An Effective Algorithm Based on Curve Fitting for Measuring Protein Structure Similarity*

Yu-Chieh Lo, Chang-Biau Yang† and Yung-Hsing Peng
Department of Computer Science and Engineering
National Sun Yat-sen University, Kaohsiung, Taiwan
†cbyang@cse.nsysu.edu.tw

Abstract

It is almost believed that the function of one protein is determined by its structure. The more similar two protein structures are, the more similar their functions are. The distance RMSD (Root Mean Square Deviation) is a popular method used by most researchers to measure the distance (or similarity) between two protein structures, usually one is the predicted structure and the other is its real structure. In this paper, we propose a new algorithm to compare two protein structures, which is the combination of sequence alignment and the B-spline curve fitting in the space. To test and verify our method, we randomly choose some families in the CATH database and try to identify them. Experimental results show that our method outperforms the distance RMSD method. Furthermore, we apply the SVM (Support Vector Machine) tool to help us obtain the better classifications.

Key words: protein structure, B-spline, dynamic programming, RMSD, SVM

1 Introduction

Proteins consist of twenty kinds of amino acids. The protein structure can be divided into four levels, where the *primary structure* is the amino acid sequence. Because different amino acids in different positions cause the force change between molecules, different sequences may yield different folding. In general, the *secondary structure* is formed due to different permutations in protein segment. There are three main models for the secondary structure, which are *α-helix* (a spiral structure), *β-sheet* (a flat structure) and *turn*. The *tertiary structure* is formed by the interaction of several secondary structures. The tertiary structure is usually represented by the 3D coordinates of the C_α atoms in amino acids, which is the backbone of the protein structure. The *quaternary structure* indicates a complex of two or more tertiary structures, which form a complete functional structure.

From biological perspective, a protein structure determines its function. However, protein structures need to be confirmed through biological experiments, such as X-ray Crystallography and NMR [14, 7], which not only takes efforts but also costs a lot of money. Therefore, many researchers have proposed many computational methods for predicting the 3D structure of proteins.

For a prediction algorithm, to know if the predicted structure is correct or not, we usually test the algorithm by selecting a target protein whose structure has already been known. When the prediction is finished, one can compare the predicted protein 3D structure with the real answer. The smaller difference between these two structures is, the better performance the method has. Thus, a 3D structure similarity measurement tool plays an important role to judge the effectiveness of the prediction method.

Nowadays, most researchers use the *distance RMSD* (Root Mean Square Deviation) [9] as a standard method to evaluate the difference or similarity between two structures, one is the predicted structure and the other is its real structure. The distance RMSD assumes that the two structures to be compared have the same length (equal number of amino acids), and it compares the two structures with the 1-1 mapping style. Though the two structures differ only at a local point and the two pairs of segments broken by the local point are almost the same, the two structures may be determined to have large difference by the distance

*This research work was partially supported by the National Science Council of Taiwan under NSC-95-2221-E-110-102.

RMSD. In other words, the distance RMSD is very sensitive to local difference.

To eliminate the influence of local difference on two structures may lead to more effective structure comparisons. To achieve better structure comparison, in this paper, we propose a new algorithm based on curve alignment, which is a combination of sequence alignment and B-spline curve fitting. To test and verify our method, we randomly choose some families in the CATH database as testing sets and try to identify them. Experimental results show that our method outperforms the distance RMSD method. Furthermore, we also invoke the SVM (Support Vector Machine) tool to help us obtain the better classifications.

The organization of this paper is described as follows. In Section 2, we will briefly introduce some preliminary knowledge for protein structure comparison, including RMSD and the B-spline curve. In Section 3, we shall propose our algorithm for computing the distance score of two protein structures. Our experimental results will be presented in Section 4. Finally, a conclusion will be given in Section 5.

2 Preliminary

2.1 The Root Mean Square Deviation

It is assumed that the 3D structure of a protein is defined by the 3D coordinates of atoms C_α's of amino acids in the protein. Given two protein 3D structures with same length, it is an intuitive way to examine the similarity between them by superimposing each pair of corresponding points (residue) in 3D space. RMSD (Root Mean Square Deviation) is a simple way for computing the distance of two 3D structures as follows. [11, 15]:

$RMSD = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(r_i^a - r_i^b)^2}$,

where $(r_i^a - r_i^b)$ denotes the 3D distance between the ith residues of protein a and protein b and n denotes the length of the two proteins. Note that the two proteins to be computed must have the same length.

However, the superimposition of the two protein structures is difficult. Therefore, the *distance RMSD* [9], a variation of RMSD, was proposed to compute the distance of two protein structures as follows.

$DRMSD = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}(x_{ij}^a - x_{ij}^b)^2}$,

where x_{ij}^a denotes the distance between the ith and jth residue in protein a and x_{ij}^b denotes the distance between the ith and jth residue in protein b.

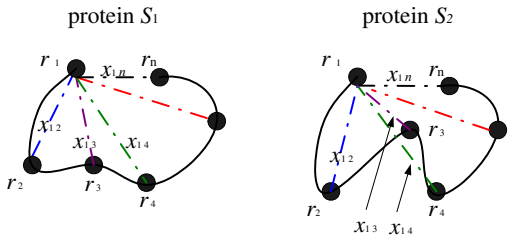

Figure 1: Computing the distance between two proteins with the distance RMSD.

Figure 1 illustrates the computation of the distance RMSD. As we can see, the distance RMSD eliminates the difficulty of superimposition. Thus, it becomes the most popular method to measure the distance or similarity of two protein structures with equal length. The smaller the distance between the two structures is, the more similar in structure they are.

2.2 The B-Spline Curve

The B-spline curve is used to fit a given set of points, which is defined as follows[5]:

$U(v) = \sum_{k=0}^{n} P_i N_{i,d}(v)$

$N_{i,d}(v)$: blending function

$d-1$: degree of the curve (usually 3 or 4)

P_i: control points

$n+1$: number of control points

The B-spline curve has the following characteristics [5]:

1. The B-spline curve is changed by the control points and lies within the convex hull of the control points.

2. The B-spline curve has the ability for local control, as shown in Figure 2.

3. The degree of a B-spline curve is not limited to the numbers of control points.

In general, more structural variations can be revealed by using B-spline curves to represent protein structures. Based on this idea, in the following section, we propose our algorithm that compares two protein structures by using B-spline curves.

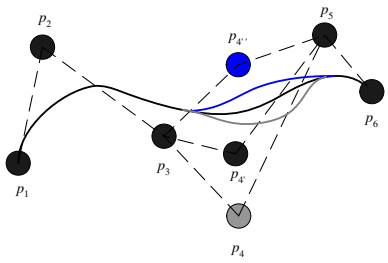

Figure 2: Good ability of local control of a B-spline curve.

3 Structure Comparison by Using the B-Spline Curve

Our structure comparison algorithm is the combination of sequence alignment and curve fitting with B-spline curves. Thus, it is not necessary that the two input structures have the same length. The main steps of our algorithm for structure comparison with B-spline (SCB) is summarized as follows:

Algorithm: SCB

Input: Two protein structures S_1 and S_2, represented by 3D coordinates.

Output: The distance score between S_1 and S_2

Step 1 Curve Fitting: From S_1 and S_2, extract all segments of four successive residues. Then, calculate the B-spline curve of each segment (four points).

Step 2 Curve Matching: Compute the distance for each pair of B-spline curves, one from S_1 and the other from S_2.

Step 3 Structure Alignment: Perform the structure alignment on S_1 and S_2, and obtain the distance score of these two input structures.

Step 4: Normalization: Normalize the distance score.

3.1 Curve Fitting

We compute the B-spline curve formed by every four consecutive residues in S_1 and S_2. As shown in Figure 3, when computing the curve representing residue r_i, we have to get four residues r_{i-1}, r_i, r_{i+1} and r_{i+2}, where $2 \leq i \leq L_1 - 2$, or $2 \leq i \leq L_2 - 2$, and L_1 and L_2 denote the lengths

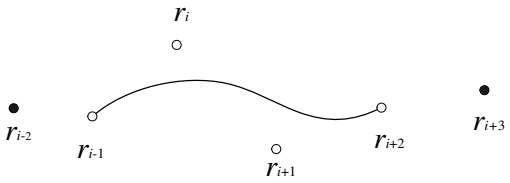

Figure 3: The B-spline curve formed by four consecutive residues.

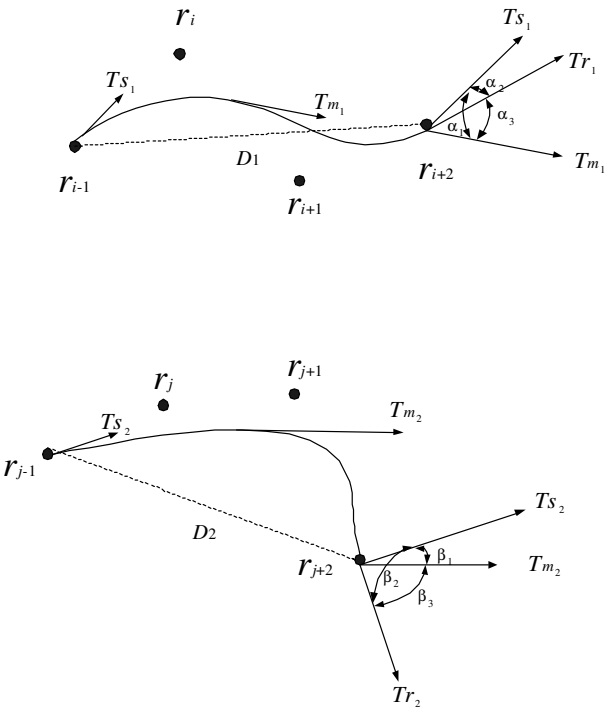

Figure 4: Curve matching.

of S_1 and S_2, respectively. After extracting the 3D coordinates of four successive residues from either S_1 or S_2, we can calculate a smooth curve for these four residues in 3D space, by using the B-spline curve function.

3.2 Curve Matching

We present a new method of curve matching, as shown in Figure 4, which is similar to the methods proposed by Hagen [6] and Sebastian et al. [13]. Our algorithm for computing the distance between two curves is given as follows.

1. Compute the start tangents T_{s_2} and T_{s_1}, middle tangents T_{m_1} and T_{m_2} ,tail tangents T_{r_1} and T_{r_2} of these two curves, respectively.

2. Compute angles α_1 of T_{s_1} and T_{m_1}, α_2 of T_{s_1}

and T_{r_1}, α_3 of T_{m_1} and T_{r_1} and β_1 of T_{s_2} and T_{m_2}, β_2 of T_{s_2} and T_{r_2}, β_3 of T_{m_2} and T_{r_2}.

3. Compute the Euclidean distance D_1 of r_{i-1} and r_{i+2} from S_1 and the Euclidean distance D_2 of r_{j-1} and r_{j+2} from S_2.

The distance score of the curve matching between the curves for the ith residue in protein S_1 and the jth residue in protein S_2, denoted as K_{ij}, is defined as

$$K_{ij} = |D_2 - D_1| + \gamma(|\alpha_1 - \beta_1| + |\alpha_2 - \beta_2| + |\alpha_3 - \beta_3|),$$

where γ is a constant to give weight between Euclidean distance and angle ($\gamma = 0.05$ in our experiment).

3.3 Structure Alignment

In order to get the finest structure alignment, here we apply the dynamic programming approach. Similar to sequence alignment, when the ith residue in S_1 is aligned to the jth residue in S_2, we will calculate K_{ij} by taking two B-spline curves, formed by the ith residue in S_1 and the jth residue in S_2, respectively. Therefore, the formula of our dynamic programming for structure alignment can be written as

$$M[i,j] = min(M[i-1,j] + penalty, M[i,j-1] + penalty, M[i-1,j-1] + K_{i,j}),$$

where *penalty* is a constant and its meaning is similar to the gap penalty in sequence alignment ($penalty = 5$ in our experiment). Finally, the value of $M[L_1, L_2]$ can be obtained and it represents the distance score of the two input protein structures.

3.4 Normalization

It is not suitable if we directly take $M[L_1, L_2]$ as the distance score of these two proteins, because higher distance score may be obtained from longer sequences. Therefore, we adopt normalization when we compute the distance score as follows:

$$distance\ score = \frac{M[L_1,L_2]}{min(L_1,L_2)}$$

4 Performance Comparison

Some databases, such as SCOP [8] and CATH [10], made a classification on proteins according to their structures. Those are structurally similar are classified into the same category. In this paper, we extract some proteins from CATH as our testing sets to test and verify our method. CATH clusters protein domain structures into four major levels, which are Class (C), Architecture (A), Topology (T) and Homologous superfamily (H). In our experiments, we adopt the grouping standard of the H-level. In the H-level, domain proteins which belong to the same group have high similarity in their structures. Because of the high similarity, there are only few domain proteins in each group. To achieve sensitive experiments, we obtain our testing data by randomly choose some families in the H-level, each contains at least 10 domain proteins, as shown in Table 1.

4.1 Methods for Comparing

In our experiments, we compare the performance of classification sensitivity of four methods, including sequence alignment, Fit4RMSD, distance RMSD and our method. Here, the sequence alignment method[1, 2, 16] adopts the scoring matrix PAM250[4, 12]. In fact, it is a tool used for comparing sequences rather than structures. In the distance RMSD method, the two given proteins to be compared should be of the same length. However, our testing proteins almost have different lengths. Thus, some modifications should be done for distance RMSD. If two given proteins are not of the same length, we apply the sliding window scheme that the shorter protein is slid on the longer one to find a best segment for computing the distance RMSD. The Fit4RMSD method is the combination of distance RMSD and sequence alignment. Fit4RMSD extracts all segments, each composed of four successive residues, then calculates the distance RMSD for each pair of 4-residue segments from two families. Next, we perform the structure alignment based on the dynamic programming approach and obtain the distance score of Fit4RMSD. Finally, we normalize the distance score in each method through dividing it by the smaller length of the two input proteins.

For each pair of families, we dynamically determine the best threshold to identify these two groups of domain proteins. For example, suppose there are two families $A = \{a_1, a_2, ..., a_m\}$ and $B = \{b_1, b_2, ..., b_n\}$, where each a_i or b_j represents one domain protein. For these two families, we not only perform inter-comparison (i.e. comparing a_i with b_j, where $1 \leq i \leq m$, $1 \leq j \leq n$), but also perform intra-comparison (i.e. comparing a_i with

Table 1: The families of CATH for testing. Number: the number of domain proteins in each family. Max length: the maximum length of domain proteins in each family. Min length: the minimum length of domain proteins in each family. Average length: the average length of domain proteins in each family.

	CATH code	Number	Max length	Min length	Average length
a	1.10.8.10	21	99	18	50
b	1.10.20.10	21	147	45	90
c	1.20.5.110	11	79	52	64
d	1.20.5.170	18	70	31	51
e	1.20.5.50	14	192	43	79
f	2.10.25.10	35	86	38	52
g	2.10.60.10	12	92	60	63
h	2.10.70.10	23	76	49	60
i	2.10.90.10	12	175	85	110
j	2.40.10.10	87	140	19	102
k	3.10.20.30	23	128	63	91
l	3.10.20.90	30	165	53	94
m	3.10.28.10	14	161	74	116
n	3.10.50.40	13	125	85	110
o	3.30.70.20	21	133	58	88

a_j, where $i \neq j$, $1 \leq i, j \leq m$, comparing b_p with b_q, where $p \neq q$, $1 \leq p, q \leq n$). For each comparison, we compute the distance scores of the above four methods. With these scores and a proper threshold, we can guess whether the two given domain proteins are in the same family or not. Note that the threshold depends on the two families and the method used for comparison. Figure 5 illustrates the flowchart of testing.

There are $C_2^m + C_2^n$ intra-comparisons and mn inter-comparisons. The intra-accuracy and inter-accuracy are defined as the numbers of correct guesses on intra-comparisons and inter-comparisons divided by $C_2^m + C_2^n$ and mn, respectively. We set the threshold to the value that achieves equal intra-accuracy and inter-accuracy. For each method, the overall accuracy is defined as the total number of correct guesses divided by $C_2^n + C_2^m + mn$.

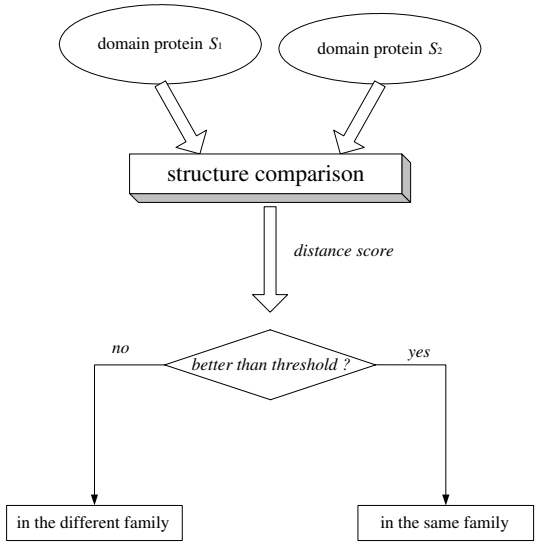

Figure 5: The flowchart of testing.

4.2 The SVM Tool

In addition to the four methods mentioned in the previous subsection, the SVM tool, Libsvm-2.82 [3] is also used to classify protein family. We have four dimensional features for the SVM tool, which are the scores obtained from the above four methods. To examine the effect of these four features, we form three versions of SVM features as follows.

SVM with two features: our method and distance RMSD.

SVM with three features: our method, distance RMSD and sequence alignment.

SVM with four features: our method, distance RMSD, sequence alignment and Fit4RMSD.

4.3 Experimental Results

Since the distance RMSD method is applied to the calculation of two protein structures with the same length, we think that the lengths of input proteins may affect the accuracy in our experiments. Thus, our experiments contains two cases. In the first case, all proteins those we have extracted from CATH are in the testing set. In other words, there is no restriction on the protein length. In the second case, we make some restriction on testing proteins that all proteins outside the lengths 65-105 are removed. In other words, the second case tests the proteins with similar lengths.

For the first experiment case, Table 2 and Table 3 list the accuracy of each pairs of family for distance RMSD and our method, respectively. In each table, there are 105 entries since 15 families are chosen for testing and our classification testing is performed on each pair of families.

The statistical results of each tested method are shown in Table 4, which compares the performance of the seven methods. Each method has two columns, the left column represents the occurrence number of accuracy within this range and the right column represents the accumulated percentage. For example, the meaning of the intersection of column D (our method) and row 90% is that 11 entries of Table 3 have accuracy w, $90\% \leq w < 95\%$, and the accumulated percentage is $(16 + 25 + 11)/105 = 49.5\%$.

As we can see in Table 4, if there is only one feature used to compare the protein structures, our method outperform all others. However, SVM can help us achieve more accurate comparisons, if more features are involved. The result shows that SVM with four features achieves the highest accuracy.

In the second experiment case, which experiments are performed on proteins with similar lengths(65-105), the results are better, as shown in Table 5. In this experiment, some families are eliminated because they contain too few proteins. One can clearly see that the accuracy of this experiment is higher than that of the previous one. In the first experiment, the accuracy is lower because some false-positive results may be yielded if one domain protein contains another one. Since a protein is not likely to contain another protein if they are of similar sizes, the number of false-positives in the second experiment should decrease. In other words, the classification will be more accurate for two families of proteins of similar size.

In our both experimental results, we can see that our method for computing the distance between two proteins is more sensitive than the other three one-feature methods. The distance RMSD is almost used a standard to measure the similarity of two protein structures, one is predicted and the other is real. With our experiment results, we can conclude that our method outperforms the distance RMSD method.

It is natural that if we combine more features to measure the similarity between two proteins, we will get better results. Thus, the SVM with four features achieves the highest accuracy.

5 Conclusion

In this paper, we propose a method to compare the protein structures based on B-spline curves. Our method not only computes the distance of residues like distance RMSD, but also considers the angle of the curve. By combining with the sequence alignment scheme, our method is able to compare proteins of different sizes, while the distance RMSD is not. From the experimental results, one can see that our method outperforms the distance RMSD.

In fact, our method can be another approach for the comparison of protein structures. In addition, if some useful features of protein structures can be extracted, SVM is an efficient tool to do classification. Our experiment results show that our method can serve as a good feature in SVM for protein classification. To raise the accuracy, designing good features for protein comparison is still worthy of study in the future.

References

[1] T. Akutsu and H. Arimura, "On approximation algorithms for local multiple alignment," *Proceedings of the Fourth Annual International Conference on Computational Molecular Biology*, Tokyo, Japan, pp. 1–7, 2000.

[2] V. Bafna, E. L. Lawler, and P. Pevzner, "Approximation algorithms for multiple sequence alignment," *Proc. of 5th Ann. Symp. On Pattern Combinatorial Matching*, Vol. 807, pp. 43–53, 1994.

Table 2: The accuracy of distance RMSD (%).

	b	c	d	e	f	g	h	i	j	k	l	m	n	o
a	77	99	80	91	75	72	79	68	65	66	68	66	65	67
b		94	76	77	81	93	87	73	86	93	91	81	84	85
c			56	68	94	100	100	93	100	100	100	100	100	100
d				54	81	86	88	66	83	84	84	80	82	85
e					85	98	97	71	94	95	96	89	94	96
f						65	75	65	62	62	66	63	65	62
g							74	77	64	64	74	70	66	71
h								74	67	76	82	75	69	77
i									80	79	79	74	80	77
j										73	69	64	66	65
k											70	76	69	69
l												77	68	72
m													66	68
n														67

Table 3: The accuracy of our method (%).

	b	c	d	e	f	g	h	i	j	k	l	m	n	o
a	64	76	66	62	99	100	100	100	95	98	99	72	98	92
b		79	68	69	95	100	100	99	97	97	96	73	91	90
c			55	70	100	100	100	100	100	100	100	77	99	97
d				58	99	100	100	100	96	99	100	75	98	94
e					97	98	99	99	96	97	98	75	96	92
f						70	78	75	75	71	85	90	78	85
g							79	71	71	82	83	94	81	84
h								74	79	86	92	97	88	92
i									73	84	84	93	80	89
j										78	81	90	76	75
k											77	85	77	78
l												82	81	84
m													83	78
n														83

Table 4: The accuracy of tested methods on proteins without length restriction. A: sequence alignment; B: Fit4RMSD; C: distance RMSD; D: our method ; E: SVM with two features (C, D); F: SVM with three features (A, C, D); G: SVM with four features (A, B, C, D). Each method has two columns, the left column means the occurrence number of accuracy within this range and the right column means the accumulated percentage.

Accuracy	A		B		C		D		E		F		G	
100 %	0	0%	0	0%	8	7.6%	16	15.2%	15	14.3%	19	18.1%	17	16.2%
95 %	0	0%	0	0%	6	13.3%	25	39.0%	31	43.8%	31	47.6%	31	45.7%
90 %	3	2.9%	5	4.8%	9	21.9%	11	49.5%	9	52.4%	11	58.1%	21	65.7%
85 %	2	4.8%	7	11.4%	8	29.5%	6	55.2%	14	65.7%	18	75.2%	21	85.7%
80 %	7	11.4%	15	25.7%	13	41.9%	13	67.6%	18	82.9%	15	89.5%	12	97.1%
75 %	8	19.0%	18	42.9%	15	56.2%	18	84.8%	10	92.4%	6	95.2%	2	99.0%
70 %	15	33.3%	16	58.1%	12	67.6%	9	93.3%	5	97.1%	2	97.1%	1	100%
65 %	16	48.6%	20	77.1%	25	91.4%	3	96.2%	1	98.1%	2	99.0%	0	100%
60 %	14	61.9%	19	95.2%	7	98.1%	2	98.1%	2	100%	1	100%	0	100%
55 %	13	74.3%	5	100%	1	99.0%	2	100%	0	100%	0	100%	0	100%

Table 5: The accuracy of tested methods on length-restricted proteins. The column meanings are the same as those in Table 4.

Accuracy	A		B		C		D		E		F		G	
100 %	4	5.1%	3	3.8%	24	30.8%	18	23.1%	25	32.1%	27	34.6%	34	43.6%
95 %	4	10.3%	3	7.7%	2	33.3%	8	33.3%	6	39.7%	4	39.7%	12	59.0%
90 %	6	17.9%	10	20.5%	5	39.7%	10	46.2%	11	53.8%	13	56.4%	12	74.4%
85 %	5	24.4%	7	29.5%	5	46.2%	15	65.4%	11	67.9%	14	74.4%	12	89.7%
80 %	19	48.7%	11	43.6%	10	59.0%	12	80.8%	7	76.9%	10	87.2%	5	96.2%
75 %	8	59.0%	12	59.0%	9	70.5%	6	88.5%	9	88.5%	3	91.0%	1	97.4%
70 %	6	66.7%	9	70.5%	8	80.8%	6	96.2%	4	93.6%	4	96.2%	2	100%
65 %	11	80.8%	9	82.1%	7	89.7%	2	98.7%	3	97.4%	1	97.4%	0	100%
60 %	8	91.0%	12	97.4%	8	100%	1	100%	1	98.7%	2	100%	0	100 %
55 %	5	97.4%	2	100%	0	100%	0	100%	0	98.7%	0	100%	0	100%

[3] C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[4] M. O. Dayhoff, W. C. Barker, and L. Hunt, "Establishing homologies in protein sequences," *Methods Enzymol*, Vol. 91, pp. 524–545, 1983.

[5] G. Farin, *Curves and Surfaces for Computer Aided Geometric Design : A Practical Guide*. Boston: Academic Press, second ed., 1990.

[6] H. Hagen, *Curves and Surfaces Design*. SIAM Activity Group on Geometric Design, 1992.

[7] R. C. T. Lee, "Computational biology." http://www.csie.ncnu.edu.tw/, Department of Computer Science and Information Engineering, National Chi-Nan University, Taiwan, 2001.

[8] A. G. Murzin, S. E. Brenner, T. Hubbard, and C. Chothia, "SCOP: a structural classification of proteins database for the investigation of sequences and structures," *Journal of Molecular Biology*, Vol. 247, pp. 536 – 540, 1995.

[9] K. Nishikawa and T. Ooi, "Comparison of homologous tertiary structures of proteins," *Journal of Theoretical Biology*, Vol. 43, pp. 351–374, 1974.

[10] C. A. Orengo, A. D. Michie, S. Jones, D. T. Jones, M. B. Swindells, and J. M. Thornton, "CATH- a hierarchic classification of protein domain structures," *Structure*, Vol. 5, No. 8, pp. 1093 – 1108, 1997.

[11] S. T. Rao and M. G. Rossmann, "Comparison of super-secondary structures in proteins," *Journal of Molecular Biology*, Vol. 76, pp. 241–256, 1973.

[12] R. M. Schwartz and M. O. Dayhoff, *Matrices for detecting distant relationships.* In M. Dayhoff, editor, Atlas of Protein Sequence and Structure, Volume 5, pages 353-358. National Biomedical Research Foundation, Washington, DC, USA, 1979.

[13] T. B. Sebastian, P. N. Kelin, and B. Kimia, "Alignment-based recognition of shape outlines.," *Proceedings of 4th International Workshop on Visual Form*, Capri, Italy, pp. 606–618, 2001.

[14] J. Setubal and J. Meidanis, *Introduction to Computational Molecular Biology*. PWS Publishing Company, Boston, second ed., 1997.

[15] W. R. Taylor and C. A. Orengo, "Protein structure alignment.," *Journal of Molecular Biology*, Vol. 208, pp. 1–22, 1989.

[16] U. Tonges, S. W. Perrey, J. Stoye, and A. W. M. Dress, "A general method for fast multiple sequence alignment," *Gene*, Vol. 172, No. 1, pp. 33–41, 1996.