Computational Approaches for Automatic Structural Analysis of Large Bio-molecular Complexes

Zeyun Yu, *Student Member, IEEE,* Chandrajit Bajaj, *Member, IEEE,*

***Abstract*— We present computational solutions to two problems of macromolecular structure interpretation from reconstructed three-dimensional electron microscopy (3D-EM) maps of large bio-molecular complexes at intermediate resolution (5Å-15Å). The two problems addressed are: (a) 3D structural alignment (matching) between identified and segmented 3D maps of structure units (e.g. trimeric configuration of proteins), and (b) the secondary structure identification of a segmented protein 3D map (i.e. locations of α-helices, β-sheets). For problem (a), we present an efficient algorithm to correlate spatially (and structurally) two 3D maps of structure units. Besides providing a similarity score between structure units, the algorithm yields an effective technique for resolution refinement of repeated structure units, by 3D alignment and averaging. For problem (b), we present an efficient algorithm to compute eigenvalues and link eigenvectors of a Gaussian convoluted structure tensor derived from the protein 3D Map, thereby identifying and locating secondary structural motifs of proteins. The efficiency and performance of our approach is demonstrated on several experimentally reconstructed 3D maps of virus capsid shells from single-particle cryo-EM, as well as computationally simulated protein structure density 3D maps generated from protein model entries in the Protein Data Bank.**

***Index Terms*— Structure Analysis, Alignment, Similarity Measure, Segmentation, Secondary Structure Detection, Skeletonization, Cryo-EM Maps, 3D Reconstruction**

I. Introduction

X-RAY crystallography [1], [2] and nuclear magnetic resonance (NMR) [3], [4] are two widely used techniques that are used to reveal the structure of individual proteins. These structural models are then deposited in the Protein Data Bank (PDB) [5]. While individual proteins provide important structural information about our bodies' fundamental blocks, the structural determination of large biological complexes (e.g., viruses, ion channels, and the ribosome) offer a more complete description of the protein machinery of life. Detailed structural knowledge of these complexes not only provides mechanistic descriptions of how macromolecules interact in an assembly, but also yields clues for developing therapeutic interventions related to disease. While x-ray crystallography and NMR spectroscopy are often restricted to relatively small biological structures, cryo-electron microscopy (cryo-EM) of single particles has emerged as a powerful technique in revealing the ultra-structure of large bio-molecular complexes [6], [7]. The rapid development of improved image acquisitions and sophisticated computational signal and image processing methods have now made it possible to resolve these large biological complexes at sub-nanometer resolutions (6Å-10Å) [8], [9], [10], [11].

Computational signal and image processing have been used extensively in 3D biological structure reconstruction and at multiple scales (tissue, cell, molecular) [12]. In particular, most modern signal/image processing algorithms such as 2D image restoration, noise reduction, contrast enhancement, feature detection, alignment, classification, 3D reconstruction, boundary segmentation, and skeletonization, have found fruitful applications in the single particle cryo-EM approach, as will be briefly reviewed in Section II. Although a number of software packages for performing 3D reconstructions from cryo-EM data have been made widely available in recent years (e.g., EMAN [13], SPIDER [14], IMAGIC [15]), quantitative and automatic analysis/interpretation techniques operating on the reconstructed 3D electron density maps of bio-molecular assemblies, remain largely undeveloped. Current methods for interpreting reconstructed 3D maps depend primarily upon manual selection and visual inspection with the help of interactive graphic tools. Due to the large physical size and structural complexity of the bio-molecular assemblies, however, manual processing is tedious and subjective. Automatic structural interpretation and analysis of 3D maps of large bio-molecular complexes have thus become the preferred avenue of research and development.

In prior work [16], [17], we presented an automatic approach to segmenting the reconstructed 3D maps of bio-molecular complexes into dozens to thousands of individual structure units. This automatic segmentation makes it considerably easier to determine the structural interactions of the bio-molecular assembly. Furthermore, with segmentation we are able to isolate the various grouped protein structure units (also called conformers), and interpret the ultra-structure of each of the proteins individually. For symmetric structures, such as most protein capsid shells of viruses [6], the segmentation of the entire capsid into asymmetric protein structure units also helps eliminate structural representation redundancy. In this paper, we present two additional automatic algorithms of protein structure interpretation from reconstructed 3D Maps. The first algorithm is for automatic spatial alignment, and computation of structural similarity score, between two 3D segmented structure units. The second algorithm automatically identifies and locates secondary structure units (i.e., protein α-helices, β-sheets) within a segmented protein structure unit. The accuracy and hence the need for the identification of protein α-helices and β-sheets increases correspondingly with the

Z. Yu and C. Bajaj are with the Computational Visualization Center, Department of Computer Sciences and The Institute of Computational Engineering and Sciences, The University of Texas at Austin, 1 University Station C0500, Austin, Texas 78712, USA. (E-mails: zeyun.yu@gmail.com, bajaj@cs.utexas.edu).

increased rise in resolution and availability of reconstructed 3D maps at sub-nanometer resolution (i.e., better than $10\AA$) [8]. Clearly, the better the understanding of the macromolecular ultra-structure the better the ability to determine its structure-function relationships.

The rest of this paper is organized as follows. Section II provides a brief introduction to 3D electron microscopy imaging and 3D map reconstruction. In Section III we present our automatic 3D structure unit alignment and similarity scoring algorithm, along with with applications to related problems. We present our algorithm of automatic protein secondary structure identification and localization in Section IV, along with several example results of our implementation on both reconstructed and simulated 3D Maps.

II. 3D Cryo-Electron Microscopy Imaging and Reconstruction

Electron microscopy (EM) imaging has been extensively used in structural biology to study the activities of cells and organelles. Three-dimensional electron microscopy (3D-EM) imaging plays a unique role in structural biology, thanks to its remarkable capability to reveal the three dimensional structure of biological entities. The mathematical principles of 3D-EM reconstruction from projection data (experimental EM images) is basically the same as that used in 3D Computed Tomography (CT) medical imaging. The major difficulty with EM images is the extremely low signal-to-noise ratio (SNR). This is true partly because the electron dose used in EM imaging has to be kept to an extremely low level (approximately 0.5$\sim$4 e/$\AA^2$) in order to reduce the radiation damage to the specimen. The flash cooling technique, known as cryo-EM, quickly cools the samples under study, to liquid nitrogen temperature (about 77 Kelvin or less) such that the surrounding water does not form crystalline ice, but remains in a vitreous state. Thus cryo-EM has proved to be quite successful and has hence gained a growing popularity in structural biology for its capability of preserving the native in-vivo structure of biological specimens while reducing the damage caused to the specimens [6], [18].

Using different sample preparations and data collection methods, 3D-EM encompasses three major techniques: electron crystallography, electron tomography, and the single particle cryo-EM method. Electron crystallography [19], similar to X-ray crystallography, can reveal the bio-molecular structures at near atomic resolution. However, the weakness of this technique is that a two-dimensional crystal has to be grown for cryo-EM imaging, which in many cases is difficult to do, especially for large macromolecular complexes. Electron tomography [20], [21] is technically very similar to CT and is used to study 3D ultra-structures of cell organelles or whole cells at relatively low resolutions. Mathematically, the 3D structure of a cell specimen can be reconstructed from a series of 2D projections generated from different tilt angles. There are several methods to collect the projection data: single-axis tilt, double-axis tilt, and uniform conical tilt, depending on how the specimen is rotated under the fixed camera. However, the tilt cannot exceed certain angles (usually $\pm 70^0$) due to the relative orientation of the tilt-stage and limitations on specimen thickness [22], [23]. For this reason, the reconstructed density maps always have significant distortions in certain regions, commonly known as the missing wedge (pyramid, or cone) problem. Partly due to this problem, and partly because of the limited electron dose used on a single specimen with consideration to the radiation damage, the resolution of this type of reconstructions is often limited to the range of $20\AA \sim 200\AA$ [20], [21].

The above resolution problems could be resolved if we had a number of structurally identical particles at multiple random orientations and found a way to align and average them together to yield a single 3D structural image. On the one hand, the average of particles in the same orientation (after proper alignment) can improve the signal-to-noise ratio (SNR). On the other hand, particles in different orientations are very likely to complement each other such that the missing wedge (pyramid, or cone) of projection orientations of one particle could be filled by the other particles. This technique, known as *single particle cryo-EM reconstruction*, has been used to resolve 3D structures at about $6\AA$ [8], [24]. To reduce the radiation damage, each particle is only imaged from a single tilt angle, but thousands of particles are used to reconstruct a single 3D structure. Fig. 1(a) shows the overall pipeline of single particle structure reconstruction and analysis. Starting from 2D digitized microscopy images, the 3D structure map reconstruction includes several major steps:

- *Particle Picking*. The goal of particle picking is to box out all particles that look reasonably good as projection of particles, rather than noise, in both size and shape [25], [26], [27]. Fig. 1(b) shows a small portion of the electron micrograph of the rice dwarf virus [8], from which we can see how noisy a typical particle image looks.
- *Particle Classification and Alignment*. Particle classification groups all the particle images that have the same appearance but do not have to be in the same orientation. The classified particles can be aligned and averaged such that the class averages have significantly improved signal-to-noise ratios (SNR) [23], [28], [29].
- *Orientation Assignment*. Since in most cases the particles appear in "in-vivo" random orientations, we do not know the orientation of each particle image. To assign the orientations, there are direct methods based on the *common line theorem* [7] and iterative methods based on initial 3D models [13].
- *Reconstruction and Refinement*. The mathematical theorem is well established for 3D reconstruction from 2D projections, given that the orientation of each projection is known. The most popular methods include the direct Fourier space reconstruction [13] and the real-space filtered back projection method [7]. Fig. 1(b) illustrates the 3D reconstruction from a series of projections at different angles. The reconstructed 3D map can be used as an initial model to refine the particle classification, alignment, and orientation assignments.

The reconstructed 3D maps do not convey meaningful information unless they are correctly interpreted. The ultra-structure of the 3D maps can be interpreted in two ways as shown in Fig. 1(a). For maps at intermediate resolutions ($6\AA - 10\AA$), the secondary structures are visually identifiable and computationally auto-detectable. A pseudo-atomic model can be built based on the secondary structure elements detected and their topological connections [8]. When the resolution of the reconstructed 3D maps degrades beyond $10\AA$, however, we cannot discern the secondary structure elements with high confidence but we still can attempt to construct a pseudo-atomic model by matching and fitting a high-resolution X-ray structure model (from the PDB) into the 3D map based on a density distribution and correlation [30]. In either case, the prior segmentation of the 3D map into individual protein structure units is both meaningful and necessary for fast and accurate ultra-structure interpretation [16].

In the subsequent sections, we present computational approaches for further automatic structural interpretation of 3D maps, namely, (a) an automatic 3D structure unit alignment coupled to structural similarity scoring, and (b) automatic protein secondary structure identification and localization.

III. 3D Structure Alignment

While our segmentation algorithm [16], [17] can decompose 3D virus maps into individual structure units (also called *subunits* below), it does not tell us how different the segmented subunits are. From the structural point of view, it is important to know the similarity between the segmented subunits and quite often useful to average the subunits of high similarity in order to improve the signal-to-noise ratio. To this end, we develop a fast algorithm to align the segmented subunits such that the similarity measure and averaging can be conducted between the spatially aligned subunits. In addition, as we shall see below, knowing the alignments between subunits can also improve the accuracy of our segmentation algorithm and, coupled with the structural fitting approaches, simplify the pseudo-atomic modeling of large bio-molecular complexes.

Technically the goal of 3D structure alignment is to find the transformation matrix from one 3D structural unit to another, such that the two 3D maps are best matched according to a similarity scoring function. In the following we first define the similarity scoring we use between two 3D maps and then present a fast algorithm for computing the transformation matrix aligning two 3D maps of structure units. We also present a number of further applications of this structural alignment.

A. Similarity Scoring Function

A traditional similarity scoring function between two 3D maps, denoted by f and g, is defined by cross-correlation as follows:

$$S^1_{f,g}(T) = \sum_{i,j,k} f(i,j,k) \times g(T(i,j,k)), \tag{1}$$

where T is a 4×4 matrix using homogeneous coordinates [34]. It is intuitively treated here as a function that transforms the coordinate system of f to that of g. The 3D structural alignment problem is then reduced to determining the best T, given two maps f and g, such that the scoring function $S^1_{f,g}(T)$ achieves its maximum. It is easy to see that maximizing $S^1_{f,g}(T)$ is equivalent to minimizing the square difference between f and g:

$$S^2_{f,g}(T) = \sum_{i,j,k} (f(i,j,k) - g(T(i,j,k)))^2. \tag{2}$$

The advantage of the cross-correlation method is that the fast Fourier transform (FFT) can be deployed to speed up the search in the relative translational space (three degrees of freedom). However, the search in rotational space has to be performed in a conventional way. As we shall see, our 3D alignment between two structure units involves both translation and rotation, but each of these is restricted to one degree of freedom. In this particular case, the real space approach, if properly utilized, can be more efficient than the FFT method. We shall explain more in Section III-D.

Our similarity scoring function is defined in real space and related to Equation 2. The intuition is to minimize the difference between two maps. To speed up the search, however, we compute the similarity score on a set of critical points of the 3D maps, instead of the entire collection of voxels that make up the 3D maps of the structure units. The critical points are those that best capture the features of a molecular density 3D map. In general, critical points include the local maxima, local minima, and saddle points of a scalar 3D map. In our experiments we define as critical points the local maxima with intensity values higher than a user-defined threshold (details demonstrated below). If the data is noisy, one initially preprocesses the 3D map using gradient vector diffusion as discussed in [16].

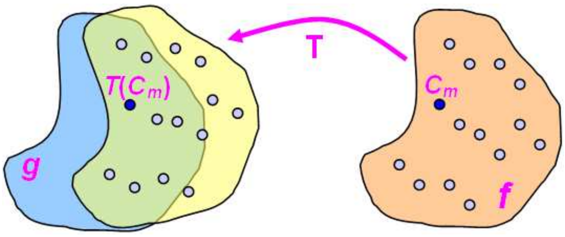
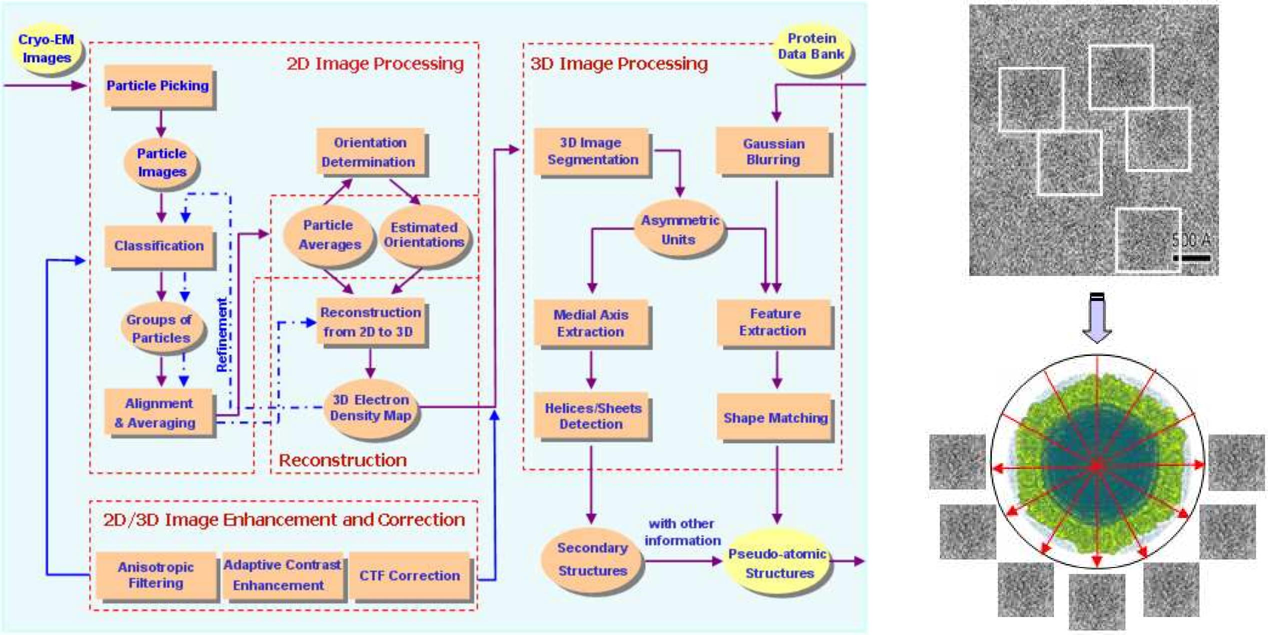


Fig. 2. Illustration of similarity calculation based on critical points. Every critical point c_m, $(m = 1, 2, \cdots, M,)$ in f is transformed to g according to T and the difference between f at c_m and g at $T(c_m)$ is calculated and summed. If the two maps are identical and the matching is perfect, the total difference should return zero. Similarly, we calculate the total difference between the densities at $d_n, n = 1, 2, \cdots, N$, in map g and their transformed positions in f. The final normalized similarity score is given in Equation 3.

Our similarity scoring function is thus defined in the following way:

$$S^3_{f,g}(T) = 1 - \frac{\sum_{m=1}^{M} |f(c_m) - g(T(c_m))| + \sum_{n=1}^{N} |f(T^{-1}(d_n)) - g(d_n)|}{\sum_{m=1}^{M} max\{f(c_m),\ g(T(c_m))\} + \sum_{n=1}^{N} max\{f(T^{-1}(d_n)),\ g(d_n)\}}, \tag{3}$$

where $c_m, m = 1, 2, \cdots, M$, are critical points of f and $d_n, n = 1, 2, \cdots, N$, are critical points of g. Fig. 2 illustrates the idea of this similarity scoring function. It is worth noting that there is no direct relationship between the number of critical points and the size of the map $-$ a large map with slowly-varying



(a) Pipeline of the single particle reconstruction and analysis

(b) An example

Fig. 1. Illustrations of single particle cryo-EM technique. (a) The overall pipeline, including particle picking, particle classification/alignment, orientation assignment, 3D reconstruction, and map interpretation. In addition, EM contrast transfer function (CTF) correction [31], anisotropic filtering [32], and adaptive contrast enhancement [33] may also be applied before or after 3D reconstructions, to improve signal-to-noise ratio. (b) An example of electron microscopy (EM) images showing particle picking (top) and 3D reconstruction from 2D projections (bottom).

densities may have a small number of critical points, while a small map with sharp density variations can have a large number of critical points. There are two major differences between $S^3_{f,g}(T)$ and $S^2_{f,g}(T)$. First, the new similarity scoring function is based only on the critical points. Therefore, the search for the best T using $S^3_{f,g}(T)$ is much faster than the search using $S^2_{f,g}(T)$, as the latter is based on all the voxels of f and g. Secondly, the scoring function of $S^3_{f,g}(T)$ is normalized such that the similarity scores are always scaled to the range of [0, 1], where 0 means no similarity and 1 signifies the highest similarity.

B. 3D Alignment Algorithm

In our previous paper [16] we discussed how to detect automatically the local symmetry axes of protein conformers in a reconstructed 3D density map of a macromolecule, and how to segment each of the individual locally symmetric structure units that comprise the building blocks of that macromolecule. Given two such individual structure units A and B, as shown in blue and magenta respectively in Fig. 3, our problem is to transform A to B in four steps, based on the symmetry axes computed in [16]:

1) Translate A by t_0.
2) Rotate A by r_0.
3) Translate A by t.
4) Rotate A by r.

Since the symmetry axes of both A and B are given, the first two of the above transformations, the translation t_0 and the rotation r_0, are uniquely determined by the relative position/orientation of the symmetry axes of A and B. More specifically, since each symmetry axis is given by two end points, we translate A from point S_A to point S_B, followed by the rotation between the two dashed blue arrows in Fig. 3. However, the translation t and the rotation r have to be decided based on the similarity scores between the density maps of A and B as discussed in Section III-A. Therefore, the transformation from one structure unit to another has two degrees of freedom: one translation and one rotation, as illustrated in Fig. 3.

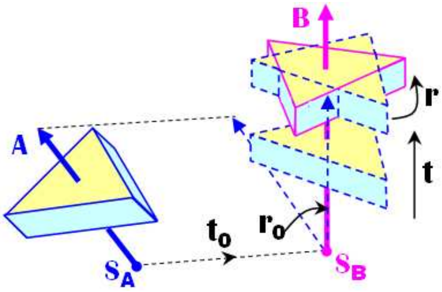


Fig. 3. The alignment between two structure units include translations t_0 and t, and rotations r_0 and r. But only t and r are unknown and need to be determined based on the similarity scoring function.

Putting the four transformation matrices together, we have the following matrix that transforms subunit A to subunit B:

$$T_{t,r} = M_4(r) \times M_3(t) \times M_2(r_0) \times M_1(t_0) \quad (4)$$

The matrices M_1, M_2, M_3, M_4 are conventional transformation matrices for translations or rotations. One can easily derive the exact expressions for these matrices based on the given information (i.e., symmetry axes with start/end points). Substituting Equation 4 into Equation 3, we have the following maximization equation for the 3D alignment between structure units A and B, with density functions f and g respectively:

$$\max_{\{t,r\}}\{S^3_{f,g}(t,r) = 1 - \frac{\sum_{m=1}^{M}|f(c_m)-g(T_{t,r}(c_m))|+\sum_{n=1}^{N}|f(T_{t,r}^{-1}(d_n))-g(d_n)|}{\sum_{m=1}^{M}max\{f(c_m),\ g(T_{t,r}(c_m))\}+\sum_{n=1}^{N}max\{f(T_{t,r}^{-1}(d_n)),\ g(d_n)\}}\}, \tag{5}$$

where $T_{t,r}$ is the transformation matrix from A to B as defined in Equation 4. Since the scoring function defined in Equation 5 indicates the similarity between two structure units, the transformation matrix that maximizes Equation 5 should give the best alignment between the two structure units. The range of t is user-defined. In our experiments we assumed that $t \in [-10,10]$ in pixel unit. The range of r is from 0° to 360° for non-symmetric objects. However many of the reconstructed 3D maps from single particle cryo-EM are virus capsid structures, which typically have icosahedral symmetries. Therefore, the segmented structure units usually have n-fold symmetry and r values in the range of 0° to $\left(\frac{360}{n}\right)^\circ$.

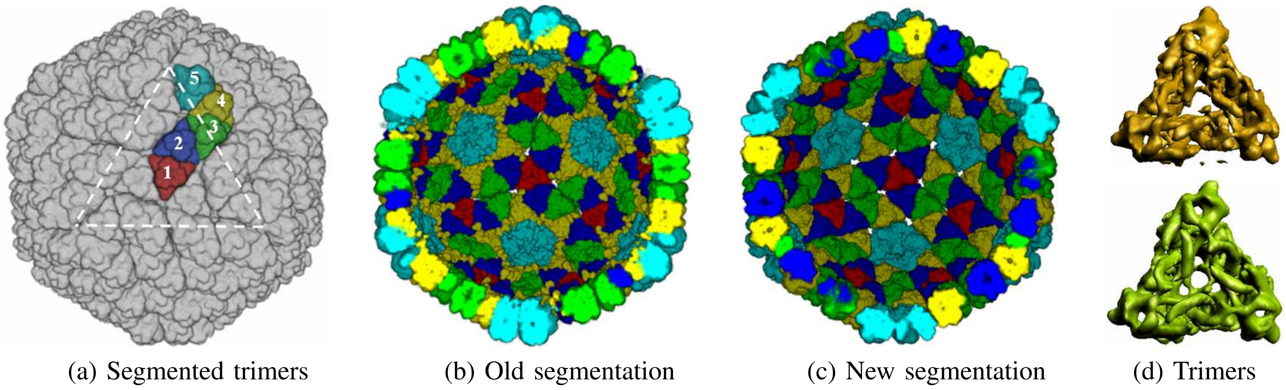
There are a number of optimization techniques to find the $\{t,r\}$ that maximize the similarity score between A and B as defined in Equation 5. In order to find the best $\{t,r\}$, we need to calculate the similarity score at values of t and r sampled regularly over the ranges discussed above. Dense samplings could yield more accurate t and r but require more computational time. In contrast, coarse samplings are faster but less accurate. We adopt a simple two-level hierarchical method for our search. On the coarse level, the translation variable t is sampled by every one pixel unit from -10 to 10 and the rotation variable r is sampled by every 5° from 0° to $\left(\frac{360}{n}\right)^\circ$ where n is the folding number of the symmetry. For each combination of these t and r, we calculate the similarity score: the one that maximizes the scoring function is taken to be the solution of Equation 5 on the coarse level, denoted by $\{t^{(1)},r^{(1)}\}$. On the fine level, the sampling is taken within a small range around $t^{(1)}$ and $r^{(1)}$. For the translation, the sampling is taken on the interval $[t^{(1)}-0.9, t^{(1)}+0.9]$ by every 0.1 pixel unit. The sampling for rotation is on the interval $[r^{(1)}-4^\circ, r^{(1)}+4^\circ]$ by every 1°, where 4° is used because the sampling rate we chose on the coarse level was 5°. Again, the similarity score for each of these samples is calculated and the best one, denoted by $\{t^{(2)},r^{(2)}\}$, gives the final solution of Equation 5. By substituting $\{t^{(2)},r^{(2)}\}$ into Equation 4, we have the transform matrix $T(t^{(2)},r^{(2)})$, which gives the 3D alignment from subunit A to B. A couple of applications of this 3D alignment are presented next.

C. Applications

We consider two examples in this section. The first one is the rice dwarf virus (RDV), which has icosahedral symmetry at a resolution of $6.8\AA$ [8]. There are five independent structure units with 3-fold local symmetry, also known as trimers, as shown in Fig. 4(a). The second example is the bacteriophage $\phi 29$ at a resolution of $15\AA$ [35]. The major components of $\phi 29$ include a major capsid, a tail, a head-tail connector, and DNAs [36]. Since in the cryo-EM map presented below, a 5-fold global symmetry is imposed along the vertical axis, only the major capsid, thanks to its 5-fold global symmetry, is well preserved during the reconstruction. For this reason, we shall consider only the major capsid in the following. As shown in Fig. 5(a), there are ten independent structure units including one tail on the bottom (structure unit #3), three 5-fold structure units or pentons (#0, #1, #2), and six 6-fold structure units or hexons (#4 $\sim$ #9).

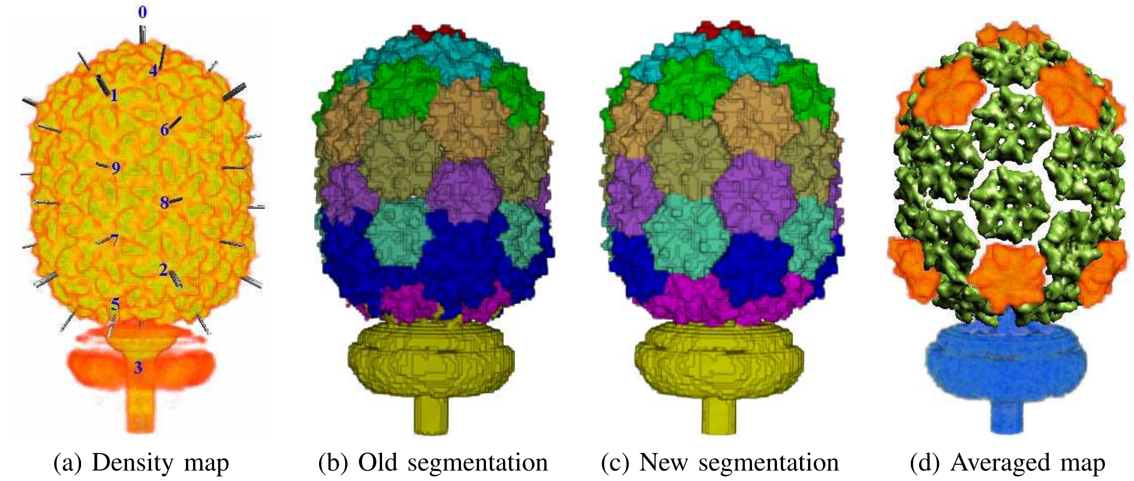
1) Improving Subunit Segmentation: Our previous segmentation algorithm [16] is based on the multi-seeded fast marching method [37]. In this method each subunit is assigned an initial contour which keeps growing according to a pre-defined speed function until it stops on the boundaries between structure units. When the structure is symmetric, some of the structure units are dependent (identical) and hence the contours corresponding to those structure units must grow simultaneously by the same amount in the same way. With this constraint, our previous segmentation [16] incorporates global symmetry (for example, icosahedral for RDV and 5-fold for $\phi 29$) and the local n-fold symmetry of each subunit into the segmentation. However, we did not consider the structural similarity between the independent structure units. In other words, the contours corresponding to independent structure units grew independently. This can sometimes yield noticeable errors especially when the boundaries between structure units are indistinguishable. An example is shown in Fig. 4(b). This is the segmentation of the outer layer of the rice dwarf virus (RDV) and the five types of trimers are colored differently (see Fig. 4(a) for their relative locations). We can see that the structure units in different colors are not consistent with each other. For example, the yellow structure units occupy more space than the others. To remedy this problem, we utilize the 3D structure unit alignment discussed in Section III-B, such that the contours within all the structure units grow simultaneously by the same amounts in the properly-aligned directions. The results with this additional constraint are shown in Fig. 4(c). We can see that the structure units in the new results look much more uniform and consistent than those in Fig. 4(b). The segmentation of $\phi 29$ is given in Fig. 5(b) without alignment and in Fig. 5(c) with alignment. We can also see that the alignment-based segmentation demonstrates much better consistency between the segmented structure units. However, it should be pointed out that the structure units must be segmented before we can align them. Therefore, our previous segmentation algorithm [16] is still fundamental in providing us with the initial segmentation from which the structure units can be aligned. Iteratively the new segmentation could also be used to refine the alignment but the improvement is not significant.

2) Quantifying 3D Structural Similarity: Another application of our 3D alignment algorithm is a similarity measure between structure units. This is functionally very important because it provides us with the structural similarity quantitatively between any two independent structure units. It is computationally straightforward to obtain a similarity score. In fact, it is given by the maximum of the similarity scoring function $S^3_{f,g}(t,r)$ when Equation 5 is optimized. In Table I we



(a) Segmented trimers (b) Old segmentation (c) New segmentation (d) Trimers

Fig. 4. Illustrations of segmentation and averaging on rice dwarf virus (RDV). (a) The five independent trimers arranged on the virus capsid (viewed from outside). (b) The segmentation results without structure unit alignment (viewed from inside). (c) The segmentation results with structure unit alignment showing better consistency between structure units. (d) The averaged trimer (in green) looks less noisy and more symmetric than the original trimer (in golden).



(a) Density map (b) Old segmentation (c) New segmentation (d) Averaged map

Fig. 5. Illustrations of segmentation and averaging on ϕ29. (a) The reconstructed cryo-EM density map of ϕ29. The symmetry axes are detected using our automatic method [16] and a total of ten independent structure units are labeled. (b) The structure units are segmented based on our previous segmentation method, without 3D alignment between structure units. One can clearly see some inconsistency between both pentons and hexons. For example, the structure units in blue are much larger than those in green although both are pentons. (c) The structure units are segmented based on our previous segmentation method, combined with the 3D alignment approach as presented in Section III-B. (d) The segmented pentons and hexons can be averaged and the whole structure can be reconstructed from the averaged structure units based on the 3D alignment matrices.

give the similarity scores between the five independent trimers as shown and indexed in Fig. 4(a). The scores in bold on the diagonal indicate how symmetric each individual trimer is, and they are calculated by Equation 3 where T is the rotation along the related symmetry axis by an amount of $\frac{2\pi}{3}$ (in general, $\frac{2\pi}{n}$, where n is the folding number). Table II shows the similarity scores between the four 5-fold structure units of ϕ29, where the indexing numbers are given in Fig. 5(a). The similarity scores between the six hexons are shown in Table III. The subunit #5 seems less similar to the others because it is close to and very likely “disturbed” by the tail.

TABLE I
SIMILARITY SCORES BETWEEN THE FIVE TRIMERS OF RDV

	S_1	S_2	S_3	S_4	S_5
S_1	**0.955**	0.911	0.848	0.900	0.894
S_2		**0.926**	0.854	0.889	0.880
S_3			**0.872**	0.848	0.845
S_4				**0.934**	0.885
S_5					**0.856**

TABLE II
SIMILARITY SCORES BETWEEN THE FOUR PENTONS OF ϕ29

	S_0	S_1	S_2
S_0	**0.991**	0.948	0.949
S_1		**0.960**	0.958
S_2			**0.961**

3) Averaging Subunits: If the segmented structure units of a large bio-molecular complex (such as viruses) have high similarities, they can be averaged such that the structural analysis of the entire complex can be simplified to the analysis of a single averaged structure unit. Averaging two structure units is done by aligning one of them to the other and taking the average density values of both. In general, the averaged map has a higher signal-to-noise ratio than each individual structure unit, which makes it easier to further analyze the structures (e.g., secondary structure identification, as will be discussed). Fig. 4(d) shows one segmented trimer of RDV (in golden) and the averaged map of the five independent trimers (in green). We can see that the averaged trimer has

TABLE III

SIMILARITY SCORES BETWEEN THE SIX HEXONS OF $\phi 29$

	S_4	S_5	S_6	S_7	S_8	S_9
S_4	**0.973**	0.793	0.947	0.940	0.869	0.880
S_5		**0.741**	0.793	0.785	0.772	0.787
S_6			**0.971**	0.963	0.884	0.884
S_7				**0.965**	0.888	0.884
S_8					**0.829**	0.938
S_9						**0.842**

TABLE IV

INDEXING NUMBERS FOR PAIRS OF HEXONS OF $\phi 29$

	S_4	S_5	S_6	S_7	S_8	S_9
S_4		1	2	3	4	5
S_5			6	7	8	9
S_6				10	11	12
S_7					13	14
S_8						15
S_9						

higher 3-fold symmetry and better signal-to-noise ratio than the original trimer. Fig. 5(d) shows the map constructed from the averaged penton (orange), the averaged hexon (green), and the tail (blue).

4) Simplifying Pseudo-atomic Modeling: As shown in Fig. 1(a), the goal of structural elucidation of large bio-molecular complexes is to build a pseudo-atomic model for a given map. As viruses often contain a number of independent subunits which are then duplicated to construct the entire capsid, it is efficient to build the model for the entire map by modeling the independent subunits and then transforming the model to the other (dependent) structure units. The transformation matrices are exactly the outputs of our alignment algorithm. As we mentioned in Section II, modeling individual subunits can be conducted in two ways: structural fitting for moderate resolution maps ($10\AA - 20\AA$) and secondary structure analysis for intermediate (sub-nanometer) resolution maps ($5\AA - 10\AA$). We shall give details on secondary structure detections in Section IV. Structural fitting is a process in which a PDB structure is fitted into the cryo-EM density map such that their cross-correlation is maximized. There have been a number of software packages which provide automatic structural fitting and/or interfaces for manual refinements. A recent review on the computational approaches can be found in [30]. In the experiments shown here, we use a software tool called *Situs* [38] to fit the PDB structure of $\phi 29$ monomer protein into the monomers of the chosen penton and hexon, as shown in Fig. 6(a) in magenta and cyan respectively. The PDB structures fitted are then automatically transformed to the other subunits using the matrices obtained from the 3D alignment algorithm. Fig. 6(b) shows a close view of the models for the chosen penton and hexon. The pseudo-atomic model in the top region of $\phi 29$ is shown in Fig. 6(c).

D. Discussions

As we described earlier, the critical points play a crucial role in our alignment approach. To demonstrate the dependency of alignment accuracy on the number of chosen critical points, we show some experiments in Fig. 7 where all six independent hexons of $\phi 29$ are aligned against each other according to different numbers of critical points. There are a total of 15 pairs and they are indexed from 1 to 15 as given in Table IV (refer to Fig. 5 for the indexes of subunits $S_i, i = 4, \cdots, 9$). The alignment between each pair of the six hexons is represented by one translation and one rotation (t and r in Fig. 3).

We first use all the critical points (2901 in total) in the entire capsid map (171^3 voxels in total) to align the hexons and the alignment results are treated as the "reference". We then choose a number of subsets of the critical points by restricting ourselves to those whose intensity values are greater than certain thresholds. The thresholds we consider here include 120, 140, 160, 180, 200, 210, 215 and 220 (remember that the original map is scaled to the range from 0 to 255), and the corresponding numbers of qualified critical points are 2177, 1336, 852, 719, 518, 277, 125, and 18, respectively. These thresholds are chosen from a range that we typically used for most experiments we did. The 3D alignments between the hexons are calculated based on the subsets of critical points and then compared with the "reference". Fig. 7(a) shows the translation errors in pixel ($\sim 5.53\AA$/pixel for this map). We can clearly see that, as long as we use more than ~ 500 critical points (for the entire capsid map!), the translation errors are bounded by ~ 0.5 pixel (or $\sim 3\AA$). For a map with a resolution of $\sim 15\AA$, this upper bound of errors is quite encouraging. From Fig. 7(a) we can also see that most of the errors corresponding to 277 and 125 critical points are also bounded by ~ 0.5 pixel. This experiment shows that our alignment method is very robust to the number of critical points used to calculate the similarity scores (see Equation 5). Compared to the translation, the rotation errors are more sensitive to the number of critical points, as shown in Fig. 7(b). When the number of critical points considered in the map is greater than 200 (i.e., the top six cases in the legend of Fig. 7(b)), the rotation errors can be roughly bounded by 0.05 *Radian* except a few pairs (indexed as $1, 6, 7, 8, 9$). Interestingly all these large errors are related to subunit S_5 (see Table IV) and seem to be caused by the significant dissimilarities between S_5 and other hexons (see Table III too). Since the radius of the hexons is roughly $80\AA$, the upper bound, 0.05 *Radian*, gives the maximal error of $80 \times 0.05 = 4\AA$ (i.e., the arc length) and the average error of $\sim 2\AA$. This is comparable to the translation errors and is also acceptable given that the resolution of the map is about $15\AA$.

Another main concern about our alignment algorithm is the speed of the real space method, compared to the FFT approach. We did not implement the FFT-based algorithm but we give below the complexity analysis of both approaches. In general cases where one has to search in three translations and three rotations, FFT performs much better than the real space method as their time complexities are $O(P \times Q log_2 Q)$ and $O(P \times Q^2)$ respectively, where Q is the total number of voxels in the volume and P is the number of samples in the rotational search space. In our particular case, however, we have only one degree of freedom for translation and the other for rotation. Let P_1 and K be the number of samples in the new rotational and translational search spaces respectively, and

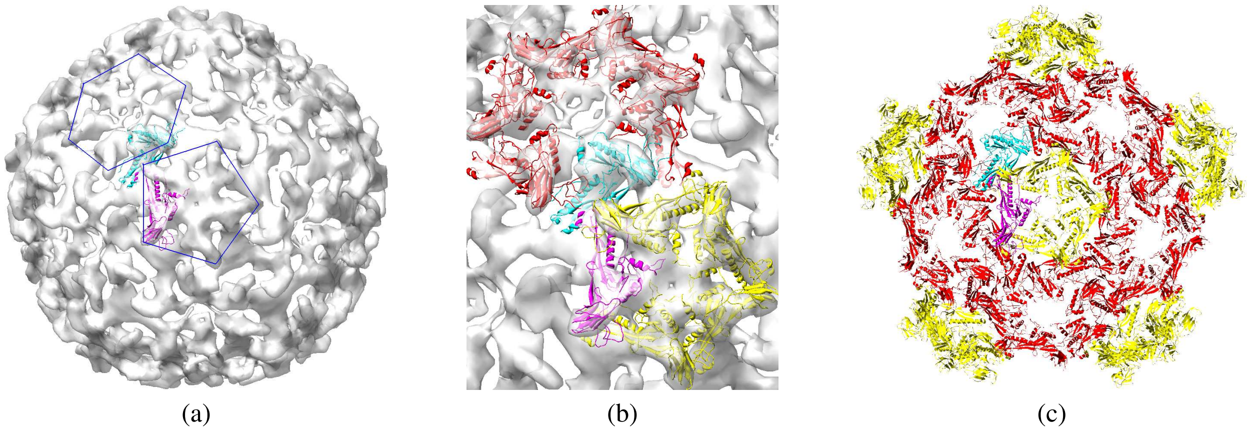


Fig. 6. Illustrations of pseudo-atomic modeling of $\phi29$. (a) Two PDB structures were fitted into the monomers of the chosen penton and hexon, using the software package called *Situs* [38]. Shown here is the view from the top (head) of $\phi29$. (b) We can transform the fitted PDB structures to the other monomers of the chosen penton and hexon according to the alignment matrices between the subunits. A closer view of the fitted penton and hexon is shown here. (c) We can also transform the PDB structure in (b) to the whole capsid and get a pseudo-atomic structure of $\phi29$. Shown here is only the top region of $\phi29$.

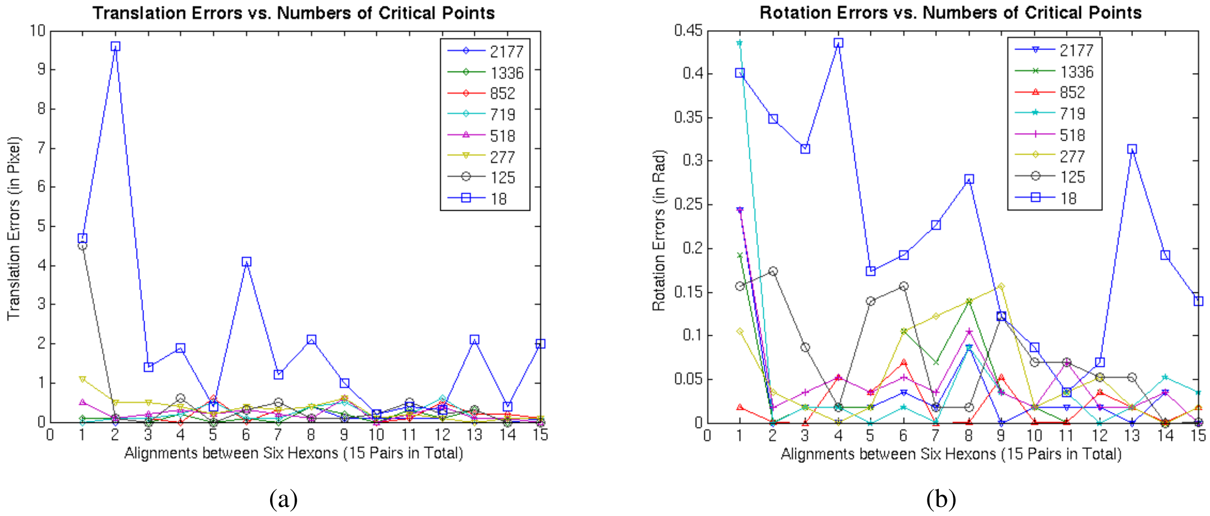


Fig. 7. Illustration of 3D alignment between hexons of *phi*29 with different numbers of critical points. (a) The translation errors with respect to the number of critical points. (b) The rotation errors with respect to the number of critical points.

the time complexities for both FFT and real space methods become $O(P_1 \times Qlog_2Q)$ and $O(P_1 \times Q \times K)$ respectively. From the alignment algorithm presented in Section III-B, we know that $K = 40$. Furthermore, our algorithm utilizes the critical points to calculate the similarity scores, which further reduces the time complexity to $O(P_1 \times Q_C \times K)$, where Q_C is the number of critical points in the volume. Q_C is usually much smaller than Q (in a factor of 100). For example, the size of the individual subunits of $\phi29$ is roughly 32^3 voxels (or $Q = 32,768$). On the other hand, Fig. 7 tells us that when ~ 1000 critical points are used, we can have reasonably good alignment results in both translation and rotation. Since there are 42 subunits in the entire map, each subunit contains an average of ~ 25 critical points. To align two subunits, we need critical points of both subunits, meaning that $Q_C \simeq 50$, which is significantly smaller than Q. As an example, it takes about one minute to align all three pentons and six hexons of $\phi29$ map on a Linux machine (processor: AMD Opteron 246, 2.0 GHz).

IV. Secondary Structure Identification

Proteins typically contain two types of secondary structure elements: α-helices and β-sheets. There have recently been a few papers published on secondary structure identification. An approach for detecting α-helices has been described in [39], where the α-helix is modeled as a cylinder (length and thickness) with a Gaussian distribution density function for each cylinder. The cylinders are then cross-correlated with the segmented protein map, in an exhaustive search and scoring procedure. The exhaustive search occurs in both translation space (three degrees of freedom) and orientation space (two degrees of freedom), and necessarily is computationally expensive. A related approach, designed for β-sheet detection, was recently presented in [40]. This method uses a planar disk model for β-sheets, and performs exhaustive cross-correlation search with the protein density function. This method [40] also inherits a similar disadvantage of high computational time complexity as of [39], due to the similar exhaustive search, and cross-correlation scoring, in five dimensional translation and orientation space.

In the following sub-sections, we present a fast skeleton-based approach for protein secondary structure identification from 3D maps. We capture the location of α-helices and β-sheets by their skeletons (curves and surfaces, respectively). Our skeletonization method is based on a prior seed selection from the 3D density map, and a tracing procedure guided by the eigenvectors of the local 3x3 structure tensor at selected seed points of the protein density map. With the help of eigen-analysis of local structure 3x3 matrices, we can extract the entire "linear" α-helical and "planar" β-sheet skeletons without exhaustively searching the relative five dimensional translation and rotation space of a geometric template and the 3D protein density map. Compared to the above prior methods, our approach is extremely fast (up to hundreds of times faster for typical protein 3D maps) and additionally yields high accuracy identification of secondary structure elements.

A. Local Structure Tensor

Local structure tensor has been used in image processing for solving a number of problems such as anisotropic filtering [41], [32] and motion detection [42]. The idea of the local structure tensor is derived from the well-known principal component analysis (PCA) [43]. It basically captures the principal orientations of a set of vectors in space. We first introduce the gradient tensor, defined on a single vector. Given a 3D map $f(x,y,z)$, the gradient tensor is defined as:

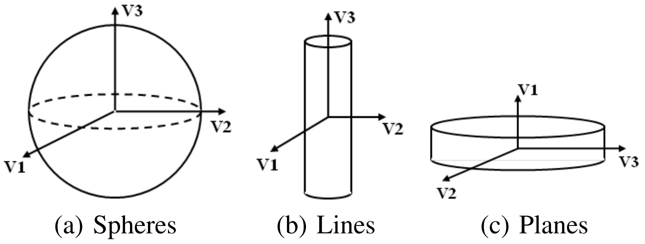
$$G = \begin{pmatrix} f_x^2 & f_xf_y & f_xf_z \\ f_xf_y & f_y^2 & f_yf_z \\ f_xf_z & f_yf_z & f_z^2 \end{pmatrix} \tag{6}$$

This matrix has only one non-zero eigenvalue: $f_x^2 + f_y^2 + f_z^2$. The corresponding eigenvector of this eigenvalue is exactly the gradient (f_x, f_y, f_z). Therefore, this matrix alone does not give more information than the gradient vector. To make the gradient tensor useful, a spatial average (over the image domain) is computed for each of the entries of the gradient tensor, yielding what is called the *local structure tensor*. The averaging is usually based on a Gaussian filter:

$$T_\alpha = \begin{pmatrix} f_x^2 * g_\alpha & f_xf_y * g_\alpha & f_xf_z * g_\alpha \\ f_xf_y * g_\alpha & f_y^2 * g_\alpha & f_yf_z * g_\alpha \\ f_xf_z * g_\alpha & f_yf_z * g_\alpha & f_z^2 * g_\alpha \end{pmatrix} \tag{7}$$

Here g_α is a Gaussian function with standard deviation α. The eigenvalues and eigenvectors of the structure tensor T_α indicate the overall distribution of the gradient vectors within a local 3D window. Three typical structures can be characterized based on the eigenvalues of this structure tensor [41]. Let the eigenvalues be $\lambda_1, \lambda_2, \lambda_3$ and $\lambda_1 \geq \lambda_2 \geq \lambda_3$. Then we have the following cases (see Fig. 8):

- Spheres: $\lambda_1 \approx \lambda_2 \approx \lambda_3 > 0$.
- Lines: $\lambda_1 \approx \lambda_2 >> \lambda_3 \approx 0$.
- Planes: $\lambda_1 >> \lambda_2 \approx \lambda_3 \approx 0$.



(a) Spheres (b) Lines (c) Planes

Fig. 8. Three typical cases for a local structure tensor.

B. Skeletonization

There are a number of prior skeletonization approaches that have been published, including boundary-based methods [44], [45] and boundary-free methods [46], [47]. While a pre-segmentation is required in boundary-based methods, boundary-free methods do not have this requirement and so the skeletons are in general extracted in two steps: volume thinning and skeleton-tracing. In the following, we present our method to trace the skeletons, based on a selection and classification of skeleton-seeds, and guided by eigenvectors of local structure tensors of the 3D density map.

1) Selecting and Classifying Seeds: As discussed in Section IV-A, the local structure tensor can be used to distinguish between spherical, linear, and planar structures. The spherical case is usually of little interest in skeletonization. Therefore, we focus on the other two cases in the following. As we shall see in Section IV-C, these two types of local structures exactly correspond to the α-helices and β-sheets of a protein density map.

Seeds are the starting points for tracing the skeletons. There are multiple ways to define/compute the seed points, with the common condition that all the seed points must be on the skeleton. As we shall see, the graphs of the 3D protein density maps are similar to 3D mountain ranges – with high densities (features) corresponding to the peaks and ridges, with the skeleton extraction akin to ridge tracing. For this particular reason, we select the local maximal (critical) points of the 3D density maps as the seed points of our skeleton-tracers. The stable computation of these critical (seed) points in existence of noise has been detailed in our previous work [16]. Note that by definition the critical points used here are the same as those seen in our segmentation/alignment algorithms. However, since the structural unit used for secondary structure identifications is usually the averaged one of the segmented subunits, the actual critical points in the structural unit here are usually different from those in the original cryo-EM density map due to their different signal-to-noise ratios.

Since we are dealing with two types of skeletons: linear and planar, for which the skeleton-tracings are different, we need to classify each maximal critical point into either ”linear” or ”planar” seed before tracing the skeletons. In [41], the authors used the three real eigenvalues of the symmetric structure tensor to distinguish lines from the planes. However, this criterion does not work well for protein secondary structures because some parts of proteins (e.g., coils, turns) look locally like helices except that they are thinner. Therefore, a better way to classify them is to use the thickness of the secondary structure elements along three principal axes. The thickness along any direction is defined by the width of the region above a pre-chosen threshold in that direction. Since we know the typical thickness of a helix, the threshold values can actually be determined automatically from the seed points that correspond to the helices based on the initial linear classification using the eigenvalue criterion in [41]. Once we know the thickness information for each seed point, we classify the seeds into ”linear” and ”planar” according to a range criterion. Let t_1, t_2, t_3 be the thicknesses corresponding to the eigenvectors as shown in Fig. 8. The following ranges are then used to classify the seeds:

- lines: $\frac{t_1+t_2}{2} > helix_{min}$ and $\frac{t_1+t_2}{2} < helix_{max}$ and $min(\frac{t_1}{t_2}, \frac{t_2}{t_1}) > max(\frac{t_1}{t_3}, \frac{t_2}{t_3})$,
- planes: $t_1 > sheet_{min}$ and $t_1 < sheet_{max}$ and $min(\frac{t_2}{t_3}, \frac{t_3}{t_2}) > max(\frac{t_1}{t_2}, \frac{t_1}{t_3})$,

where $helix_{min}$ and $helix_{max}$ are user-defined possible thicknesses ranges for α-helices in the given density map, and similarly $sheet_{min}$ and $sheet_{max}$ are user-defined thicknesses ranges for β-sheets. While we have chosen default ranges in our implementation, each of these parameters are modifiable by the biological users.

2) Tracing Skeletons: As mentioned earlier, the α-helices and β-sheets are respectively captured by curves and surfaces in our method. The seeds discussed above provide good starting points for the individual skeletons. To avoid exhaustive search, we extract the skeletons by tracing the eigenvectors of the local structure tensors. The line-tracer (used for α-helices) is one-dimensional and hence is much easier than the plane-tracer (used for β-sheets). To trace a line structure, we start from the seed in two opposite directions, and follow the principal axis, defined by the eigenvector corresponding to the minimum eigenvalue of the local structure tensor (see Fig. 9(a)). To trace a planar structure, we utilize the popular isocontouring technique [48]. We start from the seed point and compute the plane that is perpendicular to the eigenvector corresponding to the maximal eigenvalue (v_1 in Fig. 8(c)). The plane partitions all eight vertices of the cell containing the seed point into two classes: positive and negative. In addition, the plane intersects with some of the twelve edges of the cell. Both the classification of the vertices and the intersection information with the edges can be used in the isocontouring lookup table, yielding a polygon (a list of triangles) representing the skeletons within the current cell (see Fig. 9(b)). Next, we move to the neighboring cells with which the detected skeleton (polygon) intersects. For the example in Fig. 9(b), we need to check four neighboring cells (back, front, right, and bottom). For each of those new cells, the “checking” point (similar to the seeds) is calculated as the center of the existing intersecting points between the already-detected skeletons and the new cell. The new polygon within the new cell is extracted using the idea as explained above, based on the “checked” point and the new structure tensor around it. This process is repeated until a certain stopping criterion is reached. The plane-tracer outputs a triangular mesh of skeletons. The stopping criteria for both line-tracer and plane-tracer are similar – the tracing process terminates whenever no new cells satisfy the criteria as discussed in Section IV-B.1.

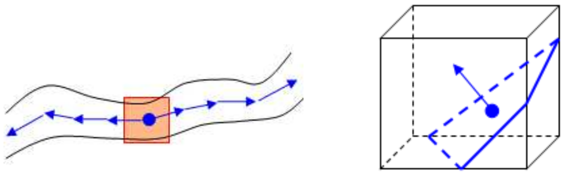

(a) Line-tracer (b) Plane-tracer

Fig. 9. Skeleton-tracers. (a) Tracing 1D skeletons. (b) Tracing 2D skeletons (the marching cube method [49]).

3) Merging Skeleton: For each of the seed points, we extract a curve or a surface, corresponding to a helix or sheet respectively. However, quite often we see more than one seed point corresponding to the same secondary structure. One example is illustrated in Fig. 10(a) and a closer view is shown in Fig. 10(b). In order to have only one curve/surface for each helix/sheet, we merge the superfluous skeletons corresponding to the same helix/sheet. Let the skeletons initiated from seeds A and B be denoted by $S(A)$ and $S(B)$ respectively. They are line segments for α-helices and triangulated surfaces for β-sheets. $S(A)$ and $S(B)$ are subject to being merged if the minimal distance between $S(A)$ and $S(B)$ is less than a preset threshold. The threshold is usually set as the thickness of α-helices or β-sheets in the given map that is assumed to be isotropic in X, Y, and Z directions. Besides eliminating the redundancy, the merging process also yields balanced skeletons because the new skeletons are the average of previously redundant ones. Fig. 10(c) shows the skeleton after merging. It is worth pointing out that the topological ambiguity problem seen in the original marching cube method [49] can be resolved by an improved algorithm as discussed in [48].

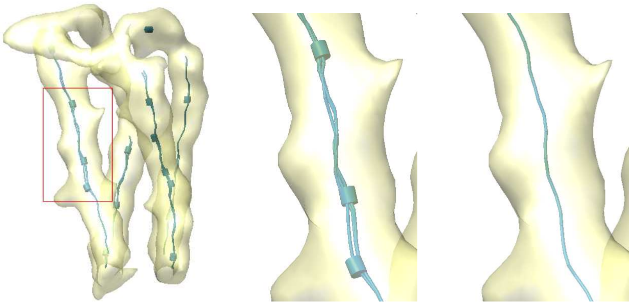

(a) Traced skeleton (b) Closer view (c) Merged skeleton

Fig. 10. Skeleton extraction. (a) The initial skeletons traced by the line-tracer. (b) A zoomed-in view of the rectangular region as shown in (a). The thick “dots” are seed points. (c) The skeletons after merging

C. Applications: Protein Secondary Structure Detection

Once the skeletons of a protein density map are extracted, it is quite straightforward to locate the protein secondary structure elements. It is generally enough to represent β-sheets using the extracted triangular mesh. In case the resolution of the given maps is high enough to distinguish between individual beta-strands, we could apply the line-tracer to extract the strands and get a more refined model of the β-sheets. As for α-helices, we construct a cylinder model for each helix based on the extracted skeletons.

We have tested our skeleton-based protein secondary structure identification approach on a large number of simulated and experimentally-reconstructed protein density 3D maps. We only provide a few typical examples here. Fig. 11 illustrates the performance of our approach on a Gaussian blurred map of two X-ray atomic structures. The first example is cytochrome c' (PDBID = 1bbh). The blurred map at $8\mathring{A}$ and detected skeletons are shown in Fig. 10. From the skeletons, four α-helices are detected as shown in Fig. 11(a). To demonstrate the accuracy of our approach, the skeletons and detected helices are compared with the PDB structure in Fig. 11(b) and (c). Another example is the blurred map of rat CD4 (PDBID = 1cid) as seen in Fig. 11(d). The detected skeletons (sheets) are shown in Fig. 11(e) and compared with PDB structures in Fig. 11(f).

We also tested our approach on two simulated maps containing both α-helices and β-sheets. Fig. 12(a) shows the blurred map of the triose phosphate isomerase from chicken muscle (PDBID = 1tim) at $8\mathring{A}$. The detected α-helices and β-sheets are shown in Fig. 12(b) with density map and in Fig. 12(c) with PDB structure. We can see that most helices/sheets agree very well with the PDB structure except that one small α-helix that is immediately adjacent to a long helix is missed (indicated by a red arrow). This result also agrees with that seen in [39]. Fig. 12(d) and (e) give another view of (b) and (c), respectively. In Fig. 13(a) we show a more complicated simulated map at $8\mathring{A}$, blurred from the PDB structure of the bluetongue virus VP7 (PDBID = 1bvp). This map contains a total of 27 α-helices in the lower domain and 3 α-helices and a few β-sheets in the upper domain. The detected α-helices and β-sheets are shown in Fig. 13(b) with density map and in Fig. 13(c) with PDB structure. All 30 α-helices and major portions of β-sheets are correctly identified and agree very well with the PDB structure. Although two small α-helices (indicated by red arrows) are misidentified due to a couple of turns getting very close to each other, our results show better performance than the method proposed in [39]. Fig. 13(d) shows another view of (c). A closer view of the skeletons together with the PDB structure is shown in Fig. 13(e).

We also demonstrate our approach on a 3D map, reconstructed from experimental cryo-EM images of the rice dwarf virus (RDV) [8]. Fig. 14(a) shows two capsid layers segmented: the outer layer (on the top) and the inner layer (on the bottom) [16], [17]. For better illustrations, only a quarter of the whole layers are seen here. The segmented P3 protein (from inner layer) and P8 protein (from outer layer) are shown in Fig. 14(b). The detected α-helices and β-sheets for both proteins are shown in Fig. 14(c).

D. Discussions

In this subsection, we would like to give some numerical analysis of the performance of our algorithms. A number of maps are used to estimate the detection errors and they are blurred at 8 $\mathring{A}$ from a list of chosen PDB structures including 1BBH, 1BVP, 1C3W, 1CID, 1DXT, 1IRK, 1LGA, and 1TIM. These structures include those of only α-helices (1BBH, 1DXT, and 1LGA), those of only β-sheets (1CID, with a negligibly short helix), and those of both types (the rest).

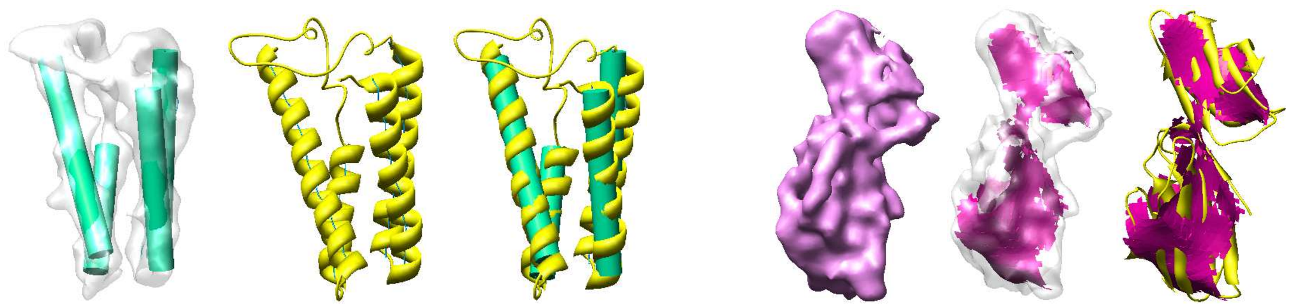
We start with the error estimation of our helix detection algorithm. We first compute the false negative or FN (missed helices) and false positive or FP (wrongly detected helices) for each of the chosen PDB structures. The total number of helices for each structure is shown in the 2nd column of Table V. As a comparison, some of the FNs and FPs by $HelixHunter$ [39] are listed in the 3rd and 4th columns of Table V (based on Figure 2 of [39]). The FNs and FPs by our method are listed in the 5th and 6th columns of Table V.

Excluding the false positives and false negatives, there should be a one-to-one correspondence between the α-helices in the PDB structure, denoted by $\{H_i, i = 1,2,\cdots,n\}$, and the α-helices detected by our method, denoted by $\{\hat{H}_i, i = 1,2,\cdots,n\}$. We can further evaluate the accuracy of our algorithm by comparing each pair of helices taken from $\{H\}$ and $\{\hat{H}\}$ respectively. For simplicity we represent each α-helix by two end points. For α-helices in $\{H\}$ the two end points, denoted by $\{A_i, B_i\}, i = 1,2,\cdots,n$, are calculated by principal component analysis of all the C_α atoms in the i^{th} helix in the PDB structure. The end points in $\{\hat{H}\}$, denoted by $\{\hat{A}_i, \hat{B}_i\}, i = 1,2,\cdots,n$, are readily available as outputs of our algorithm. To estimate the error between each pair of helices in $\{H\}$ and $\{\hat{H}\}$, we define the distance between two vectors as follows:

$$HE(A,B;C,D) = \min\left(\frac{d(A,C)+d(B,D)}{2}, \frac{d(A,D)+d(B,C)}{2}\right), \tag{8}$$

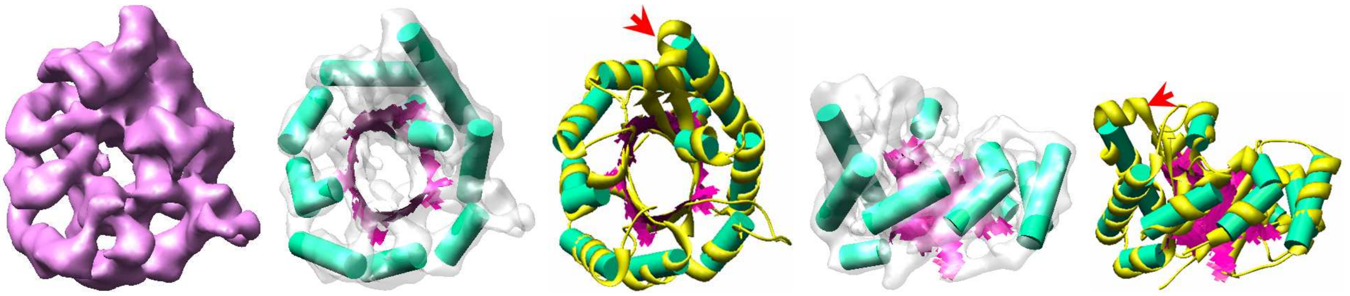
where A, B are the end points of a helix in $\{H\}$ and C, D are the end points of the corresponding helix in $\{\hat{H}\}$. $d(X,Y)$ is the Euclidean distance between points X and Y. Equation 8 gives the error estimation between a helix in PDB structure and the corresponding helix detected by our approach. Fig. 15 shows the error estimations of all helices (except the false negatives and false positives) for the chosen PDB structures. Short helices tend to have larger errors partly because the principal component analysis has significant errors when modeling short helices with cylinders due to few number of C_α atoms. The averaged helix error (denoted by AHE) for each PDB structure is given in the 7th column of Table V.

Compared to α-helices, β-sheets are more difficult to model due to their arbitrary shapes and curvatures. At low to intermediate resolutions it is impossible to distinguish between individual beta strands and sometimes it is even difficult to find the exact boundaries of individual β-sheets. In the worst cases, the extracted median surfaces (skeletons) may contain a few small “holes” near the boundaries of sheets where it is more ambiguous for the local structures to be classified into one of three cases as shown in Fig. 8. Due to the difficulty of characterizing topologically the shapes of β-sheets, the error estimation of β-sheet detection is hence approximated by the (globally) average distance between the C_α atoms and the vertices of the triangular mesh detected by our approach. Let



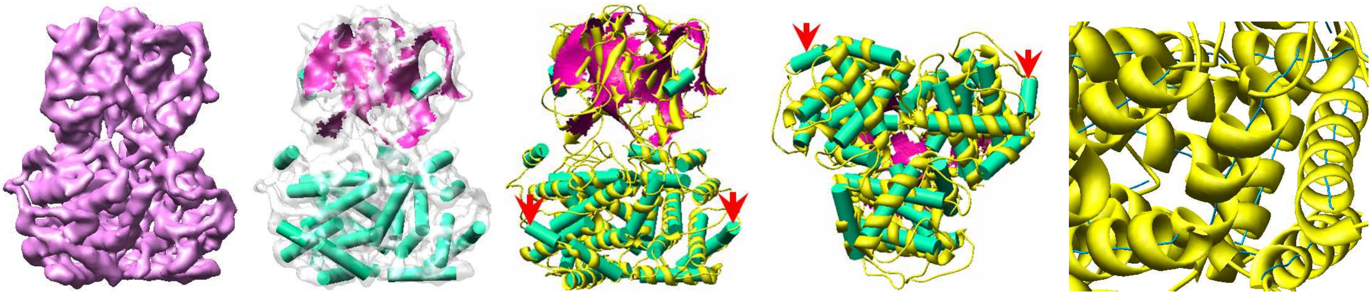
(a) Detected helices (b) Skeletons (c) Helices with PDB (d) Blurred map (e) Skeletons (f) Sheets

Fig. 11. Examples on two blurred maps of x-ray crystal structures. (a) The α-helices detected from the blurred map of cytochrome c' (PDBID = 1bbh). The blurred map at 8$\AA$ and extracted skeletons were shown in Fig. 10. (b) The skeletons detected from the blurred map and compared with the PDB structure. (c) The detected helices are compared with the PDB structures. (d) The blurred map of rat CD4 (PDBID = 1cid) at 8$\AA$. (e) The skeletons (corresponding to β-sheets) detected from the blurred map. (f) The sheets are compared with the PDB structures.



(a) Blurred map (b) Helix-sheet (map) (c) Helix-sheet (PDB) (d) Another view of (b) (e) Another view of (c)

Fig. 12. Secondary structure identification on the triose phosphate isomerase from chicken muscle (PDBID = 1tim). (a) The blurred maps at 8$\AA$ from the x-ray crystal structure. (b) The α-helices (green) and β-sheets (pink) detected using our method. (c) The detected helices/sheets are compared with the PDB structures. We can see that most helices/sheets agree very well with the PDB structure except that one small α-helix that is immediately adjacent to a long helix is missed (indicated by red arrow). This result also agrees with that seen in [39]. (d) Another view of (b). (e) Another view of (c).



(a) Blurred map (b) Helix-sheet (map) (c) Helix-sheet (PDB) (d) Another view of (c) (e) Zoomed-in skeletons

Fig. 13. Secondary structure identification on the bluetongue virus VP7 (PDBID = 1bvp). (a) The blurred maps at 8$\AA$ from the x-ray crystal structure. (b) The α-helices (green) and β-sheets (pink) detected using our method. (c) The detected helices/sheets are compared with the PDB structures. All α-helices and major portions of β-sheets are correctly identified and agree very well with the PDB structure. Although two small α-helices (indicated by red arrows) are misidentified due to a couple of turns running into each other, our results show better performance than the method proposed in [39]. (d) Another view of (c). (e) A closer view of the detected skeletons together with the PDB structure. The chosen region is roughly in the rectangular area right below the center of (c).

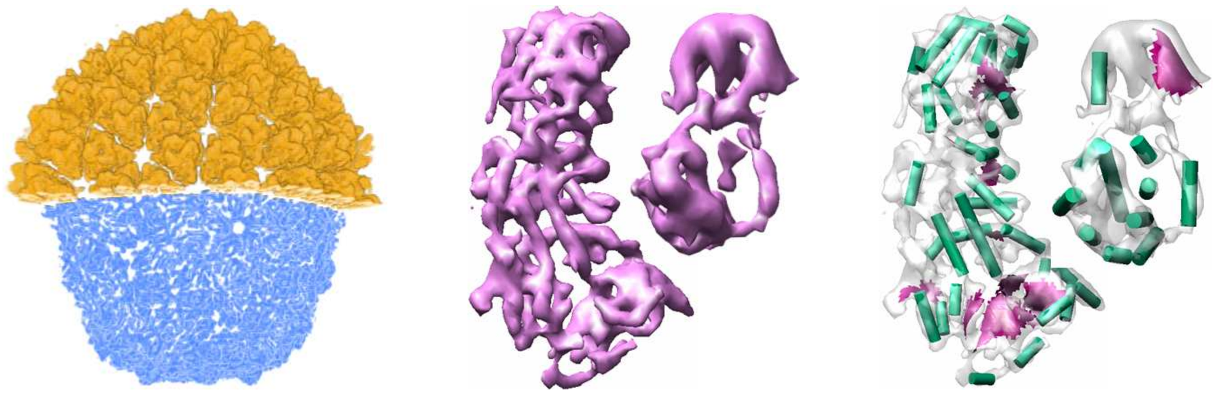
$C_i, i = 1, 2, \cdots, n$ denote the C_α atoms of all β-sheets in one PDB structure and $V_j, j = 1, 2, \cdots, m$ be the vertices of the triangular mesh detected. We calculate two errors as follows:

$$SFN(C,V) = \frac{1}{n}\sum_{i=1}^{n}\min_{j=1}^{m} d(C_i,V_j), \tag{9}$$

$$SFP(C,V) = \frac{1}{m}\sum_{j=1}^{m}\min_{i=1}^{n} d(C_i,V_j), \tag{10}$$

where $d(X,Y)$ is the Euclidean distance between points X and Y. Basically $SFN(C,V)$ indicates the false negatives of the β-sheet detection (or in other words, the β-sheets that appear in the PDB structures but are not detected). In contrast, $SFP(C,V)$ corresponds to the false positives, indicating the amount of β-sheets that are detected but are not truly β-sheets in the PDB structures. These two error estimations for each of the chosen structures are shown in the 8th and 9th columns of Table V, respectively.

Since our secondary structure detection is based on the local structure tensor, it would be worth investigating how the size of the local window (used in Equation 7) affects the accuracy of our secondary structure detection approach.



(a) Segmented layers (b) Segmented proteins (c) Detected helices/sheets

Fig. 14. The example of a real cryo-EM reconstructed map. (a) The outer layer (top) and inner layer (bottom) of the rice dwarf virus (RDV) after segmentations [8], [16]. Shown here is only a quarter of the whole layers. (b) The segmented P3 protein (left) from inner layer and P8 protein (top-right) from outer layer. (c) The α-helices (green) and β-sheets (pink) detected from the P3 and P8 proteins.

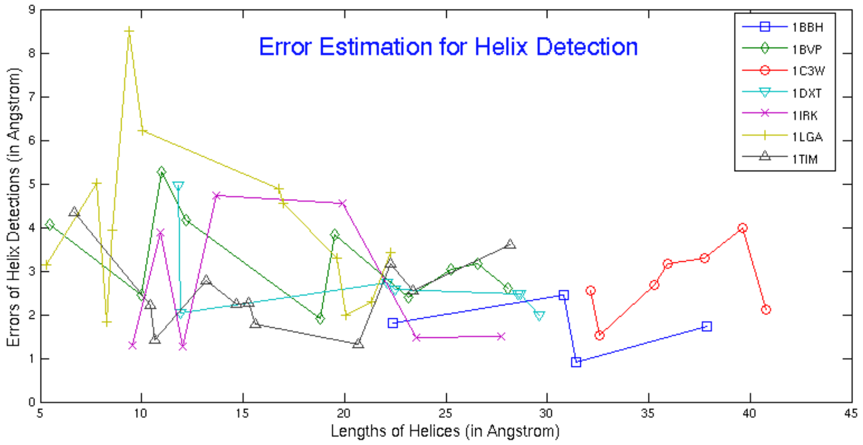


Fig. 15. The illustration of the error estimations of all helices (except the false negatives and false positives) for the chosen PDB structures. Short helices tend to have larger errors partly because the principal component analysis has significant errors when modeling short helices with cylinders due to very few number of C_α atoms.

TABLE V
ERROR ESTIMATIONS OF SECONDARY STRUCTURES DETECTION

PDB	#	HelixHunter		Our Method				
		FN	FP	FN	FP	AHE	SFN	SFP
1*BBH*	4	N/A	N/A	0	0	1.63	N/A	N/A
1*BVP*	30	6	1	0	2	3.29	1.45	2.28
1*C3W*	8	1	0	1	0	2.76	4.08	2.02
1*CID*	N/A	N/A	N/A	N/A	N/A	N/A	1.16	2.40
1*DXT*	8	N/A	N/A	1	0	2.75	N/A	N/A
1*IRK*	9	1	1	2	0	2.67	2.45	2.95
1*LGA*	13	N/A	N/A	2	1	4.09	N/A	N/A
1*TIM*	12	2	0	1	0	2.51	1.51	2.19

#: total number of helices in PDB.
FN and *FP*: false negatives and false positives, respectively.
AHE (in $\AA$): average helix error.
SFN and *SFP* (in $\AA$): sheet errors defined in Eq. 9 and Eq. 10, respectively.

As mentioned before, the local structure tensor is basically a variant of the principal component analysis (PCA), which captures statistically the features of the density map in a local window. Therefore, the window size must be big enough to characterize the local structures from a statistical point of view, but small enough to exclude any neighboring features that could influence the eigenvectors of the local structure being studied. An ideal window size is hence the size of the feature to be extracted. In case of the α-helix, for example, the radius of an α-helix is about 6 $\AA$ in the PDBs but in electron density maps an α-helix looks much "thinner" $-$ the authors in [39] suggest $\sim 2.5\AA$ as the radius of an α-helix in an intermediate resolution map. Therefore, the window size could be chosen from any value from $2.5\AA$ to $6\AA$, although the optimal one may vary slightly from structure/resolution to structure/resolution. As an example, the α-helices of 1BBH blurred map at $8\AA$ resolution are detected based on different window sizes: $2\AA - 10\AA$, and $15\AA$. The average error of the detected α-helices are: $1.59\AA$, $1.63\AA$, $1.40\AA$, $1.33\AA$, $1.65\AA$, $1.75\AA$, $2.02\AA$, $2.72\AA$, $3.01\AA$, and $3.31\AA$, respectively. The optimal size for this map is around $5\AA$. The thickness of β-sheets is usually smaller than the diameter of α-helices, but for simplicity, we use the same window size for both α-helices and β-sheets. In the experiments shown in Fig. 10 $-$ Fig. 15 and Table V, the window size is chosen as $3\AA$, which has a good balance between speed and accuracy.

In addition to the encouraging accuracy, the speed of our approach is another significant advantage compared to the previous methods [39], [40]. Depending on the parameters chosen, the *HelixHunter* [39] may take ten minutes or up to one hour on a Linux machine (processor: AMD Opteron 246, 2.0 GHz), for such maps as P8 or P3 proteins of RDV (128^3 voxels), to detect only the α-helices. Our method takes only 3 seconds for P8 protein and 10 seconds for P3 proteins on the same Linux computer, to detect both α-helices and β-sheets. In fact, the majority of time in our method is consumed in the preprocessing step, namely, the calculations and diffusions of gradient vectors, making our approach sufficiently fast for interactive adjustments of parameters by the users from an interface. We do not have the timings for *SheetMiner* as presented in [40]. However, we believe that the *SheetMiner* of [40] should be as slow as, if not slower than, the *HelixHunter* of [39] because both methods used exhaustive searching schemes in the translational and orientational spaces.

V. Conclusions

In the prior sections, we have presented computational approaches for automatic ultra-structure analysis of reconstructed 3D-EM maps of macromolecules. In particular, we give fast methods to align (match) two segmented structure units in 3D space, and to identify secondary structure elements of a protein density map. The 3D alignment algorithm yields a set of transformation matrices which are essential for a number of 3D map post-processing methods, including similarity quantification, segmentation refinement, similar structure unit averaging/improvement, and pseudo-atomic modeling. Our skeleton-based method for secondary structure identification is extremely fast (usually more than a couple of magnitude times faster) compared to existing methods [39], [40]. While the existing methods could detect either α-helices or β-sheets, our approach simultaneously identifies both types of secondary structures with high accuracy. When combined with our previous segmentation method [16], [17], the approaches presented here can be employed to interpret automatically a wide range of bio-molecular structures especially those reconstructed by single particle cryo-EM.

Acknowledgements

This work was supported in part by NSF grants IIS-0325550, CNS-0540033 and grants from the NIH 0P20 RR020647, R01 GM074258, R01-GM073087. We are grateful to Dr. Wah Chiu and Dr. Matthew Baker at Baylor College of Medicine, for helpful discussions related to this project and for providing us the reconstructed 3D map of the rice dwarf virus. We are also thankful to Dr. Tim Baker and Dr. Jinghua Tang at The University of California at San Diego, for their encouragement for automatic 3D map interpretation and for providing us with the $\phi 29$ cryo-EM density map. The implementation of all of the above structural interpretation algorithms, as well pictures used here, have been made in our inhouse image processing and visualization software tool, called VolRover. The VolRover program is in the public domain and can be freely downloaded from our center's software website (http://www.ices.utexas.edu/CVC/software/).

References

[1] M. Woolfson, *An Introduction to X-ray Crystallography*. Cambridge University Press, January 1997.

[2] "Online course on protein crystallography," *http://www-structmed.cimr.cam.ac.uk/course.html*.

[3] A. E. Ferentz and G. Wagner, "NMR spectroscopy : a multifaceted approach to macromolecular structure," *Quarterly Reviews of Biophysics*, vol. 33, pp. 29–65, 2000.

[4] P. Guntert, "Structure calculation of biological macromolecules from NMR data," *Quarterly Reviews of Biophysics*, vol. 31, pp. 145–237, 1998.

[5] H. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. Bhat, H. Weissig, I. Shindyalov, and P. Bourne, "The protein data bank," *Nucleic Acids Research*, vol. 28, pp. 235–242, 2000.

[6] T. S. Baker, N. H. Olson, and S. D. Fuller, "Adding the third dimension to virus life cycles: three-dimensional reconstruction of icosahedral viruses from cryo-electron micrographs," *Microbiology and Molecular Biology Reviews*, vol. 63, no. 4, pp. 862–922, 1999.

[7] M. van Heel, B. Gowen, R. Matadeen, E. Orlova, R. Finn, T. Pape, D. Cohen, H. Stark, R. Schmidt, M. Schatz, and A. Patwardhan, "Single-particle electron cryo-microscopy: towards atomic resolution," *Quarterly Reviews Biophysics*, vol. 33, no. 4, pp. 307–369, 2000.

[8] Z. H. Zhou, M. L. Baker, W. Jiang, M. Dougherty, J. Jakana, G. Dong, G. Lu, and W. Chiu, "Electron cryomicroscopy and bioinformatics suggest protein fold models for rice dwarf virus," *Nature Structural Biology*, vol. 8, no. 10, pp. 868–873, 2001.

[9] W. Jiang, Z. Li, M. L. Baker, P. E. Prevelige, and W. Chiu, "Coat protein fold and maturation transition of bacteriophage P22 seen at subnanometer resolution," *Nature Structural Biology*, vol. 10, no. 2, pp. 131–135, 2003.

[10] Z. Zhou, M. Dougherty, J. Jakana, J. He, F. Rixon, and W. Chiu, "Seeing the herpesvirus capsid at 8.5 angstrom," *Science*, vol. 288, pp. 877–880, 2000.

[11] R. Matadeen, A. Patwardhan, B. G. E. Orlova, T. Pape, M. Cuff, F. Mueller, and R. B. M. van Heel, "The e. coli large ribosomal subunit at 7.5 angstrom resolution," *Structure*, vol. 7, pp. 1575–1583, 1999.

[12] *IEEE Transactions on Image Processing: Special Issue on Molecular and Cellular Bioimaging (Guest Editors: R.F. Murphy, E. Meijering and G. Danuser)*, vol. 14, no. 9, 2005.

[13] S. Ludtke, P. Baldwin, and W. Chiu, "EMAN: semiautomated software for high-resolution single-particle reconstructions," *Journal of Structural Biology*, vol. 128, pp. 82–97, 1999.

[14] J. Frank, M. Radermacher, P. Penczek, J. Zhu, Y. Li, M. Ladjadj, and A. Leith, "SPIDER and WEB: processing and visualization of images in 3D electron microscopy and related fields," *Journal of Structural Biology*, vol. 116, pp. 190–199, 1996.

[15] M. V. Heel, G. Harauz, E. Orlova, R. Schmidt, and M. Schatz, "A new generation of the IMAGIC image processing system," *Journal of Structural Biology*, vol. 116, pp. 17–24, 1996.

[16] Z. Yu and C. Bajaj, "Automatic ultra-structure segmentation of reconstructed cryo-em maps of icosahedral viruses," *IEEE Transactions on Image Processing*, vol. 14, no. 9, pp. 1324–1337, 2005.

[17] M. Baker, Z. Yu, W. Chiu, and C. Bajaj, "Automated segmentation of molecular subunits in electron cryomicroscopy density maps," *Journal of Structural Biology*, vol. 156, no. 3, pp. 432–441, 2006.

[18] W. Chiu, "What does electron cryomicroscopy provide that x-ray crystallography and nmr spectroscopy cannot?" *Ann. Rev. Biophys. Biomol. Struct.*, vol. 22, pp. 233–255, 1993.

[19] L. Amos, R.Henderson, and P.N.Unwin, "Three-dimensional structure determination by electron microscopy of two-dimensional crystals," *Prog. Biophys. Mol. Biol*, vol. 39, pp. 183–231, 1982.

[20] B. McEwen and M. Marko, "The emergence of electron tomography as an important tool for investigating cellular ultrastructure," *The Journal of Histochemistry & Cytochemistry*, vol. 49, no. 5, pp. 553–563, 2001.

[21] A. Koster, R. Grimm, D. Typke, R. Hegerl, A. Stoschek, J. Walz, and W. Baumeister, "Perspectives of molecular and cellular electron tomography," *J. of Structural Biology*, vol. 120, pp. 276–308, 1997.

[22] P. Penczek, M. Marko, K. Buttle, and J. Frank, "Double-tilt electron tomography," *Ultramicroscopy*, vol. 60, pp. 393–410, 1995.

[23] J. Frank, *Three-Dimensional Electron Microscope of Macromolecular Assemblies*. San Diego: Academic Press, 1996.

[24] S. Ludtke, D. Chen, J. Song, D. Chuang, and W. Chiu, "Seeing groel at 6-angstrom resolution by single particle electron cryomicroscopy," *Structure*, vol. 12, pp. 1129–1136, 2004.

[25] A. Roseman, "Particle finding in electron micrographs using a fast local correlation algorithm," *Ultramicroscopy*, vol. 94, pp. 225–236, 2003.

[26] Z. Yu and C. Bajaj, "Detecting circular and rectangular particles based on geometric feature detection in electron micrographs," *Journal of Structural Biology*, vol. 145, no. 1-2, pp. 168–180, 2004.

[27] W. Nicholson and R. Glaeser, "Review: automatic particle detection in electron microscopy," *Journal of Structural Biology*, vol. 133, no. 2-3, pp. 90–101, 2001.

[28] F. Sigworth, "A maximum-likelihood approach to single-particle image refinement," *Journal of Structural Biology*, vol. 122, pp. 328–339, 1998.

[29] M. Sjors, M. Valle, R. Nuez, C. Sorzano, R. Marabini, G. Herman, and J. Carazo, "Maximun likelihood multi-reference refinement for eletron microscopy images," *J. Molecular Biology*, vol. 348, pp. 139–149, 2005.

[30] W. Wriggers and P. Chacon, "Modeling tricks and fitting techniques for multiresolution structures," *Structure*, vol. 9, pp. 779–788, 2001.

[31] J.Fernandez, J.R.Sanjurjo, and J.Carazo, "A spectral estimation approach to contrast transfer function detection in electron microscopy," *Ultramicroscopy*, vol. 68, pp. 267–295, 1997.

[32] J. Weickert, *Anisotropic Diffusion In Image Processing*. ECMI Series, Teubner, Stuttgart, ISBN 3-519-02606-6, 1998.

[33] Z. Yu and C. Bajaj, "A fast and adaptive algorithm for image contrast enhancement," in *Proceedings of 2004 IEEE International Conference on Image Processing*, 2004, pp. 1001–1004.

[34] D. VanArsdale, "Homogeneous transformation matrices for computer graphics," *Computers and Graphics*, vol. 18, no. 2, pp. 177–191, 1994.

[35] M. Morais, K. Choi, J. Koti, P. Chipman, D. Anderson, and M. Rossmann, "Conservation of the capsid structure in tailed dsdna phage the pseudoatomic structure of phi29," *Mol. Cell*, vol. 18, pp. 149–159, 2005.

[36] Y. Tao, N. Olson, W. Xu, D. Anderson, M. Rossmann, and T. Baker, "Assembly of a tailed bacterial virus and its genome release studied in three dimensions," *Cell*, vol. 95, pp. 431–437, 1998.

[37] E. Sifakis and G. Tziritas, "Moving object localization using a multi-label fast marching algorithm," *Signal Processing: Image Communication*, vol. 16, no. 10, pp. 963–976, 2001.

[38] W. Wriggers, R. Milligan, and J. McCammon, "Situs: A package for docking crystal structures into low-resolution maps from electron microscopy," *Journal of Structural Biology*, vol. 125, pp. 185–195, 1999.

[39] W. Jiang, M. Baker, S. Ludtke, and W. Chiu, "Bridging the information gap: computational tools for intermediate resolution structure interpretation," *Journal of Molecular Biology*, vol. 308, pp. 1033–1044, 2001.

[40] Y. Kong and J. Ma, "A structural-informatics approach for mining b-sheets: locating sheets in intermediate-resolution density maps," *Journal of Molecular Biology*, vol. 332, pp. 399–413, 2003.

[41] J.-J. Fernandez and S. Li, "An improved algorithm for anisotropic nonlinear diffusion for denoising cryo-tomograms," *Journal of Structural Biology*, vol. 144, pp. 152–161, 2003.

[42] G. Khne, J. Weickert, O. Schuster, and S. Richter, "A tensor-driven active contour model for moving object segmentation," in *Proceedings of IEEE International Conference on Image Processing*, 2001, pp. 73–76.

[43] I. Jollife, *Principal Component Analysis*. Springer-Verlag (NY), 1986.

[44] L. Lam, S. Lee, and C. Suen, "Thinning methodologies - a compresensive survey," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14(9), pp. 869–885, 1992.

[45] R. Ogniewicz and O. Kubler, "Hierarchic voronoi skeletons," *Pattern Recognition*, vol. 28, no. 3, pp. 343–359, 1995.

[46] J. Jang and K. Hong, "A pseudo-distance map for the segmentation-free skeletonization of gray-scale images," in *Proc. Int'l Conf. Computer Vision*, 2001, pp. 18–23.

[47] Z. Yu and C. Bajaj, "A segmentation-free approach for skeletonization of gray-scale images via anisotropic vector diffusion," in *Proc. Int'l Conf. Computer Vision and Pattern Recognition*, 2004, pp. 415–420.

[48] A. Lopes and K. Brodlie, "Improving the robustness and accuracy of the marching cube algorithm for isosurfacing," *IEEE Trans. Visualization and Computer Graphics*, vol. 9, no. 1, pp. 16–29, 2003.

[49] W. Lorensen and H. E. Cline, "Marching cubes: a high resolution 3D surface construction algorithm," *Computer Graphics*, vol. 21, no. 4, pp. 163–169, 1987.

Zeyun Yu (S'03) received the B.S. degree in Mathematics from Peking University, Beijing, China, in 1996 and the M.S. degree in Pattern Recognition and Machine Intelligence from Chinese Academy of Sciences, Beijing, China, in 1999. He received his Ph.D. degree in Computer Science from The University of Texas at Austin in 2006 and is currently a postdoctoral scholar in the Department of Mathematics at The University of California at San Diego.

His research interests include image processing, pattern recognition, geometric/physical modelling for visualization, computational biology and bioinformatics, and bio-molecular finite element simulations. He is a student member of IEEE and ACM. He is also a member of IEEE Engineering in Medicine and Biology Society.

Chandrajit Bajaj (M'84) graduated from the Indian Institute of Technology, Delhi with a Bachelor's Degree in Electrical Engineering, in 1980 and received his M.S. and Ph.D. degrees in Computer Sciences from Cornell University, in 1983, and 1984 respectively.

Bajaj is currently the Computational Applied Mathematics Chair in Visualization Professor of computer sciences at the University of Texas at Austin, as well as the director of the Center for Computational Visualization, in the Institute for Computational and Engineering Sciences (ICES). His research areas of interest include Image Processing, Computational Geometry, Geometric Modeling, Computer Graphics, Visualization, and Computational Mathematics. Current research topics include the design and development of efficient and robust 2D/3D/4D image and geometry filtering, reconstruction, compression, matching and meshing algorithms. He is applying these algorithms to (a) the structure elucidation and construction of multi-scale domain models of molecules, organelles, cells, tissues and organs from multi-modal, microscopy and bio-medical imaging and (b) a fast high-dimensional search and scoring procedure for identifying energetically favourable binding of proteins with bio-compounds (virtual screening for anti-viral drugs). Bajaj is also involved in developing integrated approaches to computational modeling, mathematical analysis and interrogative visualization, especially for dynamic bio-medical phenomena. He has over 190 publications, has written one book and edited three other books in his area of expertise. He is on the editorial boards for the International Journal of Computational Geometry and Applications, the ACM Transactions on Graphics, and ACM Computing Surveys. He is on numerous national and international conference committees and has served as a scientific consultant to national labs and industry.