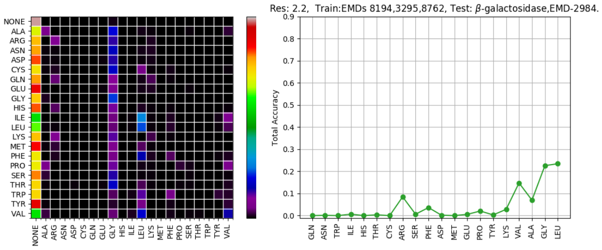
2. We combined the real maps with the simulated maps and created a new dataset.

The first method, averaging over multiple networks, did not result in any significant improvement of the classification accuracy. This indicates that the obtained low accuracy is not due to overfitting, but due to the small training set. Best classification results were achieved by combining in the training set simulated and experimental data in equal proportion. We denote the CNN trained on this combined dataset as N_{ES}^{22}. The detection accuracies for this case are shown in Figures 6a and 6b.

Figure 8 shows the estimated reliability curves for N_{ES}^{22}. Due to lack of data, reliability curves cannot be estimated for every input, but the overall tendency is that the output of the softmax layer of N_{ES}^{22} underestimated the classification accuracy.

4.1.3 Resolutions 2.9$\mathring{A}$ and 3.1$\mathring{A}$

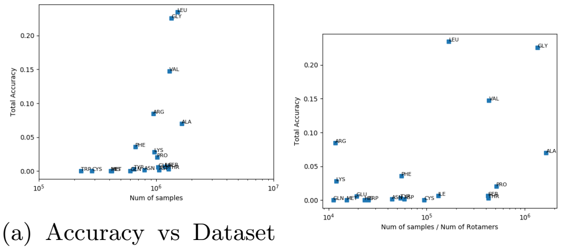
Coarsening the resolution has two opposite effects on the classification accuracy. Intuitively, in higher resolution maps the amino acids shape is sharper, but



(a) Confusion Matrix

(b) Total Accuracy

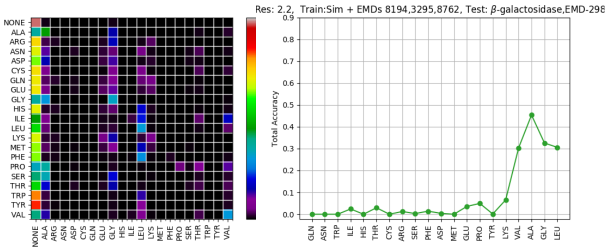
Figure 4: Classification Results: Resolution 2.2$\mathring{A}$, Train: EMDs 8762,3295,8184, Test : β-galactosidase, (Bartesaghi et al. 2015).



(a) Accuracy vs Dataset size

(b) Accuracy vs Dataset size normalized to a number of Rotamers

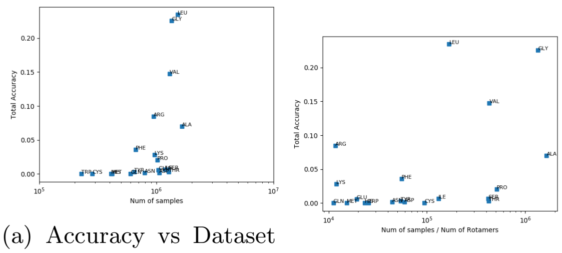
Figure 5: Effect of Training Dataset Size on the Classification Accuracy: Train: EMDs 8762,3295,8184, Test : β-galactosidase, (Bartesaghi et al. 2015).



(a) Confusion Matrix

(b) Total Accuracy

Figure 6: Classification Results: Resolution 2.2$\mathring{A}$, Train: Simulation + EMDs 8762,3295,8184, Test : β-galactosidase, (Bartesaghi et al. 2015).



(a) Accuracy vs Dataset size

(b) Accuracy vs Dataset size normalized to the number of Rotamers

Figure 7: Effect of Training Dataset Size on the Classification Accuracy: Train: EMDs 8762,3295,8184, Test : β-galactosidase, (Bartesaghi et al. 2015).

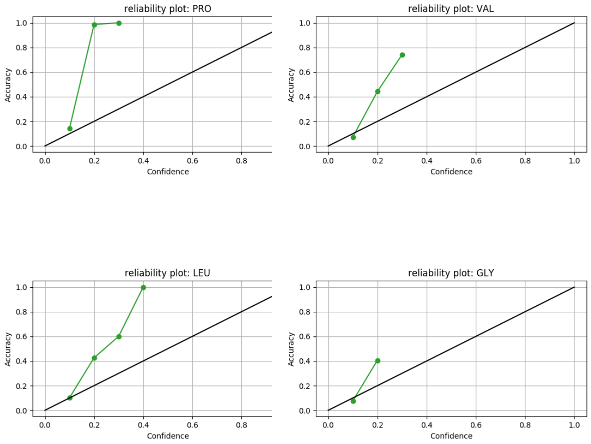


Figure 8: Estimated Reliability Curves for PRO, LEU, VAL, GLY : Resolution 2.2$\mathring{A}$, Train: Simulation + EMDs 8762,3295,8184, Test : β-galactosidase, (Bartesaghi et al. 2015).

lower resolution maps benefit from a larger training dataset. The size of the training dataset for three tested resolutions is shown on Table 2. The effect of a map resolution on the classification accuracy is shown in Figures 9a and 9b. While having only a small effect on simulated data (Figure 9a), the effect of resolution on the real data is well expressed (Figure 9b). Accuracies for 2.9$\mathring{A}$ and 3.1$\mathring{A}$ are significantly better than those for 2.2$\mathring{A}$. This is clearly due to the increased training dataset. The only exception is Alanin, which is probably too small to be detected at resolutions above 2.2$\mathring{A}$. However, moving from 2.9$\mathring{A}$ to 3.1$\mathring{A}$ we see that for the majority of the amino acids the total accuracy values are decreased. Thus, in this transition the effect of increasing the training dataset size did not compensate for the degradation in map precision.

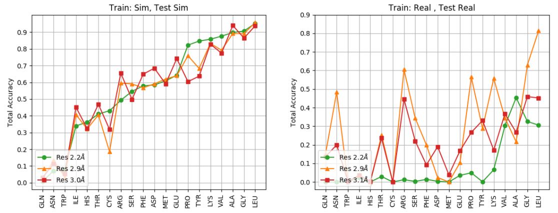
Resolution Span	N. Maps	Map for Validation
1.8 – 2.3$\mathring{A}$	3	EMD-2984 , 2.2$\mathring{A}$ *beta* -galactosidase (Banerjee et al. 2016)
2.7 – 2.9$\mathring{A}$	8	EMD-6224 ,2.9$\mathring{A}$ Anthrax toxin protective antigen pore (Jiang et al. 2015)
2.9 – 3.1$\mathring{A}$	13	EMD -8015, 3.1$\mathring{A}$ Lysenin Pore (Bokori-Brown et al. 2016)

Table 2: Experimental Train and Test data for Various Resolutions

The confusion matrices for 2.9$\mathring{A}$ Anthrax toxin protective antigen pore and 3.1$\mathring{A}$ Lysenin Pore are presented in Figures 10a and 10b, respectively. The Reliability Curves for the 2.9$\mathring{A}$ Anthrax toxin protective antigen pore and 3.1$\mathring{A}$ Lysenin Pore are presented in Figures 10a 10b, respectively. While at resolution of 2.9$\mathring{A}$ the estimated confidence is still less than the observed one, at 3.1$\mathring{A}$ resolution the estimated confidence is of good precision.

4.2 Detection Results

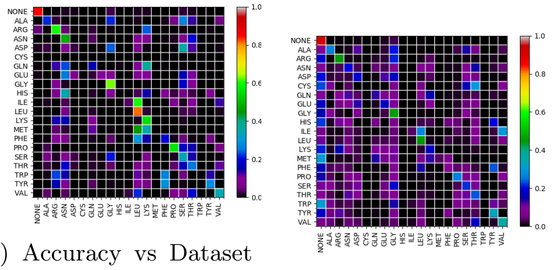
In the detection (also called localization) task, the exact location of an amino acid is unknown. Whilst the localization of all amino acids of a protein seems to be a hard problem at this time, detection of a subset of the amino acids obtained with high confidence is achievable. In this paper we focused on identifying **anchors**, i.e., amino acids, which have been located



(a) Simulated Data

(b) Real Data (Mixed with Simulation for Resolution 2.2$\AA$

Figure 9: Total classification accuracies for different resolutions.



(a) Accuracy vs Dataset size

(b) Accuracy vs Dataset size normalized to a number of Rotamers

Figure 10: Classification Accuracy Results for Resolution 2.9 and 3.1

and labeled with confidence above 80%.

For each resolution case we tested a number of different classification methods, while the preprocessing and post-processing phases remain unchanged.

Table 3 summarizes the detection picks for the 2.2$\AA$ map of *beta*-galactosidase (Bartesaghi et al. 2015). There we succeed to detect amino acids of four types: ARG, LEU, PRO, VAL with confidence above 80%. The fraction of the detected high confidence amino acids is up to 10% from the total amount of the same type. We expect this result to improve, as more high resolution cryo EM maps are being released.

Table 4 summarizes the detection picks for the test

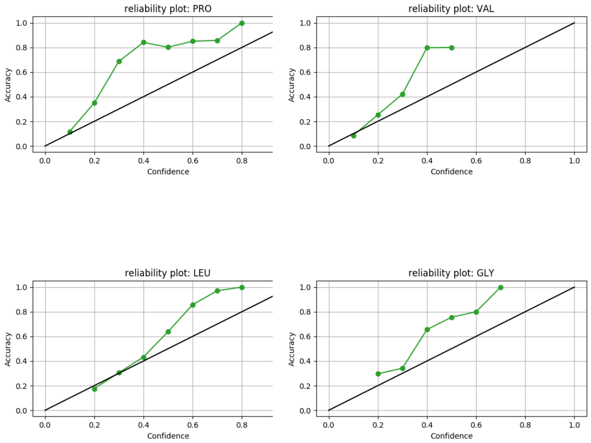


Figure 11: Estimated Reliability Curves for PRO, LEU, VAL, GLY : Resolution 2.9$\AA$, Train: Experimental Data, Test : Anthrax toxin protective antigen pore (Jiang et al. 2015).

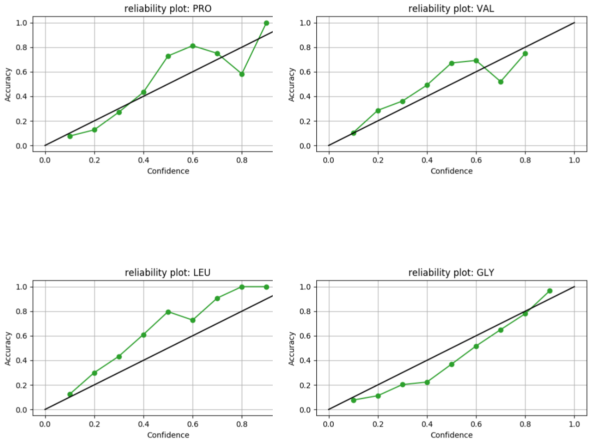


Figure 12: Estimated Reliability Curves for PRO, LEU, VAL, GLY : Resolution 3.1$\AA$, Train: Experimental Data, Test : Lysenin Pore (Bokori-Brown et al. 2016).

Amino Acid Type	Picks with Confidence of 80%	Total in Protein	Best Method
ARG	17	133	$maj(N_S, N_E, N_{ES})$
LEU	15	231	$mean(N_S, N_E, N_{ES})$
PRO	10	189	N_{ES}
VAL	10	119	$mean(N_S, N_E, N_{ES})$

Table 3: Detection Results 2.2 $\mathring{A}$ cryo-EM single particle reconstruction of beta galactosidase (emd-2984)

Amino Acid Type	Picks with Confidence of 80%	Total in Protein	Best Method
ASN	5	287	N_E
ARG	20	119	N_E
LEU	40	231	N_E
LYS	30	189	$mean(N_S, N_E)$
PRO	80	133	$maj(N_S, N_E, N$
TYR	18	91	N_{ES}
VAL	35	161	N_E

Table 4: Detection results for 2.9 $\mathring{A}$ CryoEM single particle reconstruction of an "anthrax toxin protective antigen pore (emd-6224)"

map of the 2.9 $\mathring{A}$ resolution anthrax toxin protective antigen pore (Jiang et al. 2015). We succeded to detect 70% of the Prolines. Figure 13 illustrates the results of proline detection. The detection rate of ARG, LEU, TYR,VAL was about 20%.

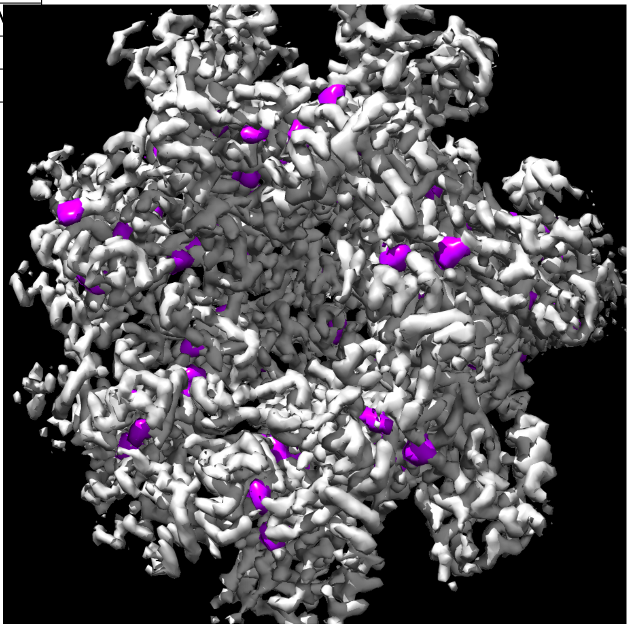
Table 5 summarizes the detection picks for the test map of 3.1 $\mathring{A}$ resolution Lysenin Pore (Bokori-Brown et al. 2016). We detected more than 20% of the Leucine residues. We also succeded to detect about 10% of LYS, TYR, ARG, GLY and PRO. Figure 14 illustrates the detection results for LEU, LYS, TYR and GLY at map resolution of 3.1$\mathring{A}$.

Note that the use of simulated data was crucial for the 2.2$\mathring{A}$ resolution experiment. For resolutions 2.9$\mathring{A}$ and 3.1$\mathring{A}$ the role of simulated data is less important, since more experimental maps are available.

Amino Acid Type	Picks with Confidence of 80%	Total in Protein	Best Method
ARG	10	117	N_{ES}
GLY	25	207	N_E
LEU	25	108	N_E
LYS	20	171	N_E
PRO	9	72	$mean(N_R, N_E)$
TYR	18	144	$maj(N_R, N_E, N_{ES})$

Table 5: Detection results for 3.1 $\mathring{A}$ CryoEM single particle reconstruction of an "Lysenin Pore (emd-8015)"

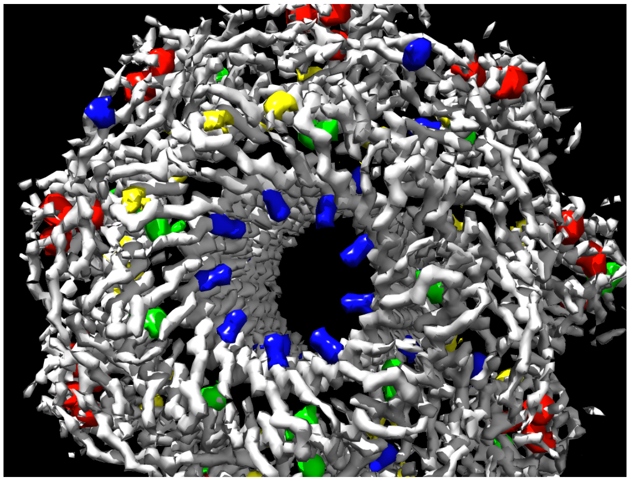
Figure 13: Proline detection in 2.9 $\mathring{A}$ CryoEM single particle reconstruction of an "anthrax toxin protective antigen pore (emd-6224). Purple regions indicate the detected proline residues



5 Conclusions

In this paper we present a CNN based method for localization and classification of amino acids in high

Figure 14: Detection leucines, lysines, glycines and tyrosines in 3.1 $\mathring{A}$ CryoEM single particle reconstruction of single particle reconstruction of an "Lysenin Pore (emd-8015)". Red regions indicate the detected leucines, blue regions indicate the detected lysines, yellow regions indicate the detected glycines, and green regions indicate the detected tyrosines



resolution cryo EM maps. Whilst the *de-novo* detection of all protein residues is still a hard problem, we succeeded to detect with high confidence a significant percentage of some amino acids. Experimental results show that the proposed method is capable of detecting a sufficient number of amino acid "anchors" in a cryo Em map of resolution 3.1$\mathring{A}$ or higher. The reported confidence of a detection is at least as the ground truth accuracy. These anchors can be further exploited in conjunction with several proposed modeling techniques as well as in the development of novel modeling methods.

We analyzed the detection process and factors affecting the classification task and concluded that the number of rotamers for a given residue and the size of the training data set have a dominant effect on the detection rate, while the amino acid size plays a secondary role. Contrary to expectation, the results for $2.9 - 3.1$ $\mathring{A}$ resolution maps have a better detection potential than the more accurate 2.2 $\mathring{A}$ maps. This is due to the limited training data set existing for the better resolution samples. Also the detection accuracy can be significantly improved by combining simulated and experimental cryo EM maps to compensate for the lack of experimental data.

As the number of released high resolution maps grows (Lawson et al. 2016), the detection accuracy will rise. Applying post processing techniques such as Linear Regression (Naseem et al. 2010) and SVM (Girshick et al. 2014) may potentially decrease the number of false detections. Hopefully, with the accumulation of a sufficiently large number of experimental maps the classification of all voxels in a map to the different amino acids will be solvable using the Fully Convolutional CNN (Long et al. 2015) technique.

References

Banerjee, S., Bartesaghi, A., Merk, A., Rao, P., Bulfer, S. L., Yan, Y., Green, N., Mroczkowski, B., Neitz, R. J., Wipf, P., Falconieri, V., Deshaies, R. J., Milne, J. L., Huryn, D., Arkin, M. & Subramaniam, S. (2016), '2.3 Å resolution cryo-EM structure of human p97 and mechanism of allosteric inhibition', *Science* .

Bartesaghi, A., Merk, A., Banerjee, S., Matthies, D., Wu, X., Milne, J. L. S. & Subramaniam, S. (2015), '2.2 A resolution cryo-EM structure of β-galactosidase in complex with a cell-permeant inhibitor', *Science* **348**(6239), 1147–1151.
URL: *http://science.sciencemag.org/content/348/6239/1147*

Bokori-Brown, M., Martin, T. G., Naylor, C. E., Basak, A. K., Titball, R. W. & Savva, C. G. (2016), 'Cryo-EM structure of lysenin pore elucidates membrane insertion by an aerolysin family protein', *Nature Communications* .

Cassidy, C. K., Himes, B. A., Luthey-Schulten, Z. & Zhang, P. (2018), 'CryoEM-based hybrid modeling approaches for structure determination', *Curr. Opin. MicroBio.* **43**, 14—-23.

Chen, M., Baldwin, P. R., Ludtke, S. J. & Baker, M. L. (2016), 'De Novo modeling in cryo-EM density maps with Pathwalking', *Journal of Structural Biology* .

Chollet, F. & Others (2015), ‘Keras’, https://keras.io.

DiMaio, F. & Chiu, W. (2016), Tools for Model Building and Optimization into Near-Atomic Resolution Electron Cryo-Microscopy Density Maps, *in* ‘Methods in Enzymology’.

Dong, Y., Liu, Y., Jiang, W., Smith, T. J., Xu, Z. & Rossmann, M. G. (2017), ‘Antibody-induced uncoating of human rhinovirus B14’, *Proceedings of the National Academy of Sciences* .

Dror, O., Lasker, K., Nussinov, R. & Wolfson, H. (2007), ‘EMatch: An efficient method for aligning atomic resolution subunits into intermediate-resolution cryo-EM maps of large macromolecular assemblies’, *Acta Crystallogr., Sect D* **D63**(1), 42—-49.

Fawcett, T. (2006), ‘An introduction to ROC analysis’, *Pattern Recognition Letters* **27**(8), 861–874.
URL: *https://www.sciencedirect.com/science/article/pii/S016786550500303X*

Girshick, R., Donahue, J., Darrell, T. & Malik, J. (2014), Rich feature hierarchies for accurate object detection and semantic segmentation, *in* ‘Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition’.

Goodfellow, I., Bengio, Y. & Courville, A. (2016), *Deep Learning*, MIT Press.

Guo, C., Pleiss, G., Sun, Y. & Weinberger, K. Q. (2017), ‘On Calibration of Modern Neural Networks’.

Jiang, J., Pentelute, B. L., Collier, R. J. & Hong Zhou, Z. (2015), ‘Atomic structure of anthrax protective antigen pore elucidates toxin translocation’, *Nature* .

Jiang, W., Baker, M. L., Ludtke, S. J. & Chiu, W. (2001), ‘Bridging the information gap: Computational tools for intermediate resolution structure interpretation’, *J. Mol. Biol.* **308**(5), 1033—-1044.

Krizhevsky, A., Sutskever, I. & Hinton, G. E. (2012), ImageNet Classification with Deep Convolutional Neural Networks, *in* ‘Proceeding NIPS’12 Proceedings of the 25th International Conference on Neural Information Processing Systems’, Lake Tahoe, Nevada, pp. 1097–1105.

Kühlbrandt, W. (2014), ‘The Resolution Revolution’, **343**(March).

Lawson, C. L., Patwardhan, A., Baker, M. L., Hryc, C., Garcia, E. S., Hudson, B. P., Lagerstedt, I., Ludtke, S. J., Pintilie, G., Sala, R., Westbrook, J. D., Berman, H. M., Kleywegt, G. J. & Chiu, W. (2016), ‘EMDataBank unified data resource for 3DEM’, *Nucleic Acids Research* .

Long, J., Shelhamer, E. & Darrell, T. (2015), ‘Fully convolutional networks for semantic segmentation’, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **07-12-June**, 3431–3440.

Merk, A., Bartesaghi, A., Banerjee, S., Falconieri, V., Rao, P., Davis, M. I., Pragani, R., Boxer, M. B., Earl, L. A., Milne, J. L. & Subramaniam, S. (2016), ‘Breaking Cryo-EM Resolution Barriers to Facilitate Drug Discovery’, *Cell* .

Naseem, I., Togneri, R. & Bennamoun, M. (2010), ‘Linear regression for face recognition’, *IEEE Transactions on Pattern Analysis and Machine Intelligence* .

Pettersen, E. F., Goddard, T. D., Huang, C. C., Couch, G. S., Greenblatt, D. M., Meng, E. C. & Ferrin, T. E. (2004), ‘UCSF Chimera - A visualization system for exploratory research and analysis’, *Journal of Computational Chemistry* .

Rusu, M. & Wriggers, W. (2012), ‘Evolutionary bidirectional expansion for the tracing of alpha helices in cryo-electron microscopy reconstructions’, *Journal of Structural Biology* .

Shapovalov, M. V. & Dunbrack, R. L. (2011), ‘A smoothed backbone-dependent rotamer library for proteins derived from adaptive kernel density estimates and regressions’, *Structure* .

Si, D., Ji, S., Nasr, K. A. & He, J. (2012), A machine learning approach for the identification of protein secondary structure elements from electron cryo-microscopy density maps, *in* 'Biopolymers'.

Terashi, G. & Kihara, D. (2018), 'De novo main-chain modeling with MAINMAST in 2015/2016 EM Model Challenge'.

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

Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. H. & Jackel, L. D. (1989), 'Backpropagation Applied to Handwritten Zip Code Recognition', *Neural Computation* **1**(4), 541–551.

Yu, Z. & Bajaj, C. (2008), 'Computational approaches for automatic structural analysis of large biomolecular complexes', *IEEE/ACM Transactions on Computational Biology and Bioinformatics* .