한국CAD/CAM학회 논문집
Transactions of the Society of CAD/CAM Engineers

Vol. 20, No. 4, pp. 321-327. December 2015
DOI http://dx.doi.org/10.7315/CADCAM.2015.321

〈학술논문〉

pISSN 1226-0606
eISSN 2288-6036

원자개수와 경계구에 기반한 유사 단백질 스크리닝을 위한 검색 가속 기법

이재호[1] · 박준영[2†]

[1](주)유플러스네트웍스, [2]동국대학교 산업시스템공학과

Atom Number and Bounding Sphere Based Search Speedup Technique for Similar Proteins Screening

Jaeho Lee[1] and JoonYoung Park[2†]

[1]Uplusnetworks Co. Ltd.
[2]Department of Industrial and System Engineering, Dongguk Univ.

Received 3 April 2015; received in revised form 28 May 2015; accepted 1 June 2015

ABSTRACT

In the protein database search, 3D structural shape comparison for protein screening plays a important role. Protein databases have big size and have been grown rapidly. Exhaustive search methods cannot provide a satisfactory performance. As protein is composed of a set of spheres, the similarity calculation of two set of spheres is very expensive. Thus, a reasonable filtering method could be an answer for the speedup of protein screening. In this paper, we suggest a speedup method for protein screening with atom number and bounding sphere. We also show some experimental results for the validity of our method.

Key Words: Atom number, Bounding sphere, Protein screening, Ultrafast shape recognition, Shape based search

1. 서 론

생물정보학에서 분자의 3차원 구조는 매우 중요하다. 일반적으로 분자의 3차원 구조는 해당 분자의 기능 구현과 밀접한 관계가 있다고 알려져 있다[1]. 특히, 신약 개발시, 새롭게 합성된 분자는 그 기능을 분석하기 위해서 그것과 가장 유사한 분자들을 찾아서 그것들의 특성들과 대조하는 작업을 거치게 된다. 이를 가상 스크리닝(virtual screening)이라고 한다[3].

이러한 가상 스크리닝을 위해서는 주어진 분자와 3차원 형상적으로 가장 유사한 것을 찾아내는 작업을 수반한다. 분자는 원자들의 집합으로 구성되어 있다. 원자는 반지름 r을 가진 구로 형상화된다. 분자를 3차원 형상적으로 비교한다는 것은 구들의 집합을 비교하는 것과 같다. 일반적으로 3차원 형상의 유사도를 효율적으로 계산하는 것은 어려운 것으로 알려져 있다. 그러나 Fig. 1과 같이 이를 위한 여러 방법들이 제안되었다[5,7].

3차원 형상에 대한 다양한 뷰포인트로부터의 투영을 통한 2차원 이미지들로의 변환을 이용하여

†Corresponding Author, jypark@dgu.edu

주어진 3차원 형상간 비교를 2차원 이미지들의 비교로 변환하는 방법이 제안되었다[5,11]. 또한 주어진 3차원 형상을 구성하는 원자들에 대해서 3차원 공간상의 특정 구역내에 포획되는 원자들의 개수로 구성되는 형상 히스토그램을 형성하여 주어진 3차원 형상간의 비교를 2차원 히스토그램의 비교로 변환하는 방법이 제안되었다[2].

2007년에 영국의 Ballester와 Richard는 USR (Ultrafast Shape Recognition)이라는 매우 빠르고 정밀한 방법을 제안했다[4]. Fig. 1에서 보는 바와 같이 USR은 속도와 정보 압축도 면에서 가장 우수한 것으로 평가되고 있다. 이들은 공간상의 4군데에 거리 측정을 위한 기준점을 설정한 뒤 그 기준점으로부터 나머지 원자들까지의 거리(interatomic distance)를 측정한 분포들을 형성하였다. 수학적으로 주어진 분포는 완전히 그 모멘트들로부터 결정된다[8]. 따라서 이들은 해당 거리 분포들로부터 1차, 2차, 3차 모멘트를 추출하였다. 이렇게 획득된 12개의 형상 벡터는 기존의 방법들보다 훨씬 빠르고 정확한 형상 비교능력을 제공하였다. 그러나 분자 데이터베이스는 매우 크기 때문에(30억개 이상의 화합물) 발레스터의 방법으로도 꽤 많은 시간을 소모하는 것이 현실이다[4,10]. 이에 본 연구에서는 거대 분자에 해당하고 인체내에서 약리적인 효과가 커서 최근 신약개발의 중심에 있는 단백질 데이터베이스를 기반으로 한 검색 가속화 방안을 제안하고자 한다.

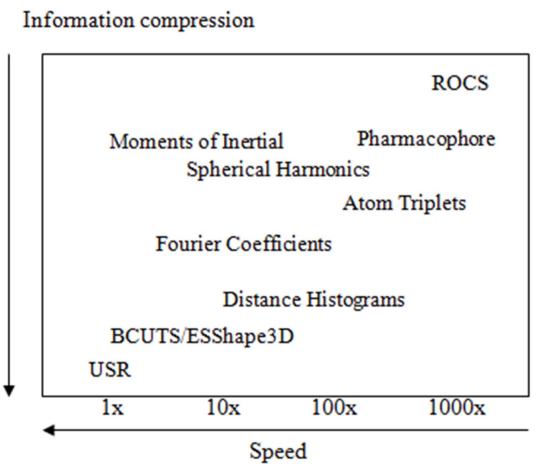

Fig. 1 Various 3D shape descriptors, their speed and information compression ratio

2. 유사한 단백질들의 스크리닝

2.1 형상 기술자

본 연구에서는 분자 기술자로써 속도와 정밀도가 검증되어 있는 USR의 확장 버전 *M3D*를 사용하고자 한다. *M3D* 포맷과 단백질 파일로부터의 생성방법은 본 연구의 선행연구에서 제안된 바 있다[9]. *M3D*는 아래와 같다.

$$M3D = \{n, r_1, r_2, m_1, \ldots, m_{12}, \text{PDB\_ID}\}$$

여기서, $\{m_1, \ldots, m_{12}\}$는 USR 알고리즘의 형상기술자이다. n은 단백질을 구성하고 있는 원자의 개수이다. r_1은 분자의 중심에서부터 가장 가까운 원자의 중심까지의 거리이고 r_2는 분자의 중심에서부터 가장 먼 원자의 중심까지의 거리이다. m_1, m_2, m_3는 분자의 중심(MC)에서부터 나머지 원자들의 중심까지의 거리로부터 얻은 거리 분포의 1차, 2차, 3차 모멘트들이다. m_4, m_5, m_6는 분자의 중심에서 가장 가까운 원자의 중심(CA)에서부터 나머지 원자들의 중심까지의 거리로부터 얻은 거리 분포의 1차, 2차, 3차 모멘트들이다. m_7, m_8, m_9는 분자의 중심에서 가장 먼 원자의 중심(FA)까지의 거리로부터 얻은 거리 분포의 1차, 2차, 3차 모멘트들이다. m_{10}, m_{11}, m_{12}는 분자의 중심으로부터 가장 먼 원자의 중심(FF)에서부터 다시 가장 멀리 떨어져 있는 원자의 중심에서부터 나머지 원자들의 중심까지의 거리로부터 얻은 거리 분포의 1차, 2차, 3차 모멘트들이다.

선행 연구에서는 두개의 경계구(r_1, r_2)를 사용하여 검색을 하였으나 본 연구에서는 검색을 위해서 하나의 경계구(r_2, 이후 r로 명명)만을 사용한다. 주어진 *M3D* 포맷은 다음과 같이 수정된다.

$$M3D+ = \{n, r, m_1, \ldots, m_{12}, \text{PDB\_ID}\}$$

본 연구에서는 Fig. 2와 같이 상업용 RDBMS를 쓰지 않고 3차원 형상벡터를 빠르게 저장/검색할 수 있도록 자료구조를 만든다. 단백질의 형상 벡터를 데이터베이스화하여 이를 원자의 개수로 정렬하려면 C++의 표준자료구조인 STL(Standard Template Library)의 Multimap 자료구조를 쓰는 것이 바람직하다. Multimap은 <key, data>를 쌍(pair)으로 갖는 자료구조이며 key에 대해서 이진트리(binary tree)로 정렬되어 있어 검색이 매우 빠르

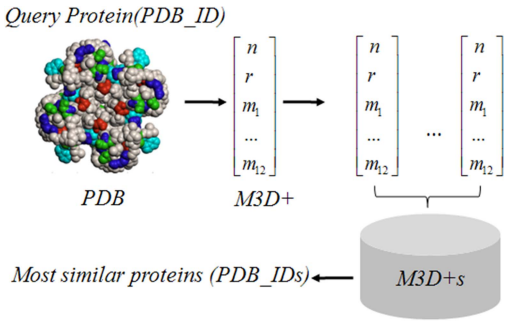

Fig. 2 Proposed search scheme in overall view

다. 이때, 원자의 개수를 key 값으로 쓰면 N개의 단백질의 형상 벡터에 대해서 M개만 추려낼 수 있다.

2.2 형상 검색을 위한 2단계 검색 도식

형상 검색을 위한 검색도식은 다음과 같다. 먼저, 사용자가 질의 단백질을 *M3D+* 포맷으로 변환한다. 변환된 *M3D+* 포맷으로부터 원자의 개수(n)를 읽는다. 다음으로 검색하고자 하는 단백질들의 검색을 제한하기 위해서 원자의 개수 범위를 결정한다. 대게 질의 단백질의 원자의 개수의 1/10에서 10배 범위를 입력해준다. 원자의 개수가 2배에서 1/2배 사이의 범위만 지정해도 유사 단백질 검색에 큰 문제는 없을 것이지만 본 연구에서 제안한 *M3D+* 기반의 검색은 속도에 장점을 갖고 있기 때문에 그보다 5배 넓은 범위를 검색하도록 설정하였다. 이러한 검색범위는 최적화된 검색 범위는 아니며 최적화된 검색 범위를 찾는 문제는 또 다른 주요 연구 주제로써 본 논문의 범위를 벗어나므로 논의하지 않는다.

그러면 Multimap에서 Key를 조회하여 검색범위를 결정한다. 그 다음에는 주어진 단백질의 경계구인 r을 읽는다. 제안된 자료구조는 Multimap의 <Key, Data>에서 Data에 다시 Multimap<Key=r, Data>을 갖는 구조이다(Fig. 3). 질의 단백질의 *M3D+*로부터 읽어들인 r로부터 r/2에서 2 r까지 검색을 한다. 그러면 두번의 검색구간 단축을 이룰 수 있다. 이때 r의 검색범위를 어떻게 설정하는 것이 최적인가 하는 문제는 아직 탐색되지 않았다. 주어진 데이터베이스내의 단백질 파일들은 전부 *M3D+* 포맷으로 변환되어 있다고 가정한다. 참고로 오프라인에서 Visual Studio 2008의 C++ 컴파일러로 프로그래밍한 프로그램에서 PDB 파일의 *M3D+* 포맷으로의 변환은 2초 이내이다.

Multimap < Key, Data1(= Multimap<Key, Data2>>

= Multimap < n, Data1(= Multimap<r, Data2>>

where, Data2 = (m_1, m_2, … , m_{12})

Fig. 3 Proposed data structure for protein screening

2.2.1 원자개수에 의한 검색 구간 단축

주어진 질의 단백질 p에 대해서 정렬이 되지 않은 단백질 데이터베이스를 D={p_1, …, p_N}라고 할 때, 단백질 p는 N개의 단백질과 3차원 비교 연산을 수행한다. 이때, 단백질 형상을 직접 비교하는 것은 아니며 대게 3차원 형상벡터(3D shape vector)나 3차원 형상기술자(3D shape descriptor)로 변환된 자료를 대상으로 검색을 수행한다. 만약 N의 개수가 많으면 이 계산은 매우 비싸며 따라서 N의 개수를 보다 적은 M(N⊃M)으로 바꿀 수 있으면 계산량을 줄일 수 있을 것이다.

검색해야 하는 양을 줄이기 위해서는 합리적인 기준이 적용되어야 하며 검색에서 배제되는 단백질들이 주어진 질의 단백질 p와 유사할 가능성이 적어야 한다.

이러한 기준에 적합하면서 데이터베이스의 키(key)의 역할을 할 수 있는 것이 바로 원자의 개수이다. 주어진 단백질 p에 대해서 원자의 개수가 1,000개라면 이 단백질은 원자의 개수를 10,000을 갖는 단백질이나 100을 갖는 단백질과 형상적으로 유사할 가능성은 매우 낮다는 것을 직관적으로 알수가 있다.

2.2.2 경계구에 의한 검색 구간 단축

주어진 질의 단백질 p에 대해서 유사한 형상이 될 가능성은 그 크기에 달려 있다. 주어진 단백질보다 너무 크거나 작은 단백질들은 검색에서 배제해도 무방하다. 이를 위해서는 단백질의 경계구를 이용하는 것이 바람직하다. 구들의 집합으로 구성된 단백질의 정확한 경계구를 구하는 것은 다음 Fig. 4와 같다. 먼저, 분자의 중심에서부터 가장 먼 원자의 중심까지의 거리를 계산하기 위해서는 각 원자의 중심점의 위치를 알아야 한다. 이는 Fig. 5에서 보는 바와 같이 PDB 파일의 ATOM 레코드의 x, y, z 필드를 이용하면 된다. 여기서 분자의 중심점(MC)를 찾고 그 다음에 MC로부터 가장 멀

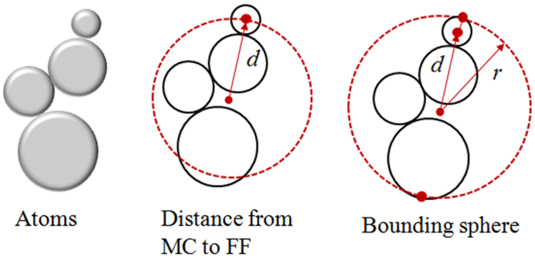

Fig. 4 Determination of r

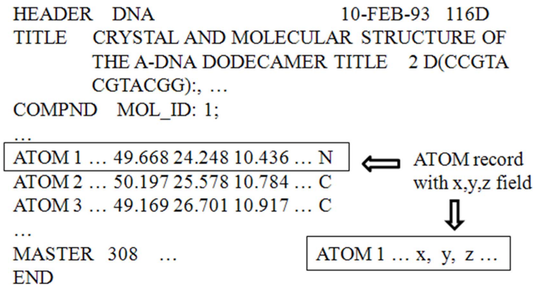

Fig. 5 PDB file format and ATOM records (with x, y, z field)

리 떨어져 있는 원자의 중심점의 위치(FA)를 찾는다. 우리가 찾고자 하는 분자의 반지름의 길이(r)는 거리(MC, FA) 값에 FF가 속한 원자의 반지름을 더한 값이다.

2.2.3 검색 구간 단축 알고리즘

2.2.1절과 2.2.2절에서 논의된 검색 구간 단축에 의한 검색 속도 가속화 방안에 대한 알고리즘은 다음과 같다.

Procedure 1 Search with n and r.
Input: query protein p and target proteins(t) of *M3D+* database : Multimap<n, Multimap<r, USR>>
Output: Most similar proteins(PDB_IDs)
Step 1. Input user specific number range(n_1, n_2).
Step 2. Search with key (=n_1 and n_2) in Multimap<n, Multimap<r, USR>>.
Step 3. Read r in query protein p.
Step 4. Search with key (=r/2 and 2r) in Multimap<r, USR>.
Step 5. Compare and calculate the similarity USR of p and each USR of *M3D+* using Eq. 1 in search range.
Step 6. Return the list of most similar *M3D+*(sorted by high score). Stop.

여기서, Step 5의 질의 단백질의 USR과 *M3D+*의 USR과의 유사도 계산은 다음의 식을 이용한다.

$$S = \frac{1}{1 + \frac{1}{12}\sum_{i=1}^{12}\left|M_i^p - M_i^t\right|} \in (0,\ 1] \qquad (1)$$

여기서, S는 두 형상의 유사도 값(similarity value)이며 p는 질의 단백질을 의미하고 t는 타겟 단백질을 의미한다. 이 값은 (0, 1] 사이의 값을 가지며 높을수록 형상의 유사도가 높다. 제안된 방법은 두 단백질의 형상 비교를 12개의 수치값을 갖는 두 벡터의 비교로 압축하였기 때문에 매우 빠르다. 또한 형상 유사도 함수도 맨하턴 거리 척도를 사용하기 때문에 매우 빠르다는 장점이 있다.

Fig. 6은 제안된 검색용 자료구조를 도식화한 것이다. Multimap은 이진트리이며 항상 정렬되어 있다. 이 Multimap은 key로 n을 가진다. 그리고 데이터로 Multimap<r, USR>을 가진다. 이를 통해서 두가지 종류의 키로 검색을 수행하면서 검색범위를 줄일수 있다. 특정 질의 단백질에 대해서 검색을 수행하는 모습은 Fig. 7과 같다.

Fig. 7은 제안된 검색구조(Multimap<r, Multimap<r, USR>>에서 실제로 n과 r의 값을 읽어서 검색을 실시하는 도식이다. 검색범위(search range)가 2.2절에서 논의한 바와 같이 결정되면(n_q, r_q) Multimap은 Key인 n에 의해서 하 나씩 원소들을 조회하게 된다(가로 진행 방향).

그런데 해당 Multimap은 key에 매핑되는 원소가 또다른 Multimap이기 때문에 이번에는 해당 Multimap의 key인 r에 의해서 원소들인 USR을 조회하게 된다(세로 진행 방향). 이렇게 가로→세로→가로→세로를 반복하면서 지정한 구간만을 조

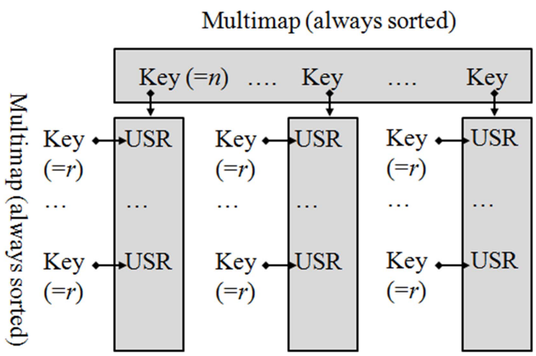

Fig. 6 Search structure with Multimap<n, Multimap<r, USR>>

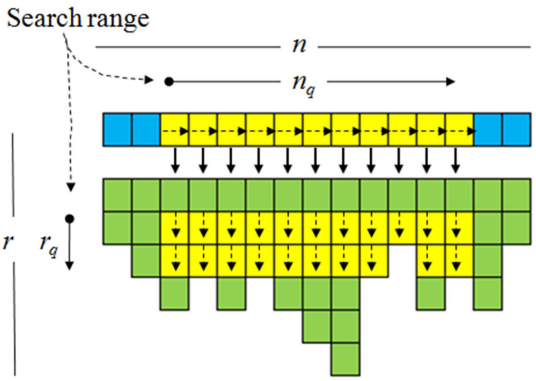

Fig. 7 Search direction in Multimap<*n*, Multimap<*r*, USR>>

Table 1 PDB files in the experiment

No	PDB ID	File size (KB)
1	163d	317
2	1a20	82
3	1a89	254
4	1a8x	83
5	1a9a	182
6	1aag	203
7	1aao	78
8	1abl	492
9	1afy	122
10	1aji	1,663
11	1akf	622
12	1alm	128
13	1als	127
14	1alt	772
15	1an3	120
16	1apk	127
17	1as1	102
18	1at8	42
…	…	…
1375	8cel	115

Table 2 Screening for similar proteins

Step	time
Preprocessing (converting from all PDB files to *M3D+*)	10 m. 52 sec.
M3D+ converting from Query protein *p* (PDB_ID: 1AA7)	2 sec.
Screening (*n*, *r*) with *p*	21 sec.

회하게 된다.

3. 실 험

본 연구에서 제안된 *M3D+* 기반의 유사단백질 검색방법의 타당성을 실증하기 위해서 실험을 실시하였다. 실험 데이터는 RCSB Protein Data Bank에서 다운받은 PDB 파일들을 사용했다[6]. 단백질의 개수는 1,375개이며 용량은 588 MByte이다.

Table 1은 실험에 사용된 PDB 파일들과 그 용량이다. Table 2는 질의 단백질 *p*에 대해서 검색을 수행한 실행 시간을 정리한 결과이다. Table 3은 질의 단백질 *p*에 대해서 유사 단백질을 검색한 결과이다. 질의 단백질 *p*는 PDB_ID: 1AA7이다. 전처리 과정은 디스크에 있는 모든 PDB 파일들을 *M3D+*로 변환하는 과정이다(10분 52초). 이 *M3D+*는 Multimap<*n*, Multimap<*r*, USR>>에 저장된다. 질의 단백질 *p*는 스크리닝을 위해서 *M3D+*로 변환된다(2초). 스크리닝에는 21초가 걸렸다.

결과는 Table 3과 같다. 질의 단백질 *p*에 대해서 가장 유사한 결과를 3개 도출했는데 기존의 순차적인 정렬 상태에서의 검색결과와 제안된 방법의 검색 결과는 약 5배에서 9배 정도 빠른 것으로 나타났다.

4. 결 론

본 연구에서는 원자의 개수와 경계구에 기반한 단백질 형상 검색의 가속화 방안을 제안했다. Ballester와 Richard가 제안한 3차원 형상 기술자인 USR은 정보압축력이 뛰어나고 정확한 형상기술능력을 갖추고 있다. 그러나 검색을 빠르게 하기 위한 메커니즘은 없다. 이를 개선하기 위한 선행연구가 있었다[9]. 그러나 검색 속도를 가속화하기 위한 방식은 깊이 논의되지 않았다. 본 연구에서는 검색속도를 개선하기 위해서 선행연구에서 제안된 *M3D*를 개량한 *M3D+*를 제안했다.

본 연구에서는 또한 제안된 방법을 구현하기 위한 자료구조를 논했다. 실험의 결과, 주어진 *N*개의 단백질들을 훨씬 적은 개수인 *M*개에 대해서만 3차원 형상비교를 함으로써 검색 범위를 줄여 속도의 이점을 얻을 수 있었다. 제안된 방법은 질의 단백질의 원자의 개수로부터 1/10에서 10배의 범

Table 3 Query protein *p* and most similar proteins

Query protein *p*	Most similar proteins	PDB_ID	Finding time	
			Previous method	Proposed method
 1AA7		1HIU	28 sec	5 sec
		1A8G	52 sec	9.1 sec
		1B9T	105 sec	10.4 sec

위까지 검색범위를 계산해서 입력해준다. 이는 검색에 최적화된 범위는 아니라는 단점이 있다. 만일 주어진 단백질의 개수로부터 최적의 검색범위를 찾아낼 수 있다면 편리할 것이다. 또한 경계구의 반지름 값인 *r*을 이용해서 검색범위를 2차적으로 줄일 수 있었다. 이때, 자동으로 *r*/2에서부터 2*r*의 범위까지 검색한다. 만약 이 범위를 최적화 할 수 있다면(예를 들어, 0.8*r*에서 1.2*r*처럼) 검색범위를 최적화함으로써 또다른 속도의 이점을 얻을 수 있을 것이다. 이는 추후 연구 과제로 탐색해볼만한 연구라고 판단된다.

References

1. Akbar, S., Kung, J. and Wagner, R., 2006, Exploiting Geometrical Properties on Protein Similarity Search, *In 17th Proceedings on International Conference on Database and Expert Systems Applications (DEXA'06)*, pp.228-234.
2. Ankerst, M., Kastenmuller, G., Kriegel, H.-P. and Seidl, T., 1999, Nearest Neighbor Classification in 3D Protein Databases, *In Proceedings of 7th International Conference on Intelligent Systems for Molecular Biology*, pp.34-43.
3. Aung, Z., Fu, W. and Tan, K.L., 2003, 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.
4. Ballester, P.J. and Richard, W.G., 2007, Ultrafast Shape Recognition to Search Compound Databases for Similar Molecular Shapes, *Journal of Computational Chemistry*, 28, pp.1711-1723.
5. Bemis, G.W. and Kuntz, I.D., 2007, A Fast and Efficient Method for 2D and 3D Molecular Shape Description, *Journal of Computer Aided Molecular Design*, 6, pp.607-628.
6. Berman, H.M. *et al.*, 2000, The Protein Data Bank, *Nucleic Acid Res.*, 28, pp.235-242.
7. Good, A.C. and Richards, W.G., 1998, Explicit Calculation of 3D Molecular Similarity, *Perspective Drug Discovery Design*, 9, pp.321-338.
8. Hall, P., 1983, A Distribution is Completely Determined by Its Translated Moments, *Probability Theory and Related Fields*, 62, pp.355-359.
9. Lee, J. and Park, J.Y., 2009, 3D Shape Descriptor with Interatomic Distance for Screening the Molecular Database, *Transactions of the Society of CAD/CAM Engineers*, 14(6), pp.404-414.
10. Kransnogor, N. and Pelta, D.A., 2007, Measuring the Similarity of Protein Structures by Means of the Universal Similarity Metric, *Bioinformatics*, 20, pp.1014-1021.
11. Yeh, J.-S. *et al.*, 2005, A Web-based Three Dimensional Protein Retrieval System by Matching Visual Similarity, *Bioinformatics Applications Note*, 21, pp.3056-3057.

이 재 호

1997년 한성대학교 산업공학과 학사
1999년 동국대학교 산업공학과 석사
2007년 동국대학교 산업공학과 박사
2011년~현재 ㈜유플러스네트웍스 연구소장
관심분야: VR/RP, 2D/3D Shape Search, BioCAD

박 준 영

1982년 한양대학교 기계공학과 학사
1984년 University of Minnesota 산업공학 석사
1991년 University of Michigan 산업공학 박사
1995년~현재 동국대학교 산업시스템공학과 교수
관심분야: Mass Customization, PSS, PLM, BioCAD