한국 CAD/CAM 학회 논문집
제 14권 제 6호 2009년 12월 pp. 404-414

분자 데이터베이스 스크리닝을 위한 원자간 거리 기반의 3차원 형상 기술자

이재호*, 박준영**

3D Shape Descriptor with Interatomic Distance for Screening the Molecular Database

Jaeho Lee* and JoonYoung Park**

ABSTRACT

In the computational molecular analysis, 3D structural comparison for protein searching plays a very important role. As protein databases have been grown rapidly in size, exhaustive search methods cannot provide satisfactory performance. Because exhaustive search methods try to handle the structure of protein by using sphere set which is converted from atoms set, the similarity calculation about two sphere sets is very expensive. Instead, the filter-and-refine paradigm offers an efficient alternative to database search without compromising the accuracy of the answers. In recent, a very fast algorithm based on the inter-atomic distance has been suggested by Ballester and Richard. Since they adopted the moments of distribution with inter-atomic distance between atoms which are rotational invariant, they can eliminate the structure alignment and orientation fix process and perform the searching faster than previous methods. In this paper, we propose a new 3D shape descriptor. It has properties of the general shape distribution and useful property in screening the molecular database. We show some experimental results for the validity of our method.

Key words : molecule database, shape descriptor, shape similarity, inter-atomic distance

1. Introduction

The geometry of biological systems, such as the geometry of molecules, is a very important consideration when investigating the functions of these systems. Molecules such as protein, DNA, and RNA consist of atoms. 3D structural comparison and structural database searching of proteins play very important roles. For example, researchers may want to search an unknown protein against a database of functionally annotated proteins to infer its functions from those found to be structurally similar to it. In general, structural database searching has many applications in the area of drug discovery. It can be used to verify the 3D structure of a target drug which is modeled by structural prediction[4]. It can also be used to identify the similar fold structure and families unique to pathogenic organisms to select good drug targets[7], etc.

An additional advantage of searching a database for molecules with similar shape in that no specification of chemical structure, such as types of atoms or their bond arrangement, is made and therefore similarly shaped molecules, but with different chemical scaffolds from the template, can be found. Such ability, known as scaffold hopping, is very crucial[9]. Techniques for scaffold hopping can be used to hop molecules with different scaffolds when leading compound have desirable features such as intractable chemistry or poor pharmacological properties[12].

In recent, advances in molecular structure analysis methods such as MNR and X-ray crystallography have been developed and contributed to a significant increase in the number of known protein 3D structures. Especially, the Protein Structure Data Bank (PDB) stores over 45,000 structures[8]. When the size

*정회원, 동국대학교 디지털제품연구실
**교신저자, 종신회원, 동국대학교 산업시스템공학과
- 논문투고일: 2008. 03. 21
- 논문수정일: 2009. 09. 01
- 심사완료일: 2009. 10. 08

of the database is small, the exhaustive searching of the database by comparing the query structure against each and every structure in the database was done with acceptable performance. However, for large databases with tens of thousands of structures, such an exhaustive searching approach no longer provides a satisfactory response time. As such, extensive research has gone into developing faster searching algorithms[18].

Generally, measuring similarity and classifying proteins in a database require experts that have extensive knowledge of molecular biology domain. This is due to the fact that some measuring methods such as SCOP, DALI and FSSP, are usually based on a particular biological conception of structural similarity of proteins. These are based on sequence-alignment searches. Although they yield an accurate searching, they are very time consuming[13].

Thus, in recent, the geometric structure based model similarity calculation methods of protein have been developed. Some researchers show that approaching similarity of protein by its geometry is promising[2].

Unfortunately, we cannot currently reach the full potential of molecular shape comparison methods in these applications because of several shape comparison methods in these applications caused by several major problems like the optimal orientation problem for finding and capturing the shape adequately. Many methods using molecular shape as the pattern to recognize require previous alignment process of the molecules being compared, which is an additional source of difficulty that may lead to suboptimal molecular overlap and thus to an inaccurate similar score. Additionally, it is needed to understand how well molecular shapes are being described and thus compared. Especially, a small inaccuracy in the description of shape will lead to many similar compounds being undetected, given the very large size of interesting molecular databases.

The rest of this paper is organized as the following. In Section 2, we describe some related works done by previous researchers. In Section 3, we present our inter-atomic distance based shape descriptor and its implementation. In Section 4, we discuss the experimental result and some highlighted observations. Section 6 concludes the paper.

2. Molecular Shape Comparison

2.1 Orientation fix method

In general, these categorical methods take three steps. They are normalization, spatial partitions and comparing step. Several issues such as normalization, spatial partitions and geometric features were addressed by many researchers[1].

Normalization step is to determine the full normalized pose or semi-normalized pose of molecules. The semi-normalized pose is obtained by transforming them such that their center of mass is in the origin and scaling them to a certain unit of bounding box. The full-normalized pose is aimed to preserve the object invariant to translation, scaling, rotation and mirroring. Translation and scaling invariance are easily obtained by the same way as in the semi-normalized form.

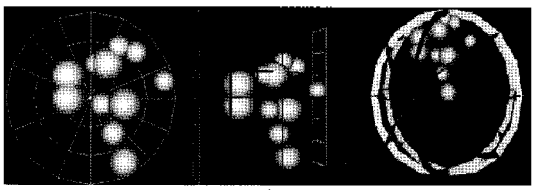


Fig. 1. Some spatial partitions (a) 3D-ecompositions, (b) 3D Grid, (c) Spherical Wedge.

However, rotational invariance is obtained by the following way. First employ Principal Component Analysis (PCA) to the objects in order to get the principal axis. Second, rotate them such that first major axis is adjacent to x-axis, the second axis to y-axis, and the third to z-axis respectively. Mirroring invariance is obtained by flipping the objects such that the larger part is on the positive side.

Spatial partition is to capture the shape features. Some spatial partitions are shown in Fig. 1. Fig. 1(a) shows the case of the 3D decomposition which enables to count the vertices of the 3D model in each spherical sector. Fig. 1(b) shows the 3D grid for counting the vertex of the 3D model. Fig. 1(c) shows the spherical wedge for counting vertices of the 3D model in each wedge like cone.

Ankerst suggested the method which extracts geometry of the molecules and maps into the 3D shape histogram, using three options of 3D decomposition techniques[3]. Each decomposition model is the ball (shell), sector and combination of both (spider-web). Since the cost of PCA is expensive and optimal orientation cannot be always guaranteed, the orientation dependent methods suffer from unintended loss for the ideal result.

2.2 Orientation free method

Many researchers suggested some methods which take the strategy by using the position of the mole-

cule which is orientation independent. As we will discuss in this section, Yeh *et al.* suggested the method with the Light Field Descriptor[17]. In this method, each model is rotated several times in order to obtain its projection image from some camera positions. Therefore, captured features are composed of several features of 2D shape and contour of projection images, i.e. which are called as Zenrike moments. After combining these features, they calculate the distance of features by using L_1 norm. This norm is called as the Manhattan distance which represents the distance between two vectors as shown in Eq. (1). The dissimilarity of two proteins is assessed by using this norm. Here, p and q are the feature vectors.

$$d_{Manhattan}(p,q) = (\sum_{i=0}^{n}|p_i - q_i|) \quad (1)$$

This method needs to generate an intermediate representation, i.e. 2D images of 3D protein, before extracting the features. It causes a hard calculation and the quality of the method depends on the number of camera positions for the projection. In case of the protein of many atoms, the projected image cannot easily guarantee the exact similarity assessment because they do not use exact distance between atoms[5].

Inter-atomic distance is defined as the Euclidean distance between two atoms. Inter-atomic distance based shape descriptor gives some advantages. At first, the distribution generated from inter-atomic distances reflects the shape of protein. In general, the shape of a molecule is uniquely determined by the relative position of its atoms[6,15]. In this way, the molecule is regarded as a group of atoms, instead of its more conventional treatment as a solid body. The relative position of atoms in the molecule is in turn completely determined by the set of all inter-atomic distances. This is a convenient representation, which directly eliminates any need for alignment or translation, as this set of distances is independent of molecular orientation or position. However, the set of all inter-atomic distances contains more information that is needed to describe the shape of the molecule accurately. This is because the values of these distances are heavily constrained by the forces that hold the atoms together and thus using less information would still provide us with the shape discrimination power necessary to distinguish between molecules. Nevertheless, it is not widely used in molecular shape comparison because the calculation of the set of all inter-atomic distances is heavy as shown in Fig. 2.

If the given molecule has n positions which represent the center of each atom, the number of all inter-atomic distance pair is $n\times(n-1)/2$. Hence, we need to develop an method for feature capturing with more simplified combination of inter-atomic distance pairs. Some researchers try to solve this problem using histogram composed of distance between each atom. However, histogram calculation can suffer from well-known limitations. Under the very large databases, it is difficult to find the bin size suitable for all molecules. It does not meet the requirement of relatively large storage and computing power[10].

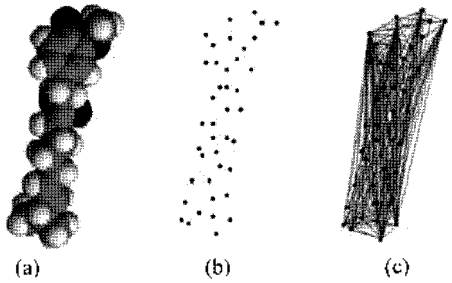


Fig. 2. Molecule, atom positions and all inter-atomic distance pairs (a) molecule, (b) atom positions, (c) all inter-atomic distance pairs.

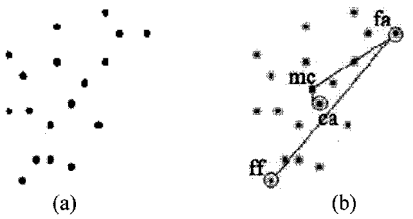


Fig. 3. Molecule. (a) the set of atoms, (b) four feature capturing locations and their relationship.

2.3 Ultrafast shape recognition method

To solve the problem mentioned above, Ballester and Richard adopted the distribution of the inter-atomic distance with only four feature capturing positions. This algorithm is called as USR (Ultrafast Shape Recognition). This method is very fast. In this method, they use the first, second and third moments of the distributions for all atomic distances in order to characterize them as a way to encode the molecular shape[5]. Each moment is calculated by Eq. (2), (3) and (4). In statistics, the first moments of the given data set is the mean which represents the central tendency as shown in Eq. (2).

The second moment is the variance which represents the dispersion as shown in Eq. (3). The third moment is the skewness which represents the asymmetric property of the given distribution as shown in

Eq. (4). Here, x_i is the data.

$$\bar{x} = \frac{1}{n}\sum_{i=0}^{n} x_i \tag{2}$$

$$S = \sum_{i=0}^{n}(x_i - \bar{x})^2 \tag{3}$$

$$b = \frac{1}{(n-1)(n-2)}\sum_{i=0}^{n}\frac{(x_i-\bar{x})^3}{s} \tag{4}$$

It is supported by a theorem from statistics, which prove that a distribution is completely determined by its moments[11,14]. They suggest very fast, and efficient method using four types of inter-atomic distance based distributions and their moments. In their method, they define the set of all inter-atomic distances from four molecular locations, the centroid of all atoms (*mc*), the closest atom from *mc* (*ca*), the farthest atom from *mc* (*fa*) and the farthest atom from *fa* (*ff*) are the locations respectively. These locations represent the center of the molecule and its extremes, and thus are well separated as shown in Fig. 3.

Four feature capturing locations are defined as follows:

Definition 1 : *mc*, The center location of all atoms of given protein *p*.

Definition 2 : *ca*, The location of the atom with the minimum distance from *mc*.

Definition 3 : *fa*, The location of the atom with the maximum distance from *mc*.

Definition 4 : *ff*, The location of the atom with the maximum distance from *fa*.

They try to capture the distribution of inter-atomic distance from the four locations to other atoms respectively. Fig. 4 shows the process of generating the distribution.

Four distributions are as follows.

1. $\{d^{mc}\}_{k=1}^{N}$ corresponds to the Fig. 4(a)
2. $\{d^{ca}\}_{k=1}^{N}$ corresponds to the Fig. 4(b)
3. $\{d^{fa}\}_{k=1}^{N}$ corresponds to the Fig. 4(c)
4. $\{d^{ff}\}_{k=1}^{N}$ corresponds to the Fig. 4(d)

where N is the number of atom, d^{mc}, d^{ca}, d^{fa} and d^{ff} are the Euclidean distances from *mc*, *ca*, *fa* and *ff* to each atom, as shown in Eq. (5). Here, p and q are the feature vectors.

$$d_{Euclidean}(p,q) = \left(\sum_{i=0}^{n}|p_i - q_i|^2\right)^{1/2} \tag{5}$$

By using these four distributions, 12 shape descriptors are defined.

First descriptor (μ_1^{mc}) corresponds to the first moment of the distribution $\{d_k^{mc}\}_{k=1}^{N}$. This value means the mean atomic distance to the geometrical center and thus it provides an estimate of the size of the molecules.

Second descriptor (μ_2^{mc}) corresponds to the second moment of $\{d_k^{mc}\}_{k=1}^{N}$. This value is the variance of these atomic distances from the centroid and hence it is related to how compact the molecule is.

Third descriptor (μ_3^{mc}) corresponds to the third moment of $\{d_k^{mc}\}_{k=1}^{N}$. This value is the skewness of the same distribution, which estimates its asymmetry and thus whether the atoms are near or far from the mean atomic position from the centroid.

Similarly, $(\mu_1^{ca}), (\mu_2^{ca}), (\mu_3^{ca}), (\mu_1^{fa}), (\mu_2^{fa}), (\mu_3^{fa})$, $(\mu_1^{ff}), (\mu_2^{ff})$, and (μ_3^{ff}) are calculated from the distributions of $\{d_k^{ca}\}_{k=1}^{N}$, $\{d_k^{fa}\}_{k=1}^{N}$ and $\{d_k^{ff}\}_{k=1}^{N}$.

After all 12 shape descriptors in a given molecule are calculated, they are assembled in an associated vector which is uniquely defined as follows.

$$\vec{M}^q = (\mu_1^{mc}, \mu_2^{mc}, \mu_3^{mc}, \mu_1^{ca}, \mu_2^{ca}, \mu_3^{ca}, \mu_1^{fa}, \mu_2^{fa}, \mu_3^{fa}, \mu_1^{ff}, \mu_2^{ff}, \mu_3^{ff})$$

Here, $\vec{M}^q$ spans a 12-dimensional molecular shape space. Now, it enables the similarity calculation between a given query molecule and each protein in a database. In this process, the normalized score function is used to quantify the degree of similarity between molecules. These values are assessed by using the inverse function of the distance between two vectors composed of twelve float values calculated from each molecule.

The inverse function of the translated and scaled

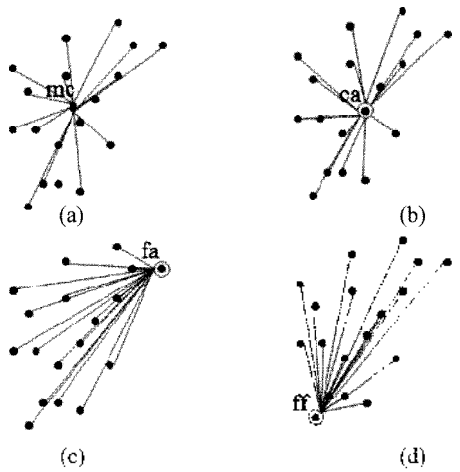


Fig. 4. The four distributions of inter-atomic distance from four feature capturing locations to other atoms.

Manhattan distance between both vectors of shape descriptors can guarantee the stable speed. This function returns a value in the range of (0, 1]. Here, the value '1' corresponds to maximum similarity and the value near '0' corresponds to near minimum similarity as shown in Eq. (6).

$$S_{qt} = \frac{1}{1 + \frac{1}{12}\sum_{i=1}^{12}\left|M_i^q - M_i^t\right|} \in (0, 1] \quad (6)$$

Here, q and t represent query molecule and each molecule being compared in the database respectively. This method is reported to be extremely fast. The main reason for such efficiency is that the defined shape descriptors only require the calculation of $4N$ distances along with a total of 12 moments of the resulting four distributions. Once, $\vec{M}^q$ is calculated, the comparison calculation cost takes $O(N)$. Since the shape information of each molecule is independently encoded a vector of shape descriptor, which is consistent with the status as an intrinsic geometrical property of the molecule, they can speed up the screening process, as cross-calculations between the query and the considered molecules are avoided. This method is faster than other methods which use the calculation of molecular surface or molecular volume.

3. Proposed Algorithm

3.1 Problem definition

In Ballester and Richard's method, we observed some limitations. First, their method using four feature capturing positions does not always guarantee the desirable matching. Second, the method can compare only representative feature vector in terms of moments of four inter-atomic distance distributions. Thus it cannot give any scale information between arbitrary two feature vectors suggested by Ballester and Richard[1].

In this paper, we focus an second problem. Generally, the molecular shape search system presents the similarity result as well as the inter-relationship in overall DB index structure between two molecules. It is important to speed up in the finer comparing process after screening process in molecular database. Therefore, our problem definition can be defined as follows:

Given the query protein q, and the target protein t, what is the effective shape descriptor for calculating the similarity measure between q and t in view of implementation time and the accuracy. What is the natural extension of USR method in view of the database index structure?

Here, q and t are point clouds in format of PDB (Protein Data Bank). The point cloud is the collection of the center for each atom.

$q = \{(x_1, y_1, z_1), (x_2, y_2, z_2), \ldots, (x_m, y_m, z_m)\}$.
$t = \{(x_1, y_1, z_1), (x_2, y_2, z_2), \ldots, (x_n, y_n, z_n)\}$.

Here, m and n are the numbers of atoms in q and t.

We use the moment vector method like Ballester and Richard's method. First, we get the four distributions using four feature capturing positions in section 2.3.

3.2 3D shape descriptor for protein with natural extension of USR method

3.2.1 3D shape descriptor

For the purpose of acquiring the natural extension of USR method in view of database index structures, we analyze the USR generation process. USR feature vector generation steps are as follows.

Procedure 1 USR feature vector generation step
Input : PDB file with x, y, z coordinate which represents the center point of each atom
output : 12 moments of four inter-atomic distances with four feature capturing positions
Step 1 Find *mc* from *xyz* using average (x, y, z)
Step 2 Find *ca* using min distance between *mc* and other atoms
Step 3 Find *fa* using max distance between *mc* and other atoms
Step 4 Find *ff* using max distance between *fa* and other atoms
Step 5 $\{d^{mc}\}$ $\{d^{ca}\}$ $\{d^{fa}\}$, and $\{d^{ff}\}$ are generated
Step 6 Calculate the moments 1st (AVG), 2nd (VAR), and 3rd (SKEW)
Step 7 Return the feature vector ($m1$, $m2$, ... $m12$)

Above steps are naturally extended into the below procedure without the loss of generality.

Procedure 2 Proposed 'M3D' feature vector generation step
Input : PDB file with x, y, z coordinate which means the center of each atom
output : n, $r1$ and $r2$, 12 moments of four inter-atomic distance with four feature capturing positions
Step 1 Find *mc* from *xyz* using average (x, y, z)
n is calculated after step 1 is done.
Step 2 Find *ca* using min distance between *mc* and other atoms. Here, this max distance is defined as $r1$. This value means the radius of the inscribed sphere in R^3.
Step 3 Find *fa* using max distance between *mc* and

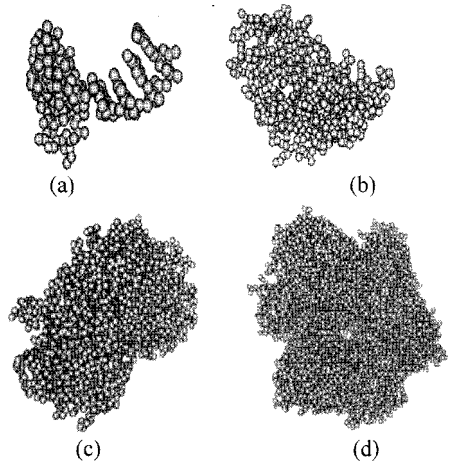


Fig. 5. The effect of n, the number of atoms (a) 116D.pdb (b) Insulin.pdb, (c) 1AA7.pdb, (d) 1CJD.pdb.

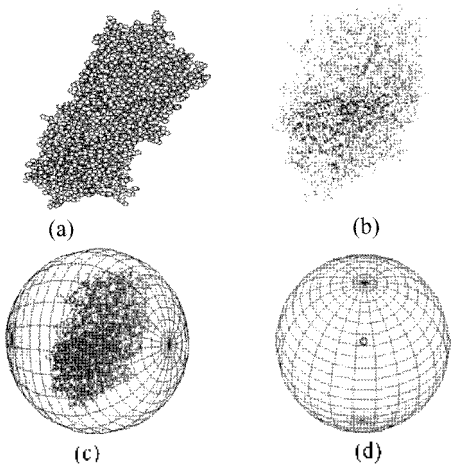


Fig. 6. The $r1$ and $r2$ as their geometric view and shape descriptor (a) protein 2LAL, (b) bounding sphere with $r1$, (c) bounding sphere with $r2$, (d) bounding spheres with $r1$ and $r2$.

other atoms. Here, this max distance is defined as $r2$. This value means the radius $r2$ is the circumscribed sphere in R^3.

Step 4 Find *ff* using max distance between *fa* and other atoms.

Step 5 $\{d^{mc}\}$ $\{d^{ca}\}$ $\{d^{fa}\}$, and $\{d^{ff}\}$ are generated

Step 6 Calculate each moments 1st (AVG), 2nd (VAR), and 3rd (SKEW)

Step 7 Return the feature vector (n, $r1$, $r2$, $m1$, $m2$,..., $m12$).

3.2.2 Two speedup factors

In the procedure 2, we can find the interesting properties. First, we can easily acquire n, the number of atoms. This value can be used to index structure for retrieving the molecular database as a starting point as shown in Fig. 5.

Second, we can compute $r1$ and $r2$ easily, the radii of bounding spheres respectively as shown in Fig. 6. Using $r1$ and $r2$, we can deal with smaller sets with similar proteins, Each term shows the differentiation power. First, our shape descriptor represents the important geometrical structure in terms of $r1$ and $r2$.

In Fig. 6, $r1$ means the distance between *mc* and *ca*. This information shows the given protein's inside structure as shown in Fig. 6(d). If this value is big, then the given protein is likely to have a hollowed part as shown in Fig. 7(a).

In Fig. 6, $r2$ means the distance between *mc* and *fa* as shown in Fig. 6(c). This implies the volume size of the given protein. In USR, since they only compare the molecules by using 1st, 2nd and 3rd moments of four distributions with four feature capturing positions, the original shape's geometry and topological information are unknown.

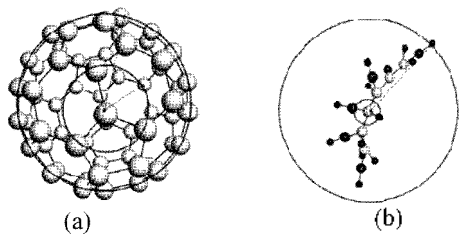


Fig. 7. The effect of $r1$ and $r2$ (a) Uckminsterfullerene, (b) Fructose.

However, if the number of atoms of two proteins is considerably different, we think two shapes are considerably different structure and it is reasonable.

Finally, our feature vector from the proposed 3D molecular shape descriptor is as follows.

$M3D = \{ n,$ // number of atoms
$r1,$ // ***dist*** (*mc*, *ca*)
$r2,$ // ***dist*** (*mc*, *fa*)
$m1, m4, m7, m10,$ // 1st moments of K
$m2, m4, m8, m11,$ // 2nd moments of K
$m3, m6, m9, m12,$ // 3rd moments of K
$\}$

Here, n is acquired by accounting the atom number from PDB file. K means four distributions, $\{\{d^{mc}\}$ $\{d^{ca}\}$ $\{d^{fa}\}$, $\{d^{ff}\}$ $\}$. $r1$ and $r2$ is made up of ***dist*** (*mc*, *ca*) and ***dist*** (*mc*, *fa*) is calculated by Eq. (5) in respectively. These values are taken from Procedure 2.

3.3 Similarity measure

Our similarity measure is composed of three parts. They are the number of atoms in the protein, n, radii of two bounding sphere $r1$ and $r2$, and the original USR, $m1$, ..., $m12$ respectively. These three parts have each role for screening the molecular database.

3.3.1 The number of atoms

Quite different number of atoms in protein reflects quite different shape of the protein as shown in Fig. 5.

It is somewhat vague measure in view of exact shape matching. However, this measure filters the potentially unmatched proteins. If a new entry protein p has n number of atoms, then our algorithm start to retrieve the exact number of atoms, n. As shown in Fig. 8, it lessens the problem space by skipping comparisons between proteins with a huge difference in the number of atoms. The criteria for the number of atoms are user specific or statistically assigned value by using Eq. (7).

$$S_1 = \begin{cases} \log_{10}|n_2 - n_1| > k \\ \text{otherwise goto } S_2 \end{cases} \quad (7)$$

Here, n_1 and n_2 are the number of query protein q and target proteins t_i. If the value of S_1 is larger than k (=3.0), then the number of atoms between two proteins is larger than 1,000, where k is the user assigned value.

This rule with S_1 can reduce the size of protein screening because of the controlling of the user defined value, k. For the determination of the value k, we test the rule 1 with various k under test set which is shown in table 1. In the case of k=100, k=1,000 and k=10,000, we can get 14, 102 and 308 proteins as the similar proteins under given query proteins. Although, we don't take the optimal k, we took k=1,000 in practical view.

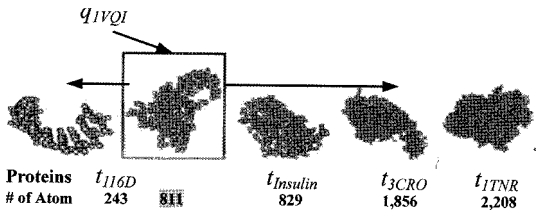


Fig. 8. n_1 as a starting point of the screening process.

3.3.2 Two bounding spheres

After assessing n as the starting point of screening process with Eq. (7), we calculate the Eq. (8) which is the similarity measure of simplified volumetric

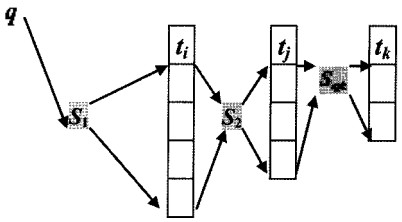


Fig. 9. The focused set generation using n with S_1 and shape filtering with S_2 and exact shape matching with S_{qt}.

structure which is formulated with $r1$ and $r2$.

$$S_2 = \frac{1}{1 + \frac{1}{2}\sum_{l=1}^{2}\left|r_l^q - r_l^q\right|} \in (0, 1] \quad (8)$$

Here, the similarity S_2 is the measurement which reflects the bounding sphere with $r1$ and bounding sphere with $r2$ as described in section 3.2.2. It can be decomposed form about $r1$ and $r2$ in respectively. Then, we can measure the similarity effects in $r1$ and $r2$ by Eq. (9) and Eq. (10) as follows:

$$S_{2\_r1} = \frac{1}{1 + \left|r_1^p - r_1^q\right|} \in (0, 1] \quad (9)$$

$$S_{2\_r2} = \frac{1}{1 + \left|r_2^p - r_2^q\right|} \in (0, 1] \quad (10)$$

In section 4.2, we show the similarity value pairs $<S_{2\_r1}, S_{qt}>$, $<S_{2\_r2}, S_{qt}>$ and $<S_2, S_{qt}>$ in Table 3 and Fig. 11. Here, S_{qt} is the similarity measure which represents 12 moments, $<m1, \ldots m12>$ by using Eq. (6).

3.3.3 Shape filtering with S_1 and S_2

After two steps, the search space to proceed is extremely shrinked as shown in Fig. 9 because of the effect of two similarity measures as shown in Eq. (7) and Eq. (8).

3.3.4 Shape match using S_{qt}

Then, we finally apply USR's 12 moments based similarity function as shown in Eq. (6). It takes a similarity calculation process between a given query molecule and each protein in a database. It uses the normalized score function to quantify the degree of similarity between molecules like USR.

Thus, they use the inverse of the translated and scaled Manhattan distance between both vectors of shape descriptors, where a value of 1 corresponds to maximum similarity and 0 to minimum similarity as shown in Eq. (6).

4. Experimental Results

4.1 Experimental data

Table 1 shows the experimental data. These files were downloaded from RCSB (Research Collaboratory for Structural Bioinformatics) Protein Data Bank (www.pdb.org) and Fig. 10 shows their PDB files. These models were rendered by using the OpenGL. Our shape descriptor and similarity calculation method were developed under the Microsoft visual studio 2005 and C · · language.

Table 1. Input PDB files to the proposed algorithm

PDB file name	# of atoms	File size (KB)
116D.pdb	243	44
1VQI.pdb	811	115
insulin.pdb	829	160
1A8G.pdb	1,527	174
3CRO.pdb	1,856	214
1TNR.pdb	2,208	224
2BBM.pdb	2,700	258
1AHS.pdb	2,842	278
1AAW.pdb	3,069	285
1AA7.pdb	2,980	288
1B9T.pdb	3,040	300
1A34.pdb	2,945	325
1A3R.pdb	3,455	329
2LAL.pdb	3,550	332
1A6C.pdb	4,016	366
1B44.pdb	4,186	378
1AYM.pdb	6,412	618
1CJD.pdb	8,502	763
crystall.pdb	23,978	1,593
fluid.pdb	26,849	2,019

4.2 Feature vector as shape descriptor of each proteins

Table 2 shows the converted feature vector generated from our method. Each protein in the given database is evaluated by using the similarity measures composed of these vectors. As described in section 3, three similarity measures were used.

To validate the proposed similarity measures, at first, we test the pair-wise comparison by using n, $r1$-$r2$ and $m1..m12$ values.

We test the two similarity results, which are results by using the $r1$ and $r2$ based measure and by using USR measure. To acquire the statistical significance, we use the Pearson product-moment correlation coefficient r which is a common measure of the correlation (linear dependence) between two variables X and Y as shown in Eq. (8). Here, we set X as the similarity result by using Eq. (5) and Y as the similarity result by using Eq. (6). It means the similarity between two given array sets which are composed of each similarity values.

$$r = \frac{n(\Sigma XY) - (\Sigma X)(\Sigma Y)}{\sqrt{(n\Sigma X^2 - (\Sigma X)^2)[n\Sigma Y^2 - (\Sigma Y)^2]}} \in [-1, 1] \quad (9)$$

The Pearson's r reflects only numerical values of the given two data sets. In general, it is widely used to measure the pair-wise similarity of two data sets [16].

We can get the results with Pearson's r as shown in Table 3. Each three measures are evaluated in Table 3. Here, <$r1$ &$m12$> means the pearson's correlation coefficient r in given two sets from $r1$'s similarity in Eq. (9) and $m12$'s similarity S_{qi} in Eq. (6). The average of <$r1$&$m12$> marked 53.7%, the average of <$r2$&$m12$> and <$r1$, $r2$&$m12$> marked 88.6% and 92.9% in respectively.

These results provide the strong correlation between result of Eq. (6) and Eq. (8). It can be analyzed that our <$r1$, $r2$> based shape descriptor by using the similarity measure, S_2 as shown in Eq. (8) in the section 3.3.2 is well adjusted in shape filtering before S_{qi} process. It can be also validated to draw

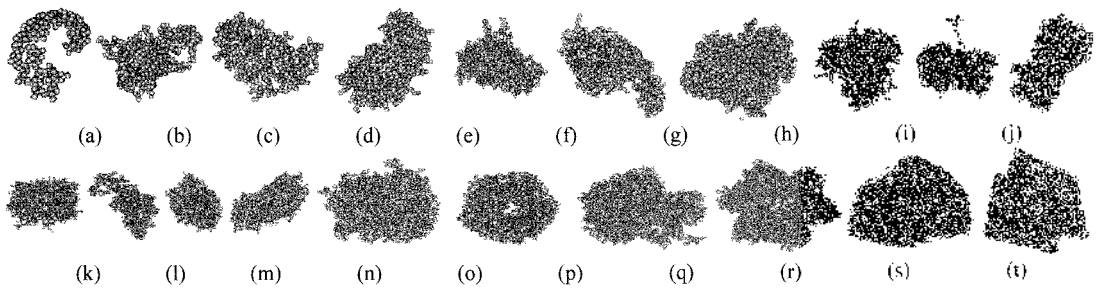


Fig. 10. PDB models. Each PDB file name is described in Table 1 and is sorted by using # of atoms.

Table 2. The resulting feature vectors as name of M3D descriptor by using the proposed algorithm

M3D Descriptor	n	$r1$	$r2$	$m1$	$m2$	$m3$	$m4$	$m5$	$m6$	$m7$	$m8$	$m9$	$m10$	$m11$	$m12$
116D_usr_dgu.txt	243	4.0	18.9	11.6	10.4	0.2	11.8	21.1	-0.1	21.0	60.9	-0.8	20.4	77.7	-0.6
1VQI_usr_dgu.txt	811	1.7	27.2	13.6	26.4	0.4	13.6	29.4	0.5	29.2	97.6	-0.4	28.5	90.9	-0.2
insulin_usr_dgu.txt	829	2.1	22.8	12.9	19.9	0.0	13.1	20.9	0.0	25.0	81.8	-0.2	24.3	98.5	-0.1
1A8G_usr_dgu.txt	1,527	1.5	33.4	16.6	30.3	-0.1	16.6	31.6	0.0	35.6	154.8	-0.2	33.6	147.1	-0.2
3CRO_usr_dgu.txt	1,856	3.2	35.3	18.4	44.1	0.1	18.6	47.6	0.2	37.8	206.8	-0.1	35.4	205.7	0.1
1TNR_usr_dgu.txt	2,208	1.1	51.1	22.9	91.5	0.4	22.9	93.2	0.5	53.2	392.2	-0.6	38.5	311.7	0.4
2BBM_usr_dgu.txt	2,700	1.0	32.0	16.2	28.7	0.0	16.2	28.8	0.0	34.8	101.2	-0.3	30.3	113.4	-0.2
1AHS_usr_dgu.txt	2,841	4.6	32.8	19.1	31.3	-0.2	19.6	34.4	-0.2	36.7	120.7	-0.3	34.1	129.5	-0.4
1A34_usr_dgu.txt	2,945	0.5	44.2	19.1	72.7	0.6	19.2	72.4	0.6	45.9	283.0	-0.5	39.5	276.8	0.2
1AA7_usr_dgu.txt	2,980	1.9	34.9	19.5	42.7	-0.3	19.5	44.0	-0.2	37.8	205.5	0.0	35.3	202.2	0.0
1B9T_usr_dgu.txt	3,040	1.7	37.0	18.8	30.3	-0.3	18.9	30.8	-0.3	40.1	144.0	-0.6	30.8	115.4	0.1
1AAW_usr_dgu.txt	3,069	1.8	41.8	20.6	45.0	0.0	20.6	46.3	0.0	46.1	91.5	-1.2	32.6	109.4	0.0
1A3R_usr_dgu.txt	3,455	3.4	43.5	24.1	55.0	0.0	24.3	58.2	0.0	46.3	381.8	-0.2	44.4	378.4	0.1
2LAL_usr_dgu.txt	3,550	2.1	44.8	23.5	71.6	0.0	23.7	72.0	-0.1	41.0	388.2	0.1	47.2	285.2	0.0
1A6C_usr_dgu.txt	4,016	1.9	55.3	25.0	91.4	0.2	25.1	92.2	0.2	58.1	398.4	-0.5	52.5	296.4	0.3
1B44_usr_dgu.txt	4,186	6.4	36.2	22.4	36.1	-0.4	23.0	49.6	-0.2	40.7	189.2	-0.3	36.9	172.0	-0.2
1AYM_usr_dgu.txt	6,412	1.7	226.3	30.8	710.7	5.9	30.9	711.6	5.9	227.6	1072.3	-4.7	51.5	1088.2	4.2
1CJD_usr_dgu.txt	8,502	6.2	47.5	29.0	60.2	-0.5	29.4	72.5	-0.4	54.1	235.8	-0.3	53.1	237.5	-0.5
crystall_usr_dgu.txt	3,978	2.5	60.8	39.5	99.9	-0.7	39.5	101.9	-0.7	69.9	469.8	-0.4	68.8	488.7	-0.4
fluid_usr_dgu.txt	6,849	1.9	67.2	38.7	125.6	-0.2	38.8	126.4	-0.2	74.8	549.6	-0.3	72.3	550.7	-0.3

Table 3. The result of the Pearson's r for calculating correlations between the similarity results calculated by using Eq. (6) and Eq. (8)

Protein ID	Pearson's r		
	<r1&m12>	<r2&m12>	<r1,r2&m12>
116D	0.800	0.987	0.975
1VQI	0.366	0.981	0.955
Insu	0.393	0.982	0.964
1A8G	0.436	0.779	0.973
3CRO	0.766	0.775	0.864
1TNR	0.541	0.975	0.966
2BBM	0.577	0.843	0.976
1AHS	0.865	0.775	0.912
1A34	0.714	0.764	0.878
1AA7	0.367	0.790	0.658
1B9T	0.373	0.843	0.881
1AAW	0.345	0.921	0.904
1A3R	0.706	0.806	0.988
2LAL	0.398	0.809	0.863
1A6C	0.333	0.985	0.967
1B44	0.768	0.773	0.962
1AYM	0.320	1.000	1.000
1CJD	0.755	0.944	0.972
Crys	0.601	0.987	0.943
Flui	0.313	0.993	0.982
Average	0.537	0.886	0.929

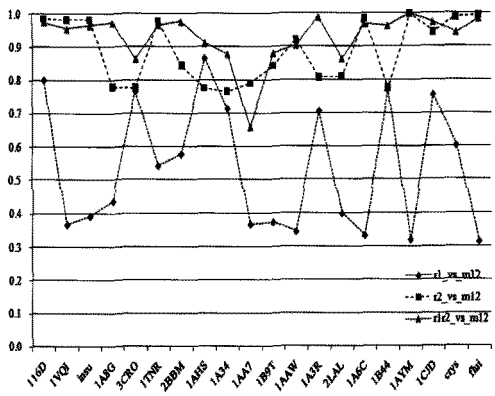


Fig. 11. Three correlation results which represent relationships of each similarity pairs, <$r1$ and $m12$>, <$r2$ and $m12$> and <$r1r2$ and $m12$>.

three correlation results which represent relationships of each similarity pairs as shown in Fig. 11. In this figure, the <$r1$, $m12$> means the correlation between the similarity by using $r1$ and $m12$. Similarly, <$r2$, $m12$> and <$r1r2$, $m12$> mean the correlation between the similarity by using $r2$, $m12$, and $r1r2$, $m12$ in respectively. Although $r1$ itself has somewhat weak correlation with $m12$, the combined metric $r1r2$ by adding the measure $r2$ has strong correlation with $m12$. Thus, it supports that our new measure $r1r2$ for shape filtering is adjusted to its filtering

purpose.

Therefore, we conclude the shape descriptor S_2 is an effective shape descriptor for screening in the geometrical and computational viewpoint. The proposed two bounding spheres effectively capture the shape of proteins and extract the useful information with less storage. In this experiment, all PDB files are converted into the feature vectors with over 130 bytes by using our M3D shape descriptor.

5. Conclusion

We present a new 3D shape descriptor for screening the molecular database which is based on the moments of distributions for the inter-atomic distances. Unlike the previous method, USR, our method has the expanded features for filtering by using Eq. (7) and Eq. (8). We can get the value 'n', the number of atoms from PDB preprocessing. This value can give the natural index for retrieving the records in DB. In this paper, we add the geometrical structure as terms of $r1$ and $r2$ by using Eq. (8) in our shape descriptor to empower the geometric interpretation for shape differentiation power. As shown in section 4, the proposed method is very fast and remarks the considerable shape differentiation power like USR. Our method is useful in the pre-processing for the exact shape comparison. Although our algorithm gives us fast and considerable result, we do not find the optimal configuration of the feature capturing positions due to tradeoffs between the accuracy and speed. This is one of the future works.

Acknowledgements

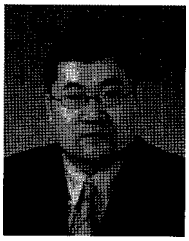
This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD, Basic Research Promotion Fund) (KRF-2007-314-D00311), the Korea Research Foundation Grant funded by the Korean Government (MOEHRD) (No. KRF-2005-041-D00903), and the Basic Research Program of the Korea Science & Engineering Foundation (No. R01-2006-000-10327-0).

References

1. Akbar, S., Kung, J. and Wagner, R., "Exploiting Geometrical Properties on Protein Similarity Search", In 17th Proceedings on International Conference on Database and Expert Systems Applications (DEXA '06), pp. 228-234, 2006.
2. Ankerst, M., Kastenmüller G., Kriegel H.-P. and Seidl T., "Nearest Neighbor Classification in 3D Protein Databases", In Proceedings of 7th International Conference on Intelligent Systems for Molecular Biology, pp. 34-43, 1999.
3. Ankerst M., Kastenmüller G., Kriegel H.-P. and Seidl T., "3D Shape Histograms for Similarity Search and Classification in Spatial Databases", *Lecture Notes in Computer Science*, Vol. 1651, pp. 207-226, 1999.
4. Aung, Z., Fu. W. and Tan, K.L., "An Efficient Index-based Protein Structure Database Searching Method", In Proceedings of 8th International Conference on Database System for Advanced Applications (DASFAA'03), pp. 311-318, 2003.
5. Ballester, P. J. and Richard, W. G., "Ultrafast Shape Recognition to Search Compound Databases for Similar Molecular Shapes", *Journal of Computational Chemistry*, Vol. 28, pp. 1711-1723, 2007.
6. Bemis, G. W. and Kuntz, I. D. "A Fast and Efficient Method for 2D and 3D Molecular Shape Description", *Journal of Computer Aided Molecular Design*, Vol. 6, pp. 607-628, 1992.
7. Berman, H. M., *et al*, "The Protein Data Bank", *Nucleic Acid Res.*, Vol. 28, pp. 235-242, 2000.
8. Bertino, E., *et al*., "The Astral Compendium for Sequence and Structure Analysis", *Nucleic Acids Res.*, Vol. 28, pp. 254-256, 2000.
9. Böhm, H.-J., Flohr, A. and Stahl, M. "Scaffold Hopping", *Drug Discovery Today: Technology*, Vol. 1, pp. 217-224, 2004.
10. Good, A. C. and Richards, W. G. ,"Explicit Calculation of 3D Molecular Similarity", *Perspective Drug Discovery Design*, Vol. 9, pp. 321-338, 1998.
11. Hall, P., "A Distribution is Completely Determined by Its Translated Moments", *Probability Theory and Related Fields*, Vol. 62, pp. 355-359, 1983.
12. Jenkins, J. L., Glick, M. and Davies, J. W., "A 3D Similarity Method for Scaffold Hopping from Known Drugs or Natural Ligands to New Chemotypes", *Journal of Medical Chemistry*, Vol. 47, pp. 6144-6159, 2004.
13. Kransnogor, N. and Pelta, D.A., "Measuring the Similarity of Protein Structures by Means of the Universal Similarity Matrix", *Bioinformatics*, Vol. 20, pp. 1015-1021, 2007.
14. Matter, L., "Completeness of Location Families, Translated Moments, and Uniqueness of Charges", Probability Theory and Related Fields, Vol. 62, pp. 137-149, 1985.
15. Nilakantan, R., Bauman, N. and Venkataraghavan, R., "New Method for Rapid Characterisation of Molecular Shapes: Applications in Drug Design", *Journal of Chemical Information Computer Science*, Vol. 33, pp. 79-85, 1993.
16. Rodgers, J. L. and Nicewander, W. A., "Thirteen Ways to Look at the Correlation Coefficient", *The American Statistician*, Vol. 42, No. 1, pp. 59-66, 1988.

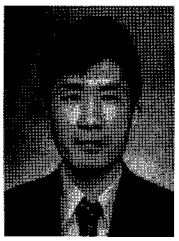
17. Yeh, J.-S. *et al.*, "A Web-based Three Dimensional Protein Retrieval System by Matching Visual Similarity", *Bioinformatics Applications Note*, Vol. 21, pp. 3056-3057, 2005.

18. 김동욱, 조영송, 김덕수, "삼차원 구의 보로노이 다이어그램 계산을 위한 두 가지 알고리듬 및 단백질 구조해석에의 응용", 한국CAD/CAM학회 논문집, Vol. 11, No. 2, pp. 97-106, 2006.



이 재 호

1997년 한성대학교 산업공학과 학사
1999년 동국대학교 산업공학과 석사
2007년 동국대학교 산업공학과 박사
관심분야: Rapid Prototyping, 3D Shape Search, Protein Screening



박 준 영

1982년 한양대학교 산업공학과 학사
1985년 University of Minnesota 산업공학과 석사
1991년 University of Michigan 산업공학과 박사
1995년~현재 동국대학교 산업시스템공학과 교수
관심분야: Geometric Modeling, Rapid Prototyping, Mass-customization, Haptic rendering