Towards Protein Interaction Analysis through Surface Labeling

Virginio Cantoni[1,2], Riccardo Gatti[2], and Luca Lombardi[2]

[1] IEF Institut d'Électronique Fondamentale, Université Paris-Sud XI
[2] University of Pavia, Dept. of Computer Engineering and System Science
{virginio.cantoni,riccardo.gatti,luca.lombardi}@unipv.it

Abstract. The knowledge of the biological function of proteins would have great impact on the identification of novel drug targets, and on finding the molecular causes of diseases. Unfortunately, the experimental determination of protein function is a very expensive and time consuming process. As a consequence, the development of computational techniques to complement and guide the experimental process is a crucial and fundamental step for biological analysis.

The final goal of the activity here presented is to provide a method that allows the identification of sites of possible protein-protein and protein-ligand interaction on the basis of the geometrical and topological structure of protein surfaces. The goal is then to discover complementary regions (that is with concave and convex segments that match each others) among different proteins. In particular, we are considering the first step of this process: the segmentation of the protein surface in protuberances and inlets through the analysis of convexity and concavity. To this end, two approaches will be described with a comparative assessment in terms of accuracy and speed of execution.

Keywords: protein-protein interaction, surface labeling, heat diffusion.

1 Introduction

There are currently about 50,000 experimentally determined 3D structures of proteins deposited in the Protein Data Bank (PDB) [14]. However this set contains a lot of identical or very similar structures. The importance of the study of structural building blocks, their comparison and classification are instrumental to the study on evolution and on functional annotation, has brought about many methods for their identification and classification in proteins of known structure.

In particular, there are several methods for defining protein secondary structure, and the DSSP [16] method is the most commonly used. The DSSP defines eight types of secondary structures, nevertheless, the majority of secondary prediction methods further simplify to the three dominant states: Helix, Sheet and Coil. Unfortunately, no standard definition is available of what a structural motif, a domain, a family, a fold, a sub-unit, a class [15], etc. really is, so that assignments have varied enormously, with each researcher using its own set of criteria. There are several DBs for structural classification of proteins; among them the most commonly used are SCOP and CATH. They differ in domain and class definition and also because the former is more based

P. Foggia, C. Sansone, and M. Vento (Eds.): ICIAP 2009, LNCS 5716, pp. 604–612, 2009.

on human expertise, whereas the latter is a semi-automatic classifier. Another well-known DB is FSSP, which is purely automatic [9].

An important research activity, with this large set of new proteins, is the prediction of interactions of these molecules by the discovery of similar or of complementary regions on their surfaces. When a novel protein with unknown function is discovered, bioinformatics tools are used to screen huge datasets of proteins with known functions and binding sites, searching for a candidate binding site in the new protein. More specifically, if a surface region of the novel protein is similar to that of the binding site of another protein with known function, the function of the former protein can be inferred and its molecular interaction predicted.

Much work has been done on the analysis of the binding sites of proteins and their identification, using various approaches based on different protein representations and matching strategies. The techniques employed are numerous ranging from geometric hashing of triangles of points and their associated physico-chemical properties [20], to clustering based on a representation of surfaces in terms of spherical harmonic coefficients [12] or by a collection of spin-images [2, 3] or by context shapes [10], to clique detection on the vertices of the triangulated solvent-accessible surface [1].

However, the shape descriptors used so far for surface matching are often too complex for real time analysis. A promising alternative approach, which we believe will be convenient to investigate, is the search for regions of interface that potentially correspond to the active sites, through the EGI introduced for applications of photometry by B. K. P. Horn [13] in the years '80 and which has been extended by K. Ikeuci [17, 21] in the years '90. To our knowledge the EGI has never been applied for protein representation.

The EGI is the histogram of the orientations placed on the unitarian sphere and it constitutes a compact and effective representation of a 3D object as a protein (or better the ligand that is usually a small molecule) or, as we plan to do, one of its part: the rotations of the object correspond to rotations of the EGI; the side surface of the object is the mass of the EGI; the barycentre of the EGI is in the sphere origin; every hemisphere is balanced by the complementary hemisphere; etc.. Just for these important properties, for the simplicity and the operational handiness on the unitarian sphere it is worth to verify if the conditions for docking conducted on the EGI are selective enough.

At the beginning, we do not think to adopt the extension of Ikeuci, Complex-EGI (C-EGI), because we consider that the hypothesis of model 'basically' convex (obviously, not always completely convex) can be applied in the search of matching between part of the proteins: the convexities of the ligand/protein and the complement to the concavity of the second protein. Moreover, also adopting the C-EGI, in presence of concavity [21], it is not guaranteed the biunivocity between C-EGI representation and 3D object.

For these reason, an effective way for computing the EGI representation is of great interest for the analysis of proteins convex and concave segments. The first phase of our planned activity for protein analysis that is presented in this paper is the development and the validation of algorithms to this purpose.

In literature, optimal 2D techniques for objects segmentation based on contour curvature, can be easily found. Nevertheless, their generalization to the 3D case is not trivial: an extra dimension will not only augment the computing time but due to the

extension from contour points to border lines the problem become even more difficult because border lines are not necessarily flat!

We are presenting here two tentative approaches and a performance comparison. The first one is an extension of a technique that has been introduced for 2D segmentation a few years ago [6] and that later it has been extended for multi-resolution detection, i.e. for the parallel detection of concavities and convexities at different resolution scales [5]. The second approach is a new solution, very simple, based on trivial near neighbor operation that can be easily implemented with 'ad hoc' hardware.

2 Labeling through Heat Diffusion Process Simulation

The first proposed method is based on the analogy of a heat-diffusion process acting through a material. The digital volume of the protein is considered like a solid isotropic metal governed by the heat equation. The equation is used to determinate concavity and convexity as respectively cold and hot points. The heat equation is a partial differential equation:

$$\frac{\partial u}{\partial t} = k\left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right] \tag{1}$$

where k is the diffusion coefficient.

The procedure starts by assigning a constant value to all the border points of the object and zero inside the object. Then a limited sequence of an isotropic adiabatic three-dimension diffusion process is performed towards the interior of the object. For simulating a discrete heat-diffusion process, equation (1) is converted to the following difference equation (2):

$$u_{t+1}(p) = u_t(p) + k\left[\sum_{q\epsilon N1}\left(u_t(p) - u_t(q)\right) + \sum_{q\epsilon N2}\frac{1}{\sqrt{2}}\left(u_t(p) - u_t(q)\right) + \sum_{q\epsilon N3}\frac{1}{\sqrt{3}}\left(u_t(p) - u_t(q)\right)\right] \tag{2}$$

where $u_t(p)$ represents the value of voxel p at time t, and $u_t(q)$ represents the value of voxel q, near neighbor of voxel p, at time t. With N1, N2 and N3 we distinguish three sets of near neighbors (NN): N1 constitutes of the six neighbors that share one face and have distance equal to 1 from the voxel p, N2 includes the twelve neighbors that share only an edge and are at distance $\sqrt{2}$, and N3 includes the neighbors than share only a vertex and are at distance $\sqrt{3}$ always from voxel p (see figure 1). The different weights of equation (2) are determined by the inverse of the voxels distance; this in order to simulate an isotropic diffusion process. Something quite similar has been presented by Borgefors and Sanniti di Baja [4] in which a subset of the 3x3x3 neighborhood includes only the N1 and N2 sets.

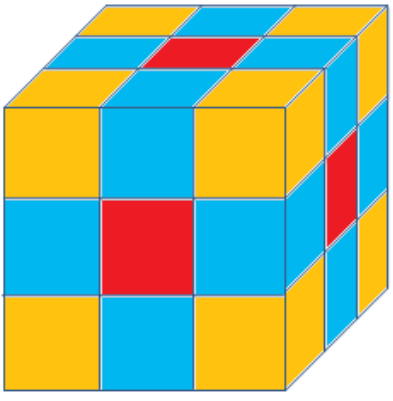

Fig. 1. Near neighbors of the voxel in the center of the elementary 3x3x3 cube: in evidence the three voxels of the set N1 in red; the nine voxels of the set N2 in blue; and the seven voxels of the set N3 in orange

At each iteration, new voxels, inside the object, are reached by the propagation process, according to equation 2. After a few number of iterations, the voxels that belong to local convexities will keep a higher value while those belonging to local concavities will have a value significantly reduced. Obviously, for $t \rightarrow \infty$ the object reaches a uniform heat distribution (i.e. constant temperature), so the number of step should be determined on the basis of the value of K (the diffusion coefficient) and of the size of the target details.

3 Iterative Detection of Convexities and Concavities

The Iterative Detection of Convexities and Concavities (IDCC) method here introduced is an algorithm that labels the voxels belonging to the border of the object according to their local convexities and concavities. Note that, also this approach constitutes a multi-resolution analysis: the higher the number of steps, the larger the receptive space involved for labeling the border voxels and the lower the details that can be analyzed. The approach is composed of two steps:

- the first step consists on labeling the border voxels of the object with the number of background voxels belonging to the 3x3x3 NN.[1]
- the second step consists on one to n iterations. At every sub-sequent step n, every border voxel is labeled with the sum of the labels at step n-1, within the 3x3x3 NN, divided by the number of the border voxels C, with label > 0 included in the NN.

[1] An alternative approach has been presented by Gallus and Neurath [11] in which the initial labeling is defined as the difference between the Freeman codes of pairs of successive contour points. Obviously this methods is not easily extendible in 3D.

The threshold $T = 9^n / C$ partition the border Voxels in the convex set (voxels having a label greater than T) and in the concave set (voxels having a label lower than T). Note that n is the number of iteration step. For a direct interpretation of the algorithm, in Figure 2 a simple visual example of a 2D implementation is shown.

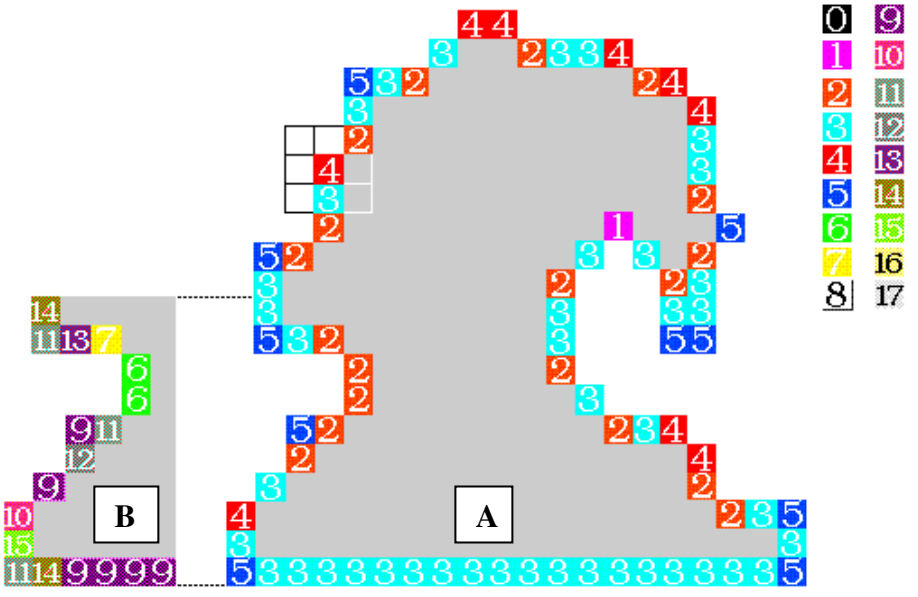

Fig. 2. Contour labeling of a 2D object by the IDCC algorithm. In A the first step is represented: the label of contour points, in 8-connection, is given by the number of background pixels (a 3x3 array is shown, putting in evidence the four background pixels). Pixels with a label greater than 3 are considered convex pixels, meanwhile pixels with label lower that 3 are considered concave pixels. In B the result obtained with the first iterative step is shown. The label of the contour point is given by the sum of the labels of the previous step, within the 3x3 surround. Pixels with a label greater than 9 are considered convex pixels, meanwhile pixels with label lower than 9 are considered concave pixels.

4 Experimental Results

Both algorithms are applied to the 'space-filling' representation of the protein, where atoms are represented as spheres with their Van der Walls radii (see figure 3). This representation is directly derived from PDB files which supply the ordered sequence of 3D positions of each atom's center. Figure 3 shows the image produced by our package for a molecule of 4PHV, a peptide-like ligand docked into HIV protease.

A first critical decision is the space resolution level for the analysis. For the first experiments we tried several voxels sizes ranging from 0.10 up to 0.75 Å, but the results here presented are given with a resolution of 0.25 Å that allows to the smallest represented atoms, a Van der Walls radius of more than five pixels. Obviously, the

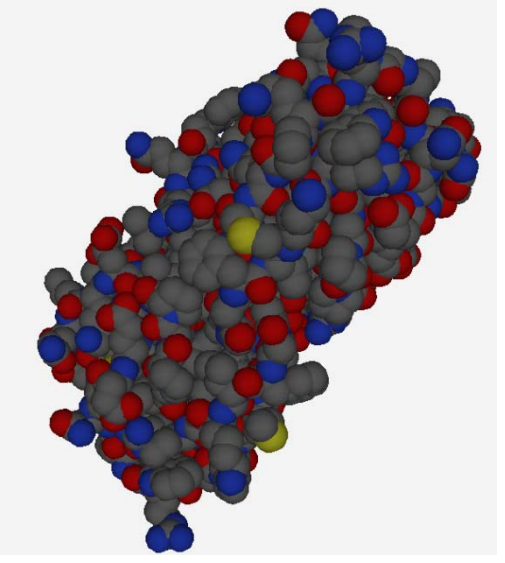

Fig. 3. 'Space filling' representation of 4PHV protein. The colors follow the standard CPK scheme.

higher the voxel size, the lower the details on the surface representation, but the faster the algorithms convergence to an acceptable result. In figure 4.1, 4.2, 4.3 and 5.1, 5.2, 5.3 the results for IDCC and Heat Diffusion processes on the 4PHV molecule are shown for different numbers of iterations.

Let first point out that both the approaches lead to a segmentation useful for docking: as expected the small details that characterizes the original surface (with apexes and cusp at the convergence of different atoms spheres) of figure 3 are overcome as the iterations of the diffusion process reaches distances that are one order of magnitude of the Van der Walls radii; protuberances and fiords are well characterized and evident as well as than the site of possible protein-{ligand, protein} interaction.

The experiments show clearly that heat diffusion approach has a faster convergence to an acceptable solution, than the IDCC methodology. In fact, 300 iterations of IDCC method (figure 4.3) produce a result qualitatively similar to only 100 iterations of heat diffusion algorithm (figure 5.1). In other words, the two methods, with a suitable different number of iteration, reach almost the same result.[2]

Both algorithms were tested on a 2.20 GHz x86 Intel processor. A single iteration of heat diffusion algorithm takes on average 5.4 sec while a single iteration of IDCC takes 7.5 sec. This, combined with a faster convergence, makes, for standard hardware, the heat diffusion method a better approach than IDCC.

[2] The time performance of the IDCC method are improved limiting the calculus to the subset N1 and N2 as indicated above [4], reaching a reduction of the computation time of 40%. This is not sufficient to reach the performances of the heat diffusion approach. Note that not including the set N3 an extra number of iterations is required to reach the same results.

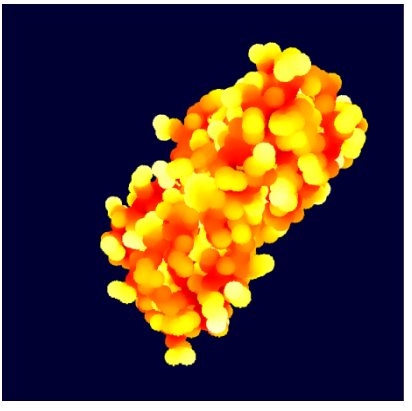

Fig. 4. 1

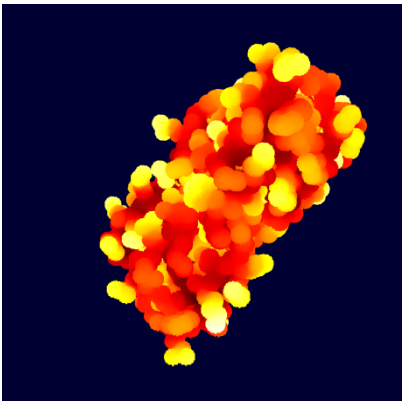

Fig. 4. 2

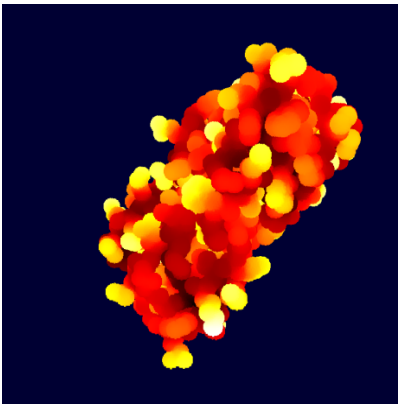

Fig. 4. 3

Fig. 4. Experimental results obtained with the IDCC approach. The values of the execution parameters are:

Voxel Side = 0.25 Å

Number of iterations: 100 (fig. 4.1), 200 (fig. 4.2), 300 (fig. 4.3)

Color scheme: intensity data using a "*Hot Iron*" color mapping, that is:
yellow → white : higher convexities
red → black: higher concavities

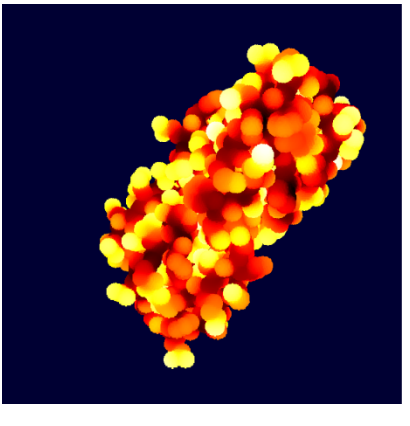

Fig. 5. 1

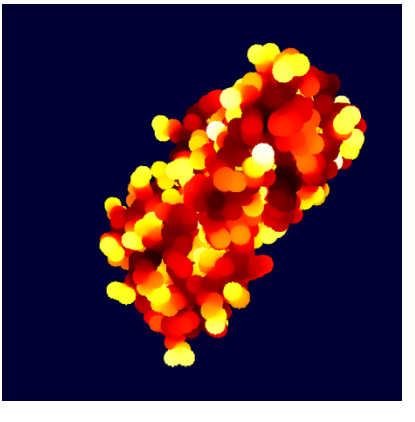

Fig. 5. 2

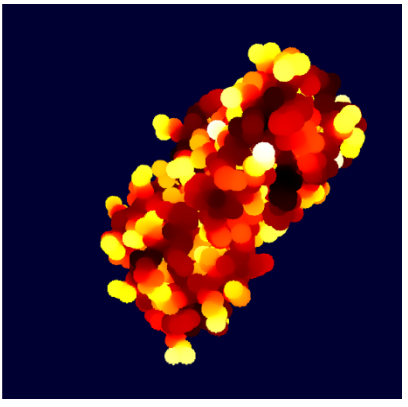

Fig. 5. 3

Fig. 5. Experimental results obtained with the Heat Diffusion approach. The values of the execution parameter are:

Voxel Side = 0.25 Å
K = 0.05
Number of iterations: 100 (fig. 5.1), 200 (fig. 5.2), 300 (fig. 5.3)

Color scheme: intensity data using a "*Hot Iron*" color mapping, that is:
yellow → white : higher convexities
red → black: higher concavities

Nevertheless, it is important to point out that both algorithms works locally and independently for each voxel. Therefore an implementation on a parallel architecture can be done without particular problem in partitioning the computation load and a great speed improvement can be easily obtained. Moreover, note that the IDCC approach looks suitable for a single chip implementation; in fact only trivial basic operations are required. This should be possibly taken into account in the future if the complete proteins interaction analysis will be successful.

5 Conclusion

The activities in proteomics are in an intensive development; for example the PDB has an annual growth rate that reaches the 3000 new proteins experimentally determined and deposited. So it is becoming increasingly necessary to reach high performance in research and analysis of these massive databases. Potential applications closely depend on the performance of the problem here discussed. The aim of our research is to provide a substantial increase of current performances for the comparison of protein surfaces through the development of new methods of analysis. The achieved results testify that the presented solutions are very promising.

References

1. Akutsu, T.: Protein structure alignment using dynamic programming and iterative improvement. IEICE Trans. Inf. and Syst. E78-D, 1–8 (1996)
2. Bock, M.E., Garutti, C., Guerra, C.: Spin image profile: a geometric descriptor for identifying and matching protein cavities. In: Proc. of CSB, San Diego (2007)
3. Bock, M.E., Garutti, C., Guerra, C.: Cavity detection and matching for binding site recognition. Theoretical Computer Science (2008), doi:10.1016/j.tcs.2008.08.018
4. Borgefors, G., Sanniti di Baja, G.: Analysing non-convex 2D and 2D patterns. Computer Vision and Image Understanding 63(1), 145–157 (1996)

5. Cantoni, V., Cinque, L., Guerra, C., Levialdi, S., Lombardi, L.: 2D Object recognition by multiscale tree matching. Pattern Recognition 31(10), 1443–1454 (1998)
6. Cantoni, V., Levialdi, S.: Contour labelling by pyramidal processing. In: Duff, M.J.B. (ed.) Intermediate-level Image Processing, ch. XI, pp. 181–190. Academic Press, New York (1986)
7. Coleman, R.G., Burr, M.A., Sourvaine, D.L., Cheng, A.C.: An intuitive approach to measuring protein surface curvature. Proteins: Struct. Funct. Bioinform. 61, 1068–1074 (2005)
8. Connolly, M.L.: Measurement of protein surface shape by solid angles. J. Mol. Graphics 4, 3–6 (1986)
9. Day, R., Beck, D.A., Armen, R.S., Daggett, V.: A consensus view of fold space: combining SCOP, CATH, and the Dali Domain Dictionary. Protein Sci. 12(10), 2150–2160 (2003)
10. Frome, A., Huber, D., Kolluri, R., Baulow, T., Malik, J.: Recognizing Objects in Range Data Using Regional Point Descriptors. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3023, pp. 224–237. Springer, Heidelberg (2004)
11. Gallus, G., Neurath, P.W.: Improved computer chromosome analysis incorporating preprocessing and boundary analysis. Phys. Med. Biol. 15, 435–445 (1970)
12. Glaser, F., Morris, R.J., Najmanovich, R.J., Laskowski, R.A., Thornton, J.M.: A Method for Localizing Ligand Binding Pockets in Protein Structures. PROTEINS: Structure, Function, and Bioinformatics 62, 479–488 (2006)
13. Horn, B.K.P.: Extended Gaussian images. Proc. IEEE 72(12), 1671–1686 (1984)
14. `http://www.pdb.org/` (visited, April 2009)
15. Jacob, F.: Evolution and tinkering. Science 196, 1161–1166 (1977)
16. Kabsch, W., Sander, C.: Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. Biopolymers 22(12), 2577–2637 (1983)
17. Kang, S.B., Ikeuchi, K.: The complex EGI: a new representation for 3-D pose determination. IEEE-T-PAMI, 707–721 (1993)
18. Masuya, M.: Shape Analysis of Protein Molecule and Legand-Receptor Docking Studies Using Mathematical Morphology, Doctoral Thesis, The University of Tokyo (1996)
19. Nicholls, A., Sharp, K.A., Honig, B.: Protein folding and association: insights from the interfacial and thermodynamic properties of hydrocarbons. Proteins 11, 281–296 (1991)
20. Shulman-Peleg, A., Nussinov, R., Wolfson, H.: Recognition of Functional Sites in Protein Structures. J. Mol. Biol. 339, 607–633 (2004)
21. Shum, H., Hebert, M., Ikeuchi, K.: On 3D shape similarity. In: Proceedings of the IEEE-CVPR 1996, pp. 526–531 (1996)
22. Sridharan, S., Nicholls, A., Honig, B.: A new vertex algorithm to calculate solvent accessible surface area. Biophys. J. 61, A174 (1992)
23. Takeshi, K.: Multi-scale Pocket Detection on Protein Surface Using 3D Image Processing Technique. IPSJ SIG 2006(99) (BIO-6), 49–56 (2006)