RENNSH: A Novel α-helices Identification Approach for Intermediate Resolution cryo-EM Maps

Lingyu Ma[1], Marco Reisert[2] and Hans Burkhardt[1]

[1] *University of Freiburg, Institute of Computer Science Georges-Köhler-Allee Geb. 052, 79110 Freiburg, Germany*

[2] *University Hospital Freiburg, Department of Diagnostic Radiology Hugstetter Str. 55, 79106 Freiburg, Germany*

November 17, 2009

Abstract

Accurate identification of protein secondary structures is beneficial to understand three-dimensional structures of biological macromolecules. In this paper, a novel α-helices identification approach for intermediate resolution cryo-EM Maps is presented. The solution is based on an iterative k-nearest neighbors (kNN) classification approach with voxelwise Spherical Harmonic Descriptors (SHD), we would like to call it the RENNSH approach. Also an energy function E_{GO} is defined for parameter optimization and evaluating the identification quality, which can also be used by any other identification methods. Comparing with existing methods, the experimental results demonstrate that the RENNSH approach gives the best identification accuracy.

1 Introduction

Almost all major advances in medicine and life sciences today stem from the better understanding of the function of key macromolecules and their complexes, and the knowledge of the three-dimensional structure of a biological macromolecule is ultimately important for understanding its function. However based on the criteria in [Dutta 05], to date only 1.02% (611 among 59790 entries) structures in the Protein Data Bank (PDB) [Berman 00] are of macromolecular complexes, since the X-ray crystallography [Woolfson 97] and nuclear magnetic resonance (NMR) [Ferentz 00], two widely used techniques to obtain high (up to atomic) resolution structures of the individual component proteins and their complexes, are restricted to relatively small biological structures.

In the past decade, electron cryo-microscopy (cryo-EM) [van Heel 00] has established itself as an important and powerful technique for studying the structures of large biological macromolecules. With only few samples, the single-particle structures of assemblies with molecular weights larger than 100 kDa could be determined at the resolution level of 5-20 Å. Especially at intermediate resolutions, 5-10 Å, the secondary structure elements (SSEs), α-helices and, in favorable cases, even β-sheets can be visualized in large macromolecular complexes. The spatial arrangement of the SSEs then could be used to query in a database of high resolution protein structures (usually PDB) to find potential structural homologues [Bajaj 07], then finally the "full picture" of large macromolecular complexes with higher resolutions could be achieved. Accurate identification of protein secondary structures is the basis of the entire process. This paper will focus on automated α-helices identification methods.

Until recently, only several approaches have been developed to extract α-helices from the intermediate resolution cryo-EM density maps. In [Jiang 01], due to the densities for helical regions appearing higher than other regions of density map, first the whole density map is masked out below an appropriate threshold, then a template-based five-dimensional (three degrees of freedom in translation space and two degrees of freedom in orientation space) cross-correlation is used to locate candidate helical regions. For each region, three eigenvectors of the 3×3 second moments tensor are used as the measure to verify whether it represents a real helix. The criterion is that the largest eigenvalue must be significantly larger than the two smaller eigenvalues, while the two smaller eigenvalues must be similar. Another cross-correlation based method is presented in [Lasker 07], which consists of four stages. In the first stage, each voxel in the input density map is assigned an estimated helix direction based on the correlation score with an ideal two-turn α-helix template. Then local cylinder-like regions are built with a graph-based method. After that a linkage stage is applied for linking fractions of the same helix. Finally the nohelical cylinders are pruned in the pruning stage. The output is a set of 3D undirected line segments, each representing the central axis and the length of a possible helix. A skeleton-based approach is developed in [Yu 08]. The local maximal (critical) points [Yu 05] of the 3D density map with cylinder-like eigenvalues are selected as the seed points. To trace a line (helix) structure, it starts from the seed in two opposite directions, and follow the principal axis, which is defined by the 3×3 local structure tensor. At last the superfluous skeletons corresponding to the same helix are merged with each other.

In this paper, we present a novel α-helices identification method for intermediate resolution density maps, named the RENNSH approach. It is inspired by the finding that the voxels on helix axis share some invariant properties, which could perfectly be depicted by the Spherical Harmonic Descriptors (SHD) [Kazhdan 03] with rotation invariance. In this way it becomes a matter of pure pattern recognition. An iterative kNN classification approach is proposed for helix-like voxel identification. The final outputs are given after three post-processing stages, which are ε-connected components, PCA analysis and redundancy pruning respectively.

The rest of this paper is organized as follows. Section 2 briefly introduces the Spherical Harmonic Transform and descriptors. In Section 3 the outline and details of the RENNSH approach are presented. The experimental results, compared to other methods, will be presented in Section 4. Finally, conclusions and future work are presented in Section 5.

2 Spherical Harmonic Transform and Descriptors

The intuitive idea of RENNSH approach is trying to find some numerical descriptors to represent the common properties or patterns of entire helix axis voxels and their neighborhood, then the

α-helices identification task will become a matter of pure pattern recognition, which could be resolved by various existing classification or clustering methods. Through lots of experiments we have found that the common structure properties of helix axis voxels can well be expressed with their Spherical harmonic descriptors (SHD), which are based on the Spherical harmonic transform (SHT). Previously SHD has been used for 3D shape retrieval [Kazhdan 03] [Laga 06], now we introduce it to extract helix axis voxels from the intermediate resolution density map. Below we will give a short review for both SHT and SHD.

As is well-known [Vilenkin 68], the Spherical Harmonics provide an orthonormal basis for the Hilbert space of square integrable (L^2) functions on the 2-sphere (S^2) in three-dimensional space [Healy 03]. For any nonnegative integer l and m with $|m| \leqslant l$, the (l, m)-spherical harmonic $Y_l^m(\theta, \phi)$ is a harmonic homogeneous polynomial of degree l. The harmonics of degree l span a subspace of $L^2(S^2)$ of dimension $2l + 1$, which is invariant under rotations of the sphere. Any spherical function $f(\theta, \phi) \in L^2(S^2)$ can be decomposed as the sum of its harmonics

$$f(\theta, \phi) = \sum_{l \geqslant 0} \sum_{|m| \leqslant l} \hat{f}(l, m) Y_l^m(\theta, \phi) \tag{1}$$

where $\hat{f}(l, m)$ denotes the spherical harmonics coefficient, equal to $\langle f, Y_l^m \rangle$, and $Y_l^m(\theta, \phi)$ can be factorized as

$$Y_l^m(\theta, \phi) = k_{l,m} P_l^m(cos\theta) e^{im\phi} \tag{2}$$

where P_l^m is the associated Legendre function of degree l and order m, and $k_{l,m}$ is a normalization constant. Consequently, the computation of the spherical harmonic transform can be reduced to regular Fourier transform in the longitudinal coordinate ϕ followed by a projection onto the associated Legendre functions

$$\hat{f}(l, m) = \langle f, Y_l^m \rangle = k_{l,m} \int_0^{\pi} \left[\int_0^{2\pi} e^{-im\phi} f(\theta, \phi) d\phi \right] P_l^m(cos\theta) sin\theta d\theta \tag{3}$$

where the $sin\theta$ is used to encode the effect that the parameterization of a sphere using polar coordinates causes the paths on the equator to be bigger and thus should have more influence on the solution of the integral compared to the paths at the poles. In order to obtain a rotation invariant representation of a voxel grid, just like the situation we face in the cryo-EM maps, we can restrict the voxels grid to concentric spheres of different radii, and obtain the spherical harmonic representation of each spherical restriction independently [Kazhdan 03]. In mathematics this idea can be expressed as the $f(\theta, \phi) \in L^2(S^2)$ is band-limited with bandwidth $B \geqslant 0$, then for each $|m| \leqslant l < B$

$$\hat{f}(l, m) = \frac{\sqrt{2\pi}}{2B} \sum_{j=0}^{2B-1} \sum_{k=0}^{2B-1} a_j^B f(\theta_j, \phi_k) e^{-im\phi_k} P_l^m(cos\theta_j) \tag{4}$$

where the sample points are chosen from the equiangular grid: $\theta_j = \pi(2j+1)/4B$, $\phi_k = 2\pi k/2B$, and the weights a_j^B play a role analogous to the $sin\theta$ factor in the integrals. Finally the SHD with rotation invariance for each degree l of particular bandwidth B could be expressed as

$$\parallel \hat{f}^B(l) \parallel \; = \sqrt{\sum_{|m| \leqslant l} \left| \hat{f}^B(l, m) \right|^2} \tag{5}$$

and this is defined as Spherical harmonic descriptors (SHD), which is a two-dimensional matrix indexed by bandwidth B and degree l, where the bandwidth B is determined by the sampling of radius r when transforming the atom coordinates from Cartesian coordinate system to Spherical coordinate system.

Note that the Equation 5 only meets the necessary condition of invariants. However, it loses some discrimination performance for abandoning the phase information.

3 RENNSH approach for α-helices identification

The whole RENNSH approach consists of three processes, training process, query process and parameter optimization process. The training process generates three training classes, which are used in the query process to classify helix-like voxels from the density map. The last parameter optimization process optimizes the parameters in the query stage based on an elaborately defined energy function.

Note: in all the figures below, the texts with frame represent some kind of data, and the others without frames represent various operations.

3.1 Training process

The training process is divided into two rounds of training stages. Figure 1 shows the *1st* round of training stage, and Figure 2 shows the *2nd* round of training stage. The *pdb* file, denoted by the round corner frame in Figure 1, is the only input for the whole training process. As an example, Figure 3 shows the intermediate results of each step when protein 1BYA goes through the training process.

3.1.1 *1st* round of training stage

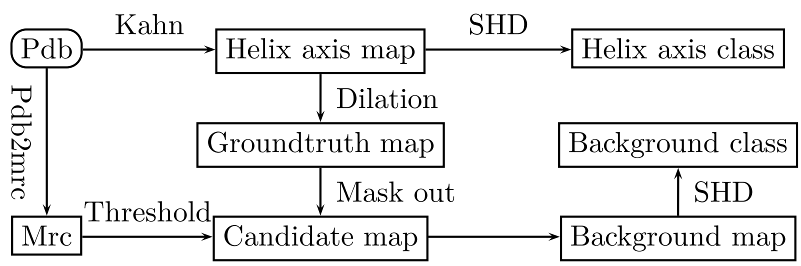


Figure 1: *1st* round of training stage

Pdb and mrc file. There are two kinds of data format we will deal with in the training process, the *pdb* file and the *mrc* file. Each *pdb* file from the PDB database contains atomic coordinates and secondary structure information of one protein. Since it is hard to get the accurate information about each helix from the real cryo-EM reconstructed maps, usually for testing and evaluating the helices identification methods, the *mrc* file, which contains the density information of one protein in three-dimensional space, will be generated from the *pdb* file to simulate the actual cryo-EM maps. By doing this the identification results, normally represented as some cylinders with orientation and length, can be compared with the helix structure information stored in the *pdb* file. In this paper, we will use the same program "pdb2mrc" as in [Jiang 01] [Lasker 07] [Yu 08] to generate *mrc* files from *pdb* files. The program runs as follows: for each atom in the *pdb* file, a Gaussian, $n_e e^{-\left(\frac{r}{k}\right)^2}$ is generated in the corresponding position in the *mrc* file, where n_e is the atomic number and r is the distance from the center of the atom. The Gaussian width, k, is constant for all atoms to approximately simulate a 3D structure at moderate resolution. In the figure, **Pdb** represents the original *pdb* file of one protein, which is downloaded from the PDB database, and **Mrc** represents the *mrc* file generated from this *pdb* file, which will be called **original mrc file** (see Figure 3(b)) in later presentation.

Map. All the maps in this paper, for example, Helix axis map in Figure 1, False helix map in Figure 2 and Helix region map in Figure 4, are all binary *mrc* file with the same size as the original *mrc* file. It will not be mentioned again when the new *maps* are presented.

Kahn method and helix axis map. All voxel densities in the helix axis map (see Figure 3(c)) are zero except some scattered voxels on which the helix axes are located. It will be clear soon why we generate a *mrc* file like this, and now let us just focus on how to get it. Several methods for finding the axis of a helix are presented in [Christopher 96]. We have chosen the *Kahn* method [Kahn 89], since no regression analysis is needed and for its elegant geometry representation. The method states that one bisector of the angle, defined by three consecutive C_α atoms, is perpendicular to the helix axis and intersects it. Repetition of this process with a different set of atoms gives another bisector also perpendicular to the helix axis. The direction vector of the axis could be represented by the cross product of these two bisectors. Since the C_α atoms coordinates and the helix structure information are already contained in the *pdb* file, labeling helix axis voxels in the *mrc* file becomes a trivial task.

SHD calculation and Helix axis class. For each nonzero voxel in the helix axis map, the SHD of the corresponding voxel, which means the voxel with the same coordinates in the original *mrc* file, is calculated with equation (5) in section 2. Note that in this paper every time when we say the SHD of one voxel is calculated, it means the SHD of the corresponding voxel is calculated in **the original *mrc* file**. After calculation, all the SHDs are stored together as the Helix axis class.

Groundtruth map. The groundtruth map (see Figure 3(d)) is the output of the helix axis map after morphology dilation. The structuring element is all one vector, whose length is determined by the helix radius. The dilation is done in all three directions. Finally the scattered nonzero voxels in the helix axis map become some nonzero regions, and each region corresponds to a helix in the *pdb* file.

Candidate map, Background map and Background class. The candidate map (see Figure 3(e)) is obtained by thresholding the original *mrc* map with an appropriate threshold. Since the densities of helical regions are higher than other regions in the density map, all the helical regions will be left after the thresholding operation, and only irrelevant regions will be eliminated. The background map (see Figure 3(f)) is the output of the candidate map **masked**

out[1] by the groundtruth map. For each nonzero voxel in the background map, the SHD of the corresponding voxel in the original *mrc* file is calculated and stored together as the Background class.

3.1.2 *2nd* round of training stage

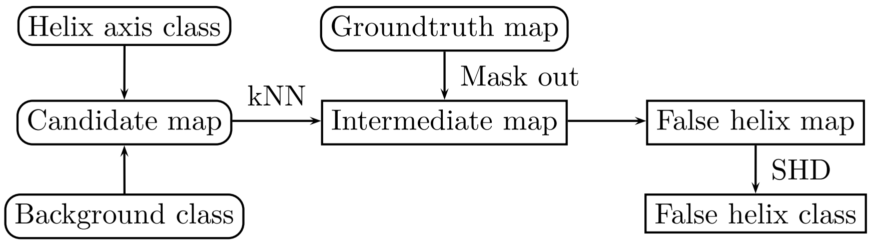


Figure 2: *2nd* round of training stage

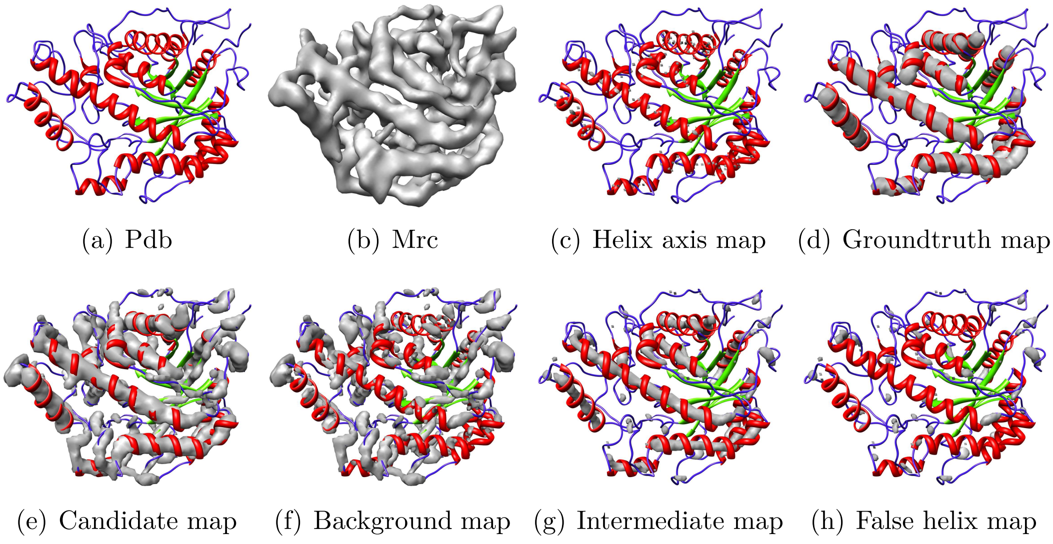
Here begins the *2nd* round of training stage. The text with round corner frame in Figure 2 denotes the data derived from the *1st* round of training stage.

Candidate map, kNN classification and Intermediate map. First for each nonzero voxel in the candidate map, the SHD of the corresponding voxel in the original *mrc* file is calculated. Then the SHD as a query is classified with kNN method into either the helix axis class or the background class, which means in the SHD feature space. We identify the $\boldsymbol{k}$ nearest SHDs from the two training sets, the helix axis class and the background class, and then assign the query SHD to the class having the largest number of representatives amongst this set. The intermediate map (see Figure 3(g)) is initialized to all zeros. For each query SHD from the candidate map, classified as the helix axis class, its corresponding voxel, which has the same coordinates in the candidate map, is evaluated to one. Finally, all nonzero voxels in the intermediate map represent the helix-like voxels after the classification.

False helix map and class. The false helix map (see Figure 3(h)) is the output of the intermediate map masked out by the groundtruth map. Each nonzero voxel denotes that it is classified as a helix-like voxel after the previous kNN classification, but its corresponding voxel in the groundtruth map does not belong to any helical region. Then for each nonzero voxel in the false helix map, the SHD of the corresponding voxel in the original *mrc* file is calculated and stored together in the False helix class.

Note: during above presentation, it is assumed that the object we are talking about is only one protein, which means everything is generated from one single *pdb* file. However the three training classes of RENNSH approach contain not only the SHDs from one protein, but the SHDs from many proteins, which compose a *training database* (it will be explained further in Section 4).

[1] When we say C is the output of A masked out by B, it means $C = A \cap B^c$.



(a) Pdb (b) Mrc (c) Helix axis map (d) Groundtruth map

(e) Candidate map (f) Background map (g) Intermediate map (h) False helix map

Figure 3: Training process of protein 1BYA

3.2 Query process

Figure 4 shows the query process of RENNSH approach. Note that the input of the query process includes the *mrc* file, the helix axis class, the background class and the false helix class, all denoted by the round corner frame. As an example, Figure 5 shows the intermediate results when protein 1AMQ goes through the query process.

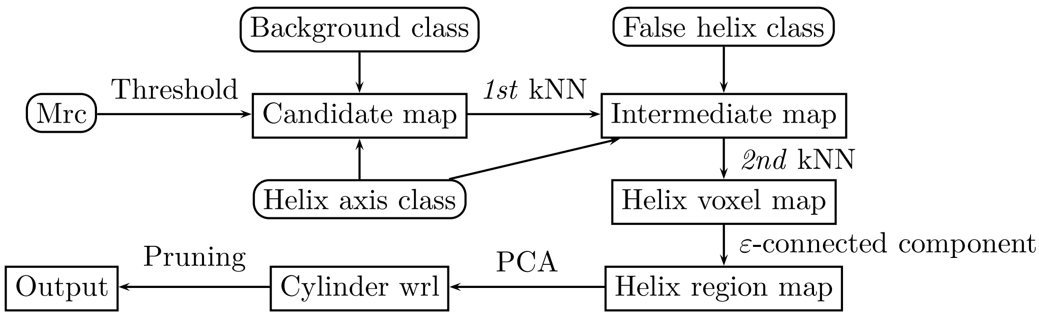
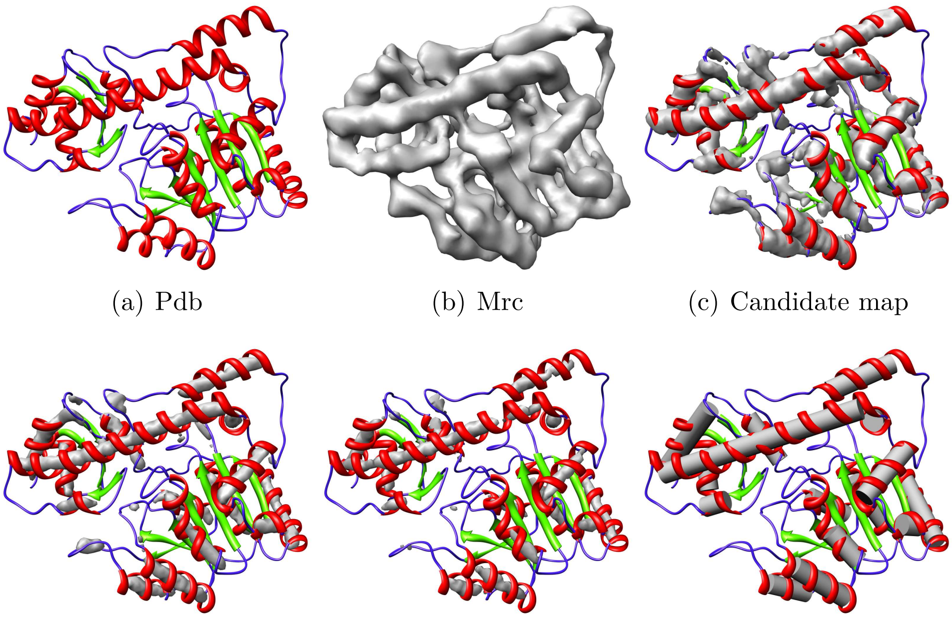


Figure 4: Query process

- First the *mrc* file (see Figure 5(b)) is masked out by a threshold, which is set the same as the one used in the training process, to generate the candidate map (see Figure 5(c)).

- In the next step *1st* kNN, the candidate map is classified exactly the same way as in the training process to generate the intermediate map (see Figure 5(d)).



(a) Pdb (b) Mrc (c) Candidate map

(d) Intermediate map (e) Helix voxel map (f) Output

Figure 5: Query process of protein 1AMQ

- For each nonzero voxel in the intermediate map, the SHD of the corresponding voxel in the original *mrc* file is calculated. Then each SHD will again be classified with kNN method, denoted by *2nd* kNN in the figure, however this time the training data consists of the helix axis class and the false helix class. The classification result is stored in the helix voxel map (see Figure 5(e)), in which each nonzero represents the query voxel, which has the same coordinates in the intermediate map, is classified into the helix axis class.

- Then the ε-connected component labeling method is used to label all the nonzero voxels in the helix voxel map as different helical regions, and the output is called the helix region map. The ε-connected component labeling runs the same as the normal connected component labeling algorithm, the only difference is for each nonzero voxel it checks all the neighbor voxels, whose distance to it is less than a pre-defined threshold ε.

- For each labeling region in the helix region map, the principal component analysis (PCA) is applied. Then all the voxels that belong to this region are projected on the first principal component vector. The two end points of the projection, meaning the two furthest projection points on the first principal component orientation, are used to define a cylinder (actually a line segment). The results are stored in a *wrl* file (based on the Virtual Reality Modeling Language), which is denoted as *Cylinder wrl* in the figure. The final output 5(f)) is just this *wrl* file with all the too short cylinders pruned according to a pre-defined minimum length.

3.3 Iterative kNN classification

We say RENNSH is an iterative kNN classification approach, since as in Figure 4 the input density map is first classified by the features stored in the helix axis class and the background class during the *1st* kNN classification. Then the result goes through the *2nd* kNN classification as input and is classified by the helix axis class and the false helix class.

Theoretically this iterative process could be done as many times as we want. For example, we could add one *3rd* kNN classification stage after the *2nd* kNN stage as in Figure 6, in which the helix voxel map (see Figure 4) goes through the *3rd* kNN classification with training features stored in the helix axis and the false helix class 2, and the output, the ***Helix voxel map 2***, could then be used to generate the helix region map (see Figure 4) by ε-connected component algorithm.

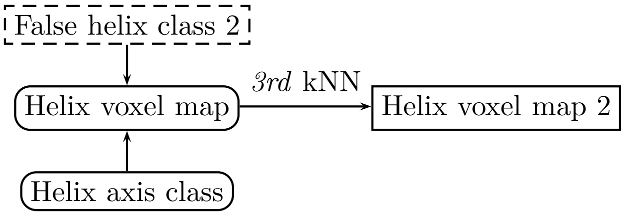


Figure 6: *3rd* kNN classification

The false helix class 2, denoted with dashed frame in Figure 6, can be obtained as in Figure 7, and the details of the process could refer to how the false helix class is obtained in the *2nd* round of training stage.

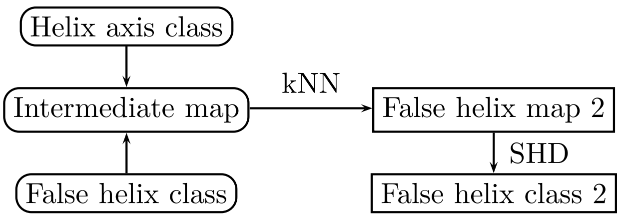


Figure 7: False helix class 2

Note: this iterative kNN classification process could also be extended into other pattern recognition applications.

3.4 Parameter optimization process

First we introduce two new maps, the *G-map* and the *O-map*.

The *G-map* of each protein is obtained as follows: first set all the voxel densities equal to zero, then generate the helix axis map the same way as in the *1st* round of training stage, label each nonzero voxel with the helix number in the *pdb* file, which could be obtained from the *pdb* file. After that each labeled helical group goes through the PCA analysis, in the way as we generated the cylinder wrl file from the helix region map in the query process. Finally evaluate all the voxels between the two end points in the *G-map* to one. Now each nonzero voxel in the *G-map* denotes that it is on one of the helix axes in the *pdb* file.

The *O-map* is much easier to obtain. First set all the voxel densities equal to zero, then utilize the cylinder information from the output of the query process. For each cylinder in the *wrl* file, calculate the two end points and evaluate all the voxels between them to one. Each nonzero voxel in the *O-map* belongs to one of the helix axes we found with RENNSH approach.

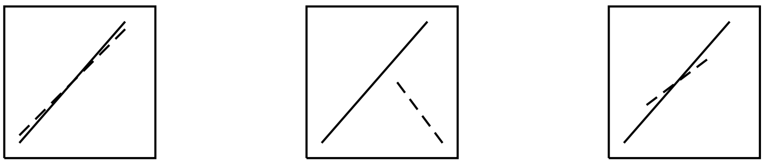
Suppose there are in all I proteins in the training data, and for each protein with index i there are M_i nonzero points in the *G-map*, for which the coordinates are denoted by $(x_{m_i}, y_{m_i}, y_{m_i})$, and N_i nonzero points in the *O-map*, for which the coordinates are denoted by $(x_{n_i}, y_{n_i}, y_{n_i})$, three energy functions E_G, E_O and E_{GO} are defined as

$$E_{GO} = \sum_{i=0}^{I-1} \left\{ \underbrace{\left[\frac{1}{M_i} \sum_{m_i=0}^{M_i-1} \min_{0 \leqslant n_i \leqslant N_i} d(m_i, n_i)^2 \right]^2}_{E_G} + \underbrace{\left[\frac{1}{N_i} \sum_{n_i=0}^{N_i-1} \min_{0 \leqslant m_i \leqslant M_i} d(n_i, m_i)^2 \right]^2}_{E_O} \right\} \tag{6}$$

where

$$d(m_i, n_i) = d(n_i, m_i) = \sqrt{(x_{m_i} - x_{n_i})^2 + (y_{m_i} - y_{n_i})^2 + (z_{m_i} - z_{n_i})^2} \tag{7}$$

in which the E_{GO} energy function is used for optimizing the parameter of RENNSH approach.



(a) (b) (c)

Figure 8: E_G, E_O and E_{GO} energies in different situations

Figure 8 illustrates how this definition works in different situations and the relationship among these three energies. The figures are made by first superposing the G-map and O-map of one protein and then projecting them on a two-dimensional plane. Suppose the protein has only one helix, the ideal situation appears in Figure 8(a), in which the two helices from G-map and O-map respectively are close to each other. In this case both E_G and E_O energies are low, so the final E_{GO} energy will also be low. However if the result is like in Figure 8(b), then all three energies are high. Finally if the situation is as shown in Figure 8(c), one energy of E_G and E_O will be low, and the other one is high. Because the square operators in the definitions will magnify the high energy, the final E_{GO} energy will also be high. In summary, the low E_{GO} energy could only be achieved when both E_G and E_O energies are low, which means the output of the query process is a good reflection to all the helix information stored in the *pdb* file.

Based on the analysis above, the ***seven*** main parameters in the query process, which are the threshold T for thresholding the original *mrc* file to the candidate map, the parameters k_1 and bandwidth B_1 for *1st kNN* classification, the parameters k_2 and bandwidth B_2 for *2nd kNN* classification, the parameter ε for connected component and the minimum length L_{min} for the final pruning, are determined by minimizing the energy E_{GO} as

$$\{T, k_1, k_2, B_1, B_2, \varepsilon, L_{min}\} = arg\ min\ E_{GO} \tag{8}$$

4 Experiments

In this section we will test the RENNSH approach and compare its performance with the other two kinds of methods, the ***helixhunter*** method in [Jiang 01] and the ***tensor skeleton*** approach in [Yu 08].

4.1 Training database

The training database for RENNSH contains 9 proteins randomly selected and downloaded from the PDB database, whose PDB IDs are 1A54, 1AKB, 1AMA, 1AMP, 1ASU, 1BE6, 1BYA, 1C4K and 1CDG. Each protein is blurred at 8 Å by *pdb2mrc* program to generate the original *mrc* files. After two rounds of training processes the feature SHDs are stored in the three classes, the helix axis class, the background class and the false helix class. Note we do not need to do a further training process to get the ***false helix class 2*** mentioned in 3.3, since we will see soon that two rounds of training processes are already good enough to identify the α-helices from the density map.

4.2 Equalize the number of SHDs in each class

After the *1st* round of training stage, it is quite natural that the number of SHDs in the helix axis class is much less than the number of SHDs in the background class, since the number of nonzero voxels in the helix axis map is restricted by the C_α atom numbers in the *pdb* file. In order to avoid the background class predominating the SHD feature space, we need to sample the background class to make the number of SHDs in these two classes roughly equal. The same operation should be applied again in the *2nd* round of training stage.

4.3 Parameter optimization for RENNSH

Consecutively run the query process and the parameter optimization process until all the parameters in the query process are determined by Equation 6 and 8. Notice that the parameter is training database dependent.

4.4 Helixhunter, Tensor skeleton methods and E_{GO} energy

Since the author does not exactly name the α-helices identification approach in [Yu 08], according to its definition, we will temporarily call it the Tensor skeleton approach in this paper for convenience. The two associated softwares for Helixhunter and Tensor skeleton methods, Eman and VolRover, could be downloaded from their websites. However the parameters for each method should be adjusted appropriately according to different database. The E_{GO} energy defined for RENNSH approach can also be used to optimize their parameters against the training database, and we believe the E_{GO} energy could be used in any α-helices identification method for parameter optimization. However as of this writing, the parameters for Helixhunter and Tensor skeleton are only determined according to the best results after trying different parameter settings, since it takes time to connect their source codes with ours.

4.5 Type of identification errors

There are two main identification errors for all of three methods, false negative (FN) and false positive (FP). FN means that some helices are unsuccessfully identified, and FP means that some nonexistent helices have been identified. During the experiments we found there is another kind of error appearing in the results from Helixhunter and RENNSH, which should not be classified as either FN or FP errors. It refers to the fact that one integrated helix breaks into several segments in the result, we will call it the ***Break*** error in this paper.

4.6 Identification results

Table 1 shows the identification results for 63 proteins downloaded from the PDB database, their IDs can be found in the second column. The number of helices in each protein is listed in the third column, ranges from 2 to 27. All these proteins are divided into two groups, one contains only α-helices and the other contains both α-helices and β-sheets, which are denoted with α and $\alpha\&\beta$ in the first column respectively. Figure 12 shows the identification results of three proteins, 1CH1, 1BFU and 1BPM, according to three identification methods.

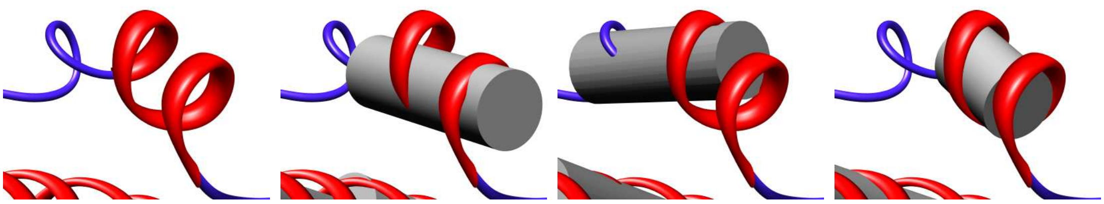
4.7 Discussion

Statistics in all. At the bottom of the table we could see there are in all 667 helices in these 63 proteins, the RENNSH has successfully identified 95.35% of them, and the other two methods have recognized 63.12% and 83.66% respectively.

Statistics contrast between different protein group. For all three methods, the identification accuracy of $\alpha\&\beta$ group is lower than the accuracy of α group. It does make sense, since the structures in $\alpha\&\beta$ proteins are usually more complicated than all α proteins. However the accuracies of Helixhunter and Tensor skeleton drop dramatically between these two protein groups, the former drops from 76.76% to 46.13%, and the latter from 93.24% to 71.72%. By contrast, the accuracy of RENNSH approach for these two groups is stable, dropping only slightly from 98.65% to 91.25%.

Helixhunter. The helixhunter gives the worst performance among these three methods. The experiments show that it is very sensitive to parameter shift, probably because it only depends on low resolution cross-correlation algorithm.

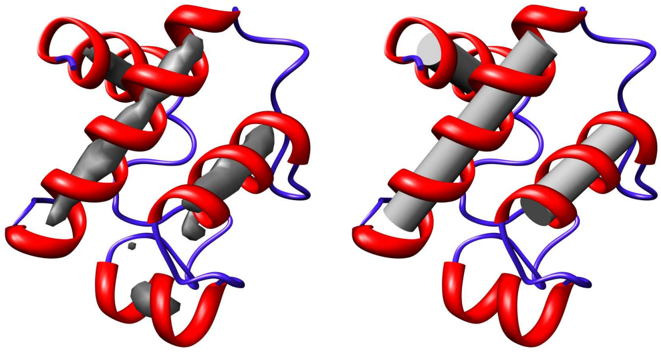
Tensor skeleton. The tensor skeleton approach provides a fine accuracy for the α protein group, with a low number of FP errors, and limited FN errors. When applied to the $\alpha\&\beta$ group, the number of FN errors is still low, however the FP errors are widespread in most proteins, which become the predominant factor of all errors. Another problem for Tensor skeleton approach is that its identified cylinders are often deflected away from the real helix positions. Figure 9(a) shows the groundtruth derived from the *pdb* file, please notice the blue structure on the top left corner, it looks quite similar to a helix, but in actuality it is not. However it probably has similar local structure tensor as the helix structure, when the Tensor skeleton approach is searching helix-like structures voxel by voxel, the blue structure is also recognized as part of the helix. Finally the identified helix (cylinder) is deflected away from the real position as in Figure 9(c). By contrast, the identification result of RENNSH approach in Figure 9(d) gives the accurate position of the helix. This kind of accuracy difference cannot be reflected by FN or FP, however it could be expressed by the E_{GO} energy defined in section 3.4. The result of Helixhunter method is shown in Figure 9(b).



(a) Pdb (b) Helixhunter (c) Tensor skeleton (d) RENNSH

Figure 9: Identification results for one helix of protein 1BGD

RENNSH. For the α protein group the RENNSH approach provides excellent identification results, 365 of 370 helices are successfully identified. For the $\alpha\&\beta$ group, only 1 FP error shows up, the relatively higher number of FN errors mainly comes from the short helices with only one or two turns. Figure 10 shows the identification results of protein 1CYJ with RENNSH approach. Notice the small helix at the bottom of Figure 10(a). There are still some nonzero voxels around its helix axis in the helix voxel map (see section 3.2), which means the iterative kNN classification method based on SHD works fine. The lost of this helix happens in the final pruning stage, because they are considered not long enough to represent a real helix. This problem could be resolved by adding a new class in the training process, which contains the SHDs from the end voxels of each helix, to increase the nonzero voxels in the helix voxel map and finally prolong the real helix (cylinder). Another important advantage of RENNSH approach is that once all the parameters are determined with E_{GO} energy according to an appropriately chosen training database, they are robust enough to be used for any query density maps. By contrast, the parameters of Helixhunter and Tensor skeleton methods need more adjustments to achieve the best identified results.



(a) Helix voxel map (b) Output

Figure 10: Identification result of protein 1CYJ with RENNSH approach

Break errors. The break errors appear in both Helixhunter and RENNSH approaches. They do not show in the Tensor skeleton approach, because it uses a skeleton-based search algorithm, in which the helix-like voxels are connected with each other as soon as they are recognized. By contrast, the RENNSH is a real voxelwise-based approach, in which the helix-like voxels are first identified independently and then connected together with the ε-connected component algorithm. If there are some voxels missing in the middle of a helix and the missing space is longer than the threshold ε, a integrated helix will then breaks into two or more segments in the final output. However, we could see from Table 1 that the Break error number of RENNSH is not high, because missing voxels normally only show up in *abnormal* helices. For example, the black circle in Figure 11 shows an abnormal helix structure of protein 1ACJ. SHDs of the voxels in that area will not be found in the helix axis class. This problem could also be solved by generating a new training class to store all SHDs from abnormal helix structures. At last we want to mention that the break errors may not be distinguished by the E_{GO} energy (see Section 3.4), since the E_G and E_O energies could still be low if only several voxels are missing.

5 Conclusions

We have proposed a novel α-helices identification approach for intermediate resolution cryo-EM maps. By representing the structure property of helix axis voxels with rotation invariant SHD transforms, the helices identification task has been transformed into a pure pattern recognition problem. An iterative kNN classification process is elaborately designed, which can be extended to other pattern recognition applications. The proposed energy function E_{GO} can not only be used for parameter optimization, but also could serve as a numerical criterion for evaluating different helices identification methods. The identification result shows that RENNSH has bet-

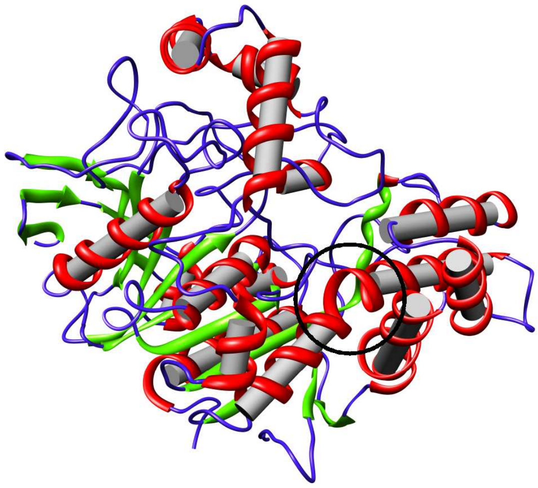


Figure 11: Abnormal helix structure of 1ACJ

ter accuracy than existing methods, and its parameters are much more robust against different queries.

Future work includes: test RENNSH approach on the real cryo-EM reconstructed maps; transform different output formats of each method into uniform, then optimize the parameters with E_{GO} energy; utilize the E_{GO} energy as an numerical criterion to evaluate different identification methods; establish new SHD classes of helix ending voxels and abnormal helix axis voxels and integrate them into RENNSH approach.

References

[Bajaj 07] C.L. Bajaj. *AUTOMATIC STRUCTURE INTERPRETATION OF SINGLE PARTICLE CRYO-ELECTRON MICROSCOPY: FROM IMAGES TO PSUEDO-ATOMIC MODELS*. In Biomedical Imaging: From Nano to Macro, 2007. ISBI 2007. 4th IEEE International Symposium on, pages 236–239, April 2007.

[Berman 00] Helen M. Berman, John Westbrook, Zukang Feng, Gary Gilliland, T. N. Bhat, Helge Weissig, Ilya N. Shindyalov & Philip E. Bourne. *The Protein Data Bank*. Nucleic Acids Res, vol. 28, pages 235–242, 2000.

[Christopher 96] Jon A. Christopher, Rosemarie Swanson & Thomas O. Baldwin. *Algorithms for finding the axis of a helix: Fast rotational and parametric least-squares methods*. Computers & Chemistry, vol. 20, no. 3, pages 339 – 345, 1996.

[Dutta 05] Shuchismita Dutta & Helen M. Berman. *Large Macromolecular Complexes in the Protein Data Bank: A Status Report*. Structure, vol. 13, no. 3, pages 381 – 388, 2005.

[Ferentz 00] Ann E. Ferentz & Gerhard Wagner. *NMR spectroscopy: a multifaceted approach to macromolecular structure*. Quarterly Reviews of Biophysics, vol. 33, no. 01, pages 29–65, 2000.

[Healy 03] D. M. Healy, D. N. Rockmore, P. J. Kostelec & S. Moore. *FFTs for the 2-Sphere-Improvements and Variations*. Journal of Fourier Analysis and Applications, vol. 9, no. 4, pages 341–385, July 2003.

[Jiang 01] Wen Jiang, Matthew L. Baker, Steven J. Ludtke & Wah Chiu. *Bridging the information gap: computational tools for intermediate resolution structure interpretation*. Journal of Molecular Biology, vol. 308, no. 5, pages 1033 – 1044, 2001.

[Kahn 89] Peter C. Kahn. *Defining the axis of a helix*. Computers & Chemistry, vol. 13, no. 3, pages 185–189, 1989.

[Kazhdan 03] M. Kazhdan, T. Funkhouser & S. Rusinkiewicz. *Rotation invariant spherical harmonic representation of 3D shape descriptors*, 2003.

[Laga 06] H. Laga, H. Takahashi & M. Nakajima. *Spherical Wavelet Descriptors for Content-based 3D Model Retrieval*. page 15, 2006.

[Lasker 07] K. Lasker, O. Dror, M. Shatsky, R. Nussinov & H.J. Wolfson. *EMatch: Discovery of High Resolution Structural Homologues of Protein Domains in Intermediate Resolution Cryo-EM Maps*. Computational Biology and Bioinformatics, IEEE/ACM Transactions on, vol. 4, no. 1, pages 28–39, Jan.-March 2007.

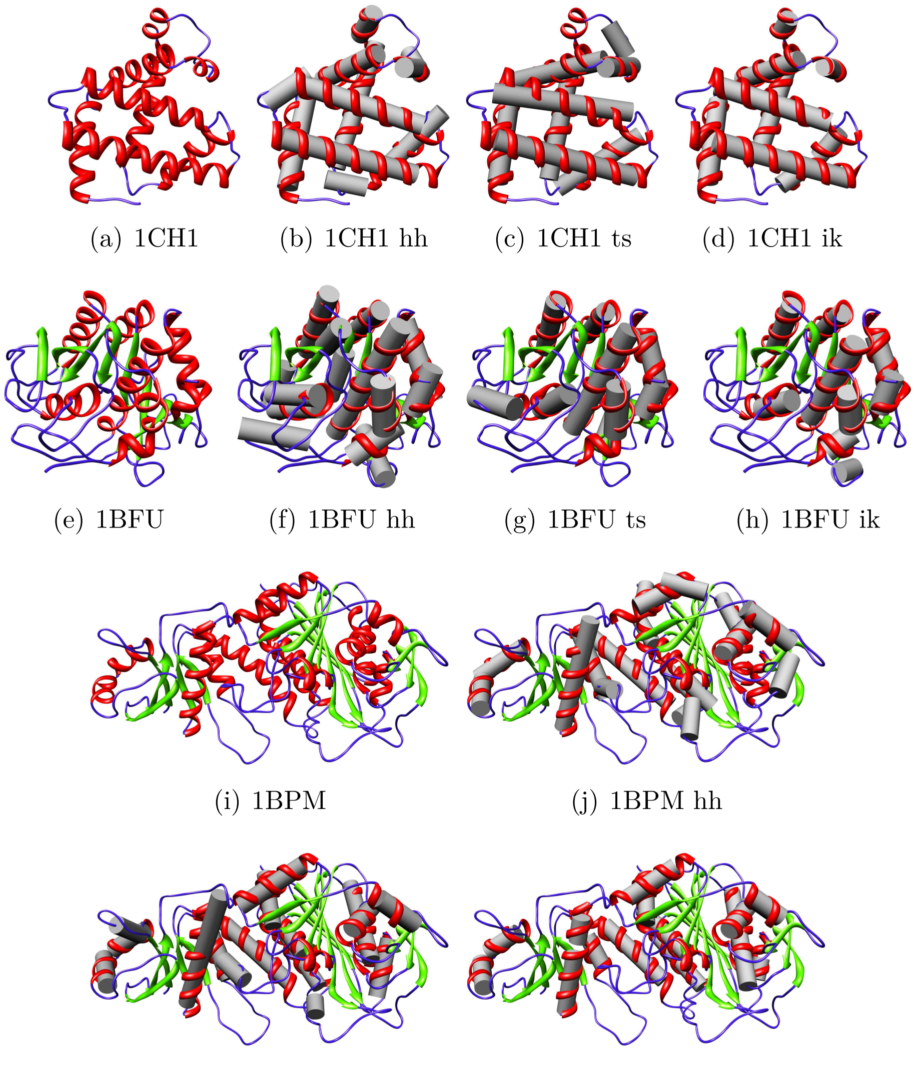
[van Heel 00] Marin van Heel, Brent Gowen, Rishi Matadeen, Elena V. Orlova, Robert Finn, Tillmann Pape, Dana Cohen, Holger Stark, Ralf Schmidt, Michael Schatz & Ardan Patwardhan. *Single-particle electron cryo-microscopy: towards atomic resolution*. Quarterly Reviews of Biophysics, vol. 33, no. 04, pages 307–369, 2000.

[Vilenkin 68] N.J. Vilenkin. *Special Functions and the Theory of Group Representations*. American mathematical Society, Providence, 1968.

[Woolfson 97] M. Woolfson. An introduction to x-ray crystallography. Cambridge Univ. Press, Jan. 1997.

[Yu 05] Z. Yu & C. Bajaj. *Automatic ultrastructure segmentation of reconstructed CryoEM maps of icosahedral viruses*. Image Processing, IEEE Transactions on, vol. 14, no. 9, pages 1324–1337, Sept. 2005.

[Yu 08] Zeyun Yu & Chandrajit L. Bajaj. *Computational Approaches for Automatic Structural Analysis of Large Biomolecular Complexes*. IEEE/ACM Trans. Comput. Biology Bioinform., vol. 5, no. 4, pages 568–582, 2008.



(a) 1CH1 (b) 1CH1 hh (c) 1CH1 ts (d) 1CH1 ik

(e) 1BFU (f) 1BFU hh (g) 1BFU ts (h) 1BFU ik

(i) 1BPM (j) 1BPM hh

(k) 1BPM ts (l) 1BPM ik

hh, ts and ik denote the identification results of Helixhunter, Tensor skeleton and RENNSH respectively

Figure 12: Identification results of different methods

	PDB	#	Helixhunter				Tensor skeleton			IKVS			
			FN	FP	B	Total	FN	FP	Total	FN	FP	B	Total
	1B6Q	2											
	1A7W	3											
	1AIL	3											
	1EDI	3	3			3							
	1ENH	3		2		2							
	1A32	4	1			1							
	1CCD	4											
	1CYJ	4	1			1	1		1	1			1
	1EOQ	4		1		1							
	1AEP	5			2	2							
	1B68	5	1	1		2							
	1BGD	5	1	1		2		1	2		1		1
	1DOV	5	2			2							
	1A17	7	2		1	3							
	1EBC	7		2		2	1		1	1			1
	1CH1	8		3		3	1	1	2			1	1
	1CH2	8	2	1		3	1	1	2				
	1CH7	8		2		2	1		1				
α	1CH9	8	3	3		6	1		1				
	1CIO	8		2	1	3	1	1	2				
	1CO8	8	1	2	1	4	1		1				
	1CP0	8		3	1	4	2		2				
	1CPW	8		3	1	4	1		1				
	1DWR	8		1	1	2	1		1				
	1DWT	8		1	2	3	2	1	3		1		1
	1BJ0	10		1		1	1	1	2				
	1AII	19											
	1ANN	19			1	1							
	1AOW	19					1		1				
	1AXN	19		2	1	3		1	1				
	1A8A	20	1	3		4							
	1A8B	20	1	2	2	5							
	1BC0	20		1	2	3							
	1BC3	20	1	3	1	5							
	1BCW	20	1		1	2	1		1				
	1BCY	20	2	3	1	6							
	1BCZ	20	1				1	1		1			
α S	37	370	24	43	19	86	18	7	25	2	2	1	5
			76.76%				93.24%			98.65%			
	1AIU	4		2		2							
	1AZL	4		1	2	3		2	2				
	1ASU	5	1	3		4		2	3				
	1AUC	5		2		2	1	1	2				
	1C4W	5						2	2				
	1BLE	7	1	2		3		1	1	1			1
	1AMP	8	1	6		7		4	5				
	1AVT	8	2	2		4		2	2				
	1BED	8	2			2							
	1BFU	8	2	4		6		2	2		1		1
	1BF2	12	3	11		14	2	6	9	3			3
	1BYA	12		11		11							
$\alpha\&\beta$	1BYB	12	3	7		10		4	4			1	1
	1BYC	12	3	4		7		4	4				
	1BYD	12	3	5		8		4	4				
	1CDG	12	7	8		15	3	3	6	2			2
	1ACJ	15	3	1		4	3	1	5	1		1	2
	1ACL	15	3	1		4	3	3	6	2		1	3
	1AKB	15	7			7	1	3	4	2			2
	1AKC	15	4		1	5	1	1	2	2			2
	1AMQ	15	2	2	1	5		1	1				
	1AMR	15	3	2	1	6		1	1				
	1AMS	15	3	2		5		2	3				
	1BPM	15	3	5	1	9	2	3	5	1		1	2
	1AMA	16	4			4	2	1	3	3			3
	1C4K	27	7	6		13	5	5	10	4			4
$\alpha\&\beta$ S	26	297	67	87	6	160	25	59	84	21	1	4	26
			46.13%				71.72%			91.25%			
Total	63	667	91	120	25	246	43	66	109	23	3	5	31
			63.12%				83.66%			95.35%			

\# : total number of helices in PDB
FN : false negative or missed helices
FP : false positive or wrongly deteceted helices
B(Break) : one integrated helix breakes into several segments
α : the protein contains only α-helices
α S : statistics of all the proteins, which only contain α-helices
$\alpha\&\beta$: the protein contains both α-helices and β-sheets
$\alpha\&\beta$ S : statistics of all the proteins, which contain both α-helices and β-sheets

Table 1: α-helices identification results of Helixhunter, Tensor skeleton and IKVS