Three-Dimensional Shape-Structure Comparison Method for Protein Classification

Petros Daras, Dimitrios Zarpalas, Apostolos Axenopoulos,
Dimitrios Tzovaras, and Michael Gerassimos Strintzis

Abstract—In this paper, a 3D shape-based approach is presented for the efficient search, retrieval, and classification of protein molecules. The method relies primarily on the geometric 3D structure of the proteins, which is produced from the corresponding PDB files and secondarily on their primary and secondary structure. After proper positioning of the 3D structures, in terms of translation and scaling, the Spherical Trace Transform is applied to them so as to produce geometry-based descriptor vectors, which are completely rotation invariant and perfectly describe their 3D shape. Additionally, characteristic attributes of the primary and secondary structure of the protein molecules are extracted, forming attribute-based descriptor vectors. The descriptor vectors are weighted and an integrated descriptor vector is produced. Three classification methods are tested. A part of the FSSP/DALI database, which provides a structural classification of the proteins, is used as the ground truth in order to evaluate the classification accuracy of the proposed method. The experimental results show that the proposed method achieves more than 99 percent classification accuracy while remaining much simpler and faster than the DALI method.

Index Terms—Information search and retrieval, classification, protein databases.

1 Introduction

The structure of a molecule in 3D space is the main factor which determines its chemical properties as well as its function. All information required for a protein to be folded in its natural 3D structure is coded in its amino acid sequence. Therefore, the 3D representation of a residue sequence and the way this sequence folds in the 3D space are very important in order to be able to understand the "logic" in which a function or biological action of a protein is based on. With the technology innovation and the rapid development of X-Ray crystallography methods and NMR spectrum analysis techniques, a high number of new 3D structures of protein molecules is determined [2]. The 3D structures are stored in the world-wide repository Protein Data Bank (PDB) [1]. The number of the 3D molecular structure data increases rapidly since almost 200 new structures are stored per month in PDB. Today there are more than 24,000 3D proteins and nucleic acid molecules in this repository.

The Protein Data Bank [1], [12] is the primary repository for experimentally determined 3D protein structures. It was created in 1971 at Brookhaven National Laboratories (BNL) in the USA and contained seven macromolecule structures. These structures were created using crystallography methods. During the 1970s, the increase rate of entries was relatively low. Since 1980, the increase rate has become dramatically high due to the rapid technological development. In addition to the atom coordinates, PDB entries may contain additional information such as references, structure details, or other features. Every new structure undergoes a correctness control by using appropriate software. After its successful evaluation, the protein is given an ID (code number) and it becomes available for public use.

Since 1958, when the first 3D structure of the protein myoglobin was determined, up to now, the complexity and the variety of the protein structures has increased as the number of the new determined macromolecules has. Therefore, a need for a classification of proteins is obvious, which may result in a better understanding of these complicated structures, their functions, and the deeper evolutionary procedures that led to their creation. In molecular biology, many classification schemata and databases are available. These are briefly reviewed below.

The SCOP (Structural Classification of Proteins) protein database, which is held at the Laboratory of Molecular Biology of the Medical Research Council (MRC) in Cambridge, England, describes the structural and evolutionary relationships between proteins of known structure [4]. Since the existing automatic tools for the comparison of secondary structure elements cannot guarantee 100 percent success in the identification of protein structures, SCOP uses experts' experience to carry out this task. This is not a simple task considering the complexity of protein structures, which vary from single structural elements to vast multidomain complexes.

Proteins are classified in a hierarchical manner that reflects their structural and evolutionary relationship. The main levels of the hierarchy are "Family" (based on the proteins' evolutionary relationships), "Superfamily" (based on some common structural characteristics), and "Fold"

- *P. Daras, D. Zarpalas, A. Axenopoulos, D. Tzovaras, and M.G. Strintzis are with the Informatics and Telematics Institute (ITI), 1st Km Thermi-Panorama Road, Thermi-Thessaloniki, PO Box 361, Gr-57001, Greece. E-mail: {daras, zarpalas, axenop, tzovaras}@iti.gr.*
- *M.G. Strintzis is with the Electrical and Computer Engineering Department, Aristotle University of Thessaloniki, Thessaloniki, GR-54124, Greece. E-mail: daras@iti.gr, strintzi@eng.auth.gr.*

Manuscript received 24 Nov. 2004; revised 23 Sept. 2005; accepted 27 Nov. 2005; published online 31 July 2006.
For information on obtaining reprints of this article, please send e-mail to: tcbb@computer.org, and reference IEEECS Log Number TCBB-0195-1104.

(based on secondary structure elements). There are four main structural classes of proteins according to the way of folding their secondary structure elements:

1. all-a (consist of a-helices),
2. all-b (consist of b-sheets),
3. a/b (a-helixes and b-sheets alternating in protein structure), and
4. a+b (a-helixes and b-sheets located in specific parts of the structure).

The CATH (Class, Architecture, Topology, and Homologous superfamily) database [5], which is held at the UCL University of London, contains hierarchically classified structural elements (domains) of the proteins stored in the PDB (Protein Data Bank) database [1]. The CATH system uses automatic methods for the classification of domains, as well as experts' contribution, where automatic methods fail to give reliable results. For the classification of structural elements, five main hierarchical levels are used:

- *Class*: The class is determined by the percentage of secondary structure elements and their packing.
- *Architecture*: Describes the organization of the secondary structure elements.
- *Topology*: Provides a complete description of the hole schema and the way the secondary structure elements are connected.
- *Homologous Superfamily*: Structural elements that have at least 35 percent amino-acid sequence identity belong to the same Homologous Superfamily.
- *Sequence*: At this last level of hierarchy, the structures of the same Homologous Superfamily are further classified according to the similarity of their amino-acid sequences.

The FSSP (Families of Structurally Similar Proteins) database, which was created according to the DALI classification method [6], [7] and is held at the European Bioinformatics Institute (EBI) [8], provides a sophisticated classification method. The similarity between two proteins is based on their secondary structure. The evaluation of a pair of proteins is a highly time consuming task, so the comparison between a macromolecule and all the macromolecules of the database requires days. Therefore, one representative protein for each class is defined. Every new protein is compared only to the representative protein of each class. However, for an all-to-all comparison of the 385 representative proteins of the database, an entire day is needed [29].

The classification method of the DALI algorithm [6], [7] is based on the best alignment of protein structures. The 3D coordinates of every protein are used for the creation of distance matrices that contain the distance between amino acids (the distance between their $C^{alpha} - C^{alpha}$ atoms). These matrices are, first, decomposed into elementary formats, e.g., hexapeptidic-hexapeptidic submatrices. Similar formats make pairs and the emerging formats create new coherent pairs. Finally, a Monte Carlo procedure is used for the optimization of the similarity measure concerning the inner-molecular distances. The DALI method contains a definition of representatives, which are proteins with some special characteristics so that no two representatives have more than 25 percent amino-acid sequence identity.

This method is very time-consuming due to the many different alignments performed, the optimization procedures, and the extremely high number of distances between amino acids since a protein may consist of thousands of amino acids.

The protein databases may contain either protein collections or proteins accompanied by annotation. An example of the latter is the SWISS-PROT database [9], with 195,000 entries, where, in addition to the protein sequences, information about their function and biological action is also available.

The PROSITE [10], [11] is a database for the classification of proteins into families of proteinic sequences and sequence domains. It is based on the observation that, despite the vast number of different proteins, those can be classified into a small number of families, according to their sequence similarities. Protein sequences or sequence domains that belong to the same family have the same functions and a common ancestor. It is obvious that proteins of the same family have parts of their sequence preserved during their evolution.

A lot of research has been performed in recent years for the classification of amino acid sequences using different approaches. In [13], a data-mining approach for motif-based classification of proteins is presented. Motifs are either short amino acid chains with a specific order or representations of multiple sequence alignments using Hidden Markov Models [14]. Motifs can be used for the prediction of proteins' properties since the behavior of a protein is a function of many motifs. By using motifs stored in several databases, such as the PROSITE database, classification rules that associate motifs with protein classes are applied. The data to be processed are in the form of a prefix tree acceptor (PTA), a tree-shaped automation. The method utilizes a Finite State Automata (FSA) algorithm to induce classification rules into a training data set. The rules are finally applied to a test data set.

As it is not feasible to study experimentally every protein in all genomes, the function and biological role of a newly sequenced protein is usually inferred from a characterized protein using sequence and/or structure comparison methods. In recent years, many methods for pairwise protein structure alignment have been proposed and are now available on the World Wide Web. In [24], a state-of-the-art survey on new methods for protein comparison that have recently been published is presented.

In [25], a method to measure structural similarity of proteins is presented. According to this method, a finite number of representative local feature (LF) patterns is extracted from the distance matrices of all protein fold families by medoid analysis. Then, each distance matrix of a protein structure is encoded by labeling all its submatrices by the index of the nearest representative LF patterns. Finally, the structure is represented by the frequency distribution of these indices, which forms the LF frequency (LFF) profile of the protein, which is, in fact, a vector of common length K. The fold similarity between a pair of

proteins can be computed by the Euclidean distance between two corresponding LFF profile vectors.

The algorithm described in [26] aims to combine the results of several existing sequence and structure comparison tools in order to map domains within protein structures with their homologs in an existing classification scheme. The comparison tools incorporated in the algorithm each utilize a different methodology for identifying homologous domains and, consequently, these tools have different advantages and limitations. The algorithm has been developed to find the homologs already classified in the SCOP database and, thus, determine classification assignments, but it can be applied to any other evolutionary-based classification scheme as well.

In [27], an information theoretic model called "coherent subgraph" mining has been developed in order to find characteristic substructural patterns within protein structural families. Protein structures are represented by graphs where the nodes are residues and the edges connect residues found within a certain distance from each other. An experimental study has been conducted in which all coherent subgraphs were identified in several protein structural families annotated in the SCOP database and a Support Vector Machine algorithm was used to classify proteins from different families under the binary classification scheme.

In [28], an approach to the problem of automatically clustering protein sequences and discovering protein families, subfamilies, etc., based on the theory of infinite Gaussian mixture models is described. The method allows the data itself to dictate how many mixture components are required to model it and provides a measure of the probability that two proteins belong to the same cluster. Finally, a classification of sequences of known structure is obtained which both reflects and extends their SCOP classifications.

Considering that proteins with similar 3D structures have similar functions, a geometric filtering can lead biologists to the investigation of new protein functions. In [15], proteins are represented as 3D models on the surface of which sample points are defined. After a translation, scaling, and rotation normalization, the models are segmented to concentric spheres and sectors and the number of sampled points is calculated per each sector and per each sphere. After this procedure, descriptor vectors are created and compared using a quadratic form distance function. The nearest neighbor indicates the class assigned to the query protein. In [16], geometric features based on geometric moments and the Fourier Transform [17] are extracted, after a translation, scaling, and rotation normalization. Descriptors are also extracted from PDB files based on primary and secondary structure characteristics. Both of the aforementioned methods use a portion of the FSSP database as ground truth and achieve a percentage of around 90 percent classification accuracy, which is very satisfactory, considering that they are less complicated than the DALI algorithm.

Another method that utilizes the geometric properties of secondary structures is based on indexing [18]. Triplets (three linear segments) of secondary structures, extracted from the 3D structures of the PDB database, are used to index 3D hash tables. The hash tables are built after computation of the angles and distances of all triplets of linear segments. In [30], a fast computational framework for classification of proteins is developed, using a series of secondary structure geometric parameter represented by an unexplored dihedral angle of a protein sequence. The comparison of two such series of dihedral angles, each representing a different protein structure, is accomplished by a similarity-search mechanism based on a translational and scale invariant indexing schema. The method is tested over 25 randomly selected proteins belonging to five different families and achieves a classification accuracy of 88 percent.

Following the same concept, we propose a new combined structure-geometric comparison algorithm, based primarily on the 3D shape of a protein and secondarily on its structure characteristics (primary, secondary structure). The method was introduced in [19] and [33] and dealt with efficient 3D model content-based search and retrieval. In this paper, the method is adapted to protein classification. More specifically, a part of the Spherical Trace Transform presented in [19] is proposed in this paper for the extraction of a vector efficiently describing the 3D structure of each protein. Having as input the PDB files, the 3D coordinates of the main atoms composing the amino acids are taken into account in order to construct a 3D model that describes the protein. These 3D protein forms are further processed in a way to be applicable to the Spherical Trace Transform. This methodology leads to the creation of completely rotation invariant descriptor vectors that perfectly describe the 3D shape of the proteins. Additionally, from the PDB files, characteristics which describe the primary and secondary structure of the proteins are also extracted. The geometrical descriptors, along with the structural descriptors, form a compound descriptor vector. This compound descriptor vector serves as input to a classification method which is used to categorize unclassified protein molecules. The classification methods used, are: 1) the Euclidean distance measure, 2) the Mean Euclidean distance measure, and 3) a variance of the Bayesian probability measure.

The paper is organized as follows: The necessary preprocessing steps are described in Section 2. The proposed method and the functionals used are described in detail in Section 3. Section 4 presents the classification schemes used in order to evaluate the classification accuracy of the method. Experimental results evaluating the proposed method are presented in Section 5. Finally, conclusions are drawn in Section 6.

2 Preprocessing

A protein P is mainly composed of Carbon (C), Nitrogen (N), Oxygen (O), Hydrogen (H), and Sulfur (S) atoms. In Fig. 1, the 3D representation of a protein is depicted. The colors used and the atomic radii are listed in Table 1. The atoms in HETATM fields are not depicted.

Since the exact 3D position of each atom and its radius are known, it may be represented by a sphere. Next, the surface of each sphere is triangulated by employing 3D modeling techniques. In this way, a sphere consists of

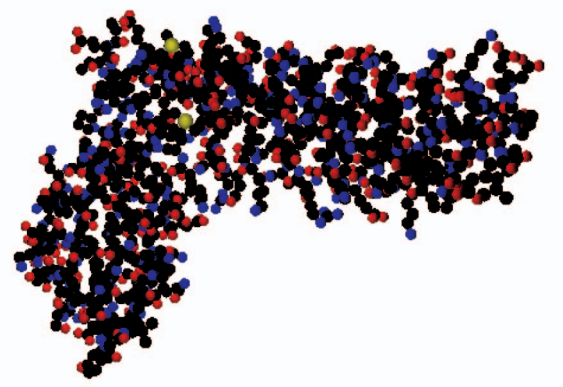

Fig. 1. The protein 1DD5.

TABLE 1
Main Atoms of a Protein

Atom	Symbol	Radius (Å)	Color
Carbon	*C*	0.77	*Black*
Nitrogen	*N*	0.70	*Blue*
Oxygen	*O*	0.66	*Red*
Hydrogen	*H*	0.37	*White*
Sulfur	*S*	1.04	*Yellow*

a small set of vertices and a set of connections between the vertices. Finally, a protein P is comprised of a set of spheres, along with the corresponding vertices $\mathbf{V}$ and the connections among them.

Then, the center of mass of P is calculated and each V is translated so that the new center of mass is at the origin. The distance d_{max} between the new origin and the most distant vertex is computed and P is scaled so that $d_{max} = 1$. The translated and scaled P is then placed into a bounding cube, which is partitioned in $(2 \cdot N)^3$ equal cube shaped voxels $\boldsymbol{u}_i$ with centers $\boldsymbol{v}_i = [x_i, y_i, z_i]$, where $i = 1, \ldots, (2 \cdot N)^3$. Let U be the set of all voxels inside the bounding cube and $U_1 \subseteq U$ be the set of all voxels belonging to the bounding cube and lying inside P.[1] Then, the discrete binary volume function $f_b(\boldsymbol{v}_i)$ of P, is defined as:

$$f_b(\boldsymbol{v}_i) = \begin{cases} 1, & when \quad \boldsymbol{u}_i \in U_1 \\ 0, & otherwise. \end{cases} \tag{1}$$

A coarser mesh is then constructed by combining every eight neighboring voxels, $\boldsymbol{u}_i$, to form a bigger voxel $\boldsymbol{\nu}_k$ with centers $\boldsymbol{\nu}_k$, $k = 1, \ldots, N^3$. The discrete integer volume function $f(\boldsymbol{\nu}_k)$ of M is defined as:

$$f(\boldsymbol{\nu}_k) = \sum_{n=1}^{8} f_b(\boldsymbol{v}_n) : \boldsymbol{u}_n \in \boldsymbol{\nu}_k. \tag{2}$$

Thus, the domain of $f(\boldsymbol{\nu}_k)$ is $[0, \ldots, 8]$.

3 The Proposed Method

The method proposed in this paper is based on the "Spherical Trace Transform" introduced in [19], which is further exploited to extract descriptors to be used for classification purposes and it is presented in the sequel for sake of completeness.

Let us define plane $\Pi(\boldsymbol{\eta}, \rho) = \{\boldsymbol{v} | \boldsymbol{v}^T \cdot \boldsymbol{\eta} = \rho\}$ to be tangential to the sphere S_ρ with radius ρ and center at the origin, at the point $(\boldsymbol{\eta}, \rho)$, where $\boldsymbol{\eta} = [cos\phi sin\theta, sin\phi sin\theta, cos\theta]$ is the unit vector in $\mathcal{R}^3$, and ρ a real positive number (Fig. 2).

The intersection of $\Pi(\boldsymbol{\eta}, \rho)$ with $f(\nu)$ produces a 2D function $\hat{f}(a, b)$, $(a, b \in \Pi(\boldsymbol{\eta}, \rho) \cap f(\boldsymbol{\nu}))$, which is then sampled and its discrete form $\hat{f}(i, j)$ $(i, j = 1, 2, \ldots, N)$ is produced. N is the number of voxels that the bounding cube is partitioned along each dimension.

The "Spherical Trace Transform" proposed in this paper can be described using the general formula:

$$SphTrace[T; F; \hat{f}] = T(F(\hat{f}(i, j))), \tag{3}$$

where $F(\boldsymbol{\eta}, \rho)$ denotes an "Initial Functional," which can be applied to each $\hat{f}(i, j)$, i.e., $F(\boldsymbol{\eta}, \rho) = F(\hat{f}(i, j))$. The set of $F(\boldsymbol{\eta}, \rho)$ is treated as a collection of spherical functions $\{F^\rho(\boldsymbol{\eta})\}_\rho$ parameterized by ρ.

Then, a set of "Spherical Functionals" $T(\rho)$ is applied to each $F^\rho(\boldsymbol{\eta})$, producing a descriptor vector $\mathbf{D1} = T(F^\rho(\boldsymbol{\eta}))$.

Let us now examine the conditions that must be satisfied by the functionals in order to produce rotation invariant descriptor vectors. Under a 3D object rotation governed by a 3D rotation matrix $\mathbf{R}$, the points $\boldsymbol{\eta}$ will be rotated:

$$\boldsymbol{\eta}' = \mathbf{R} \cdot \boldsymbol{\eta}, \tag{4}$$

therefore,

$$F(\boldsymbol{\eta}', \rho) = F(\mathbf{R} \cdot \boldsymbol{\eta}, \rho), \tag{5}$$

and, thus, rotation invariant T functionals must be applied so that $T(F(\boldsymbol{\eta}', \rho)) = T(F(\boldsymbol{\eta}, \rho))$ (Fig. 3).

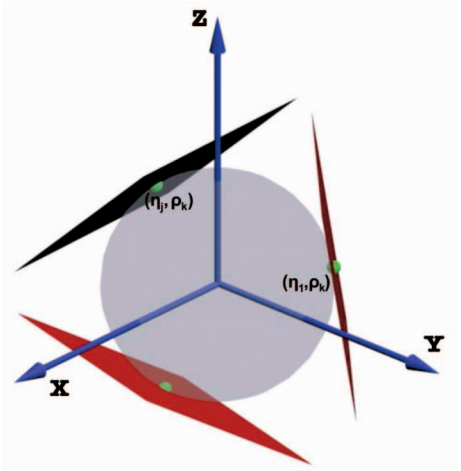

Fig. 2. Planes tangential to concentric spheres.

1. "Lying inside P" means that the corresponding voxel lies in the region that is enclosed by a sphere, which represents the atom of one of the proteins.

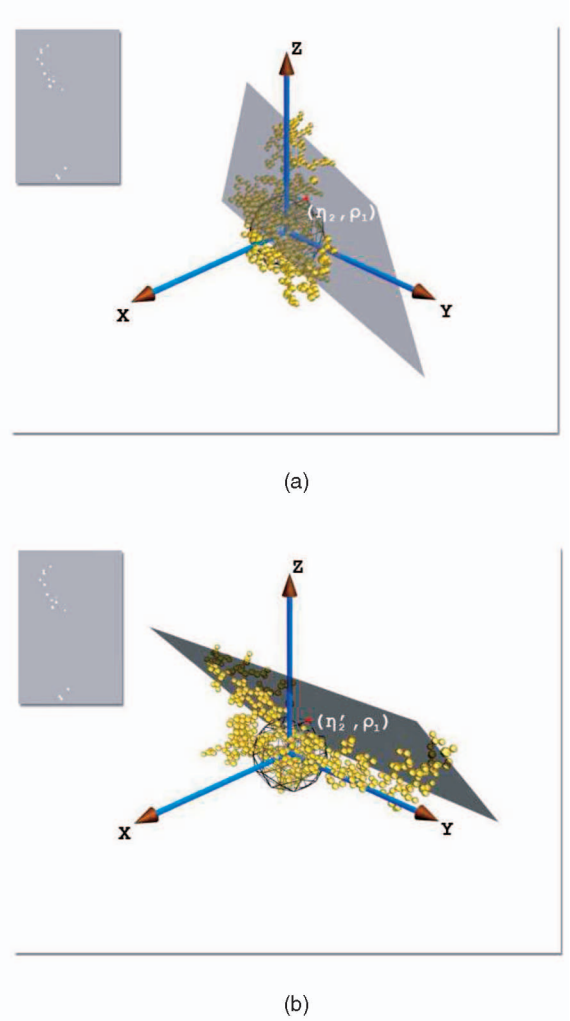

Fig. 3. Rotation of $f(\boldsymbol{x})$ rotates the $F(\boldsymbol{\eta}, \rho)$ without rotating the corresponding $f(i,j)$ (upper left image). Thus, $F(\boldsymbol{\eta}_2, \rho_1) = F(\boldsymbol{\eta}_2', \rho_1)$.

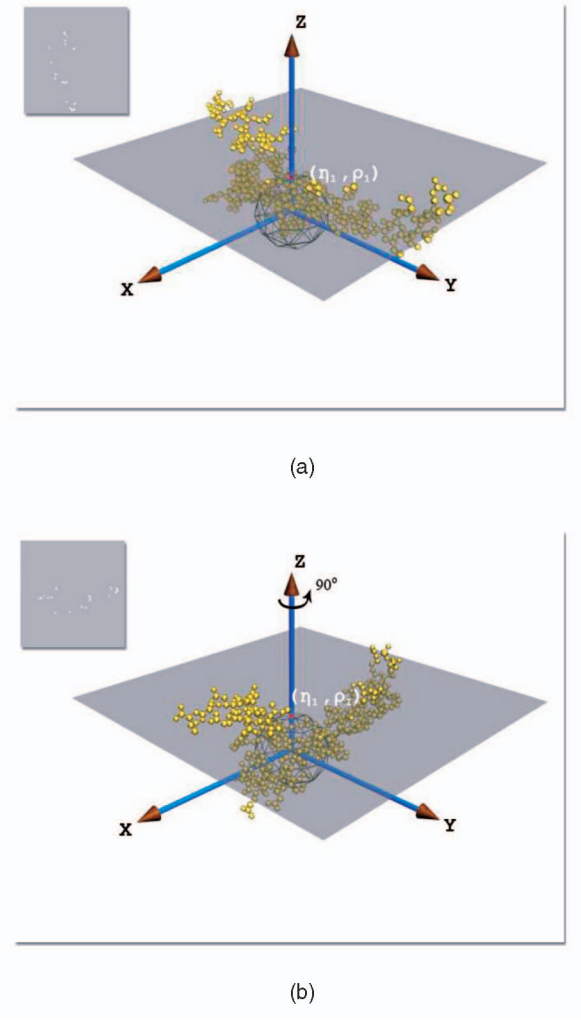

Fig. 4. Rotation of $f(\boldsymbol{x})$ rotates the $\hat{f}(i,j)$ (upper left image) without causing a rotation of the point $(\boldsymbol{\eta}_1, \rho_1)$.

In the specific case where the points $\boldsymbol{\eta}$ lie on the axis of rotation, the corresponding $\hat{f}(i,j)$ will be rotated (Fig. 4), i.e.,

$$\hat{f}'(i,j) = \hat{f}(i',j'), \tag{6}$$

and, thus, 2D rotation invariant functionals must be applied so that $F(\hat{f}'(i,j)) = F(\hat{f}(i',j'))$. Therefore, a general solution is given using 2D rotation invariant functionals F and rotation invariant spherical functionals T, producing completely rotation invariant descriptor vectors.

3.1 Initial Functionals F

The set of the Initial Functionals F consists of several harmonics of the Polar-Fourier Transform and several of the Krawtchouk moments.

3.1.1 The Polar-Fourier Transform

The Discrete Fourier Transform (DFT) is computed for each $\hat{f}_t(i,j)$, where $t = 1, \ldots, N_R$ and N_R is the total number of planes:

$$DFT_t(k,m) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} \hat{f}_t(i,j) \exp\left(-\hat{j}\left(\frac{2\pi ik}{N} + \frac{2\pi jm}{N}\right)\right), \tag{7}$$

where $k, m = 0, \ldots, N-1$. In the DFT, shifts in the spatial domain cause corresponding linear shifts in the phase component:

$$DFT_t(k,m) \exp[-\hat{j}(ak + bm)] \leftrightarrow f_t(i+a, j+b). \tag{8}$$

Thus, the DFT magnitude is invariant to circular translation. Therefore, using discrete polar coordinates:

$$\begin{aligned} r_{ij} &= \sqrt{(c_1 i + c_2)^2 + (c_1 j + c_2)^2}, \\ \xi_{ij} &= tan^{-1}\left(\frac{c_1 j + c2}{c_1 i + c_2}\right), \\ c_1 &= \frac{\sqrt{2}}{N-1} \cdot r_{max}, \\ c_2 &= -\frac{1}{\sqrt{2}} \cdot r_{max}, \end{aligned} \tag{9}$$

where $i, j = 0, \ldots, N-1$. Then, (7) becomes:

$$DFT_t(k,m) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} \hat{f}_t(r_{ij}, \xi_{ij}) \exp(-\hat{j}(kr_{ij} + m\xi_{ij})) \quad (10)$$

and rotation is converted to a circular translation of ξ. Then, the first $\mathcal{K} \times M$ harmonic amplitudes $|DFT_t(k,m)|$, where $k = 0, \ldots, \mathcal{K}-1$ and $m = 0, \ldots, \mathcal{M}-1$, are considered for each $\hat{f}_t(i,j)$. Since t refers to each plane which is described in the 3D space by the couple $(\boldsymbol{\eta}, \rho)$, $|DFT_t(k,m)|$ can be denoted as $F1_{km}(\boldsymbol{\eta}, \rho)$ or $F1^{\rho}_{km}(\boldsymbol{\eta})$.

3.1.2 Krawtchouk Moments

Krawtchouk moments [20] are a set of moments formed by using Krawtchouk polynomials as the basis function set. The nth order classical Krawtchouk polynomials are defined as:

$$K_n(x;p,N) = \sum_{\kappa=0}^{N} a_{\kappa,n,p} x^{\kappa} =_2 F_1\left(-n, -x; -N; \frac{1}{p}\right), \quad (11)$$

where $x, n-0, 1, 2, \ldots, N$, $N > 0$, $p \in (0,1)$, ${}_2F_1$ is the hypergeometric function defined as:

$${}_2F_1(a,b;c;z) = \sum_{\kappa=0}^{\infty} \frac{(a)_{\kappa}(b)_{\kappa}}{(c)_k} \frac{z^{\kappa}}{\kappa!} \quad (12)$$

and $(a)_{\kappa}$ is the Pochhammer symbol.

Following the analysis described in [19], the rotation invariant Krawtchouk moments are computed for each $\hat{f}_t(i,j)$ with spatial dimension $N \times N$ by:

$$\tilde{Q}_{km} = [\rho(k)\rho(m)]^{-(1/2)} \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} a_{i,k,p_1} a_{j,m,p_2} \nu_{ij}, \quad (13)$$

where the coefficients $a_{\kappa,n,p}$ can be determined by (11) and $\rho(k), \rho(m)$ can be calculated from the orthogonality condition [20]. It should be noted that, in our experiments, the parameters p_1, p_2 were set to 0.5 [20].

Referring to each plane $(\boldsymbol{\eta}, \rho)$, the rotation invariant Krawtchouk moments can be denoted as $F2_{km}(\boldsymbol{\eta}, \rho)$ or $F2^{\rho}_{km}(\boldsymbol{\eta})$.

3.2 Spherical Functionals T

Then, the following set of spherical functionals T is applied to each $F^{\rho}(\boldsymbol{\eta})$ in order to produce the descriptor vector:

1. $T_1(\omega) = \max\{\omega(\boldsymbol{\eta}_j)\}$,
2. $T_2(\omega) = \sum_{j=1}^{N_s} |\omega'(\boldsymbol{\eta}_j)|$,
3. $T_3(\omega) = \sum_{j=1}^{N_s} \omega(\boldsymbol{\eta}_j)$,
4. $T_4(\omega) = \max\{\omega(\boldsymbol{\eta}_j)\} - \min\{\omega(\boldsymbol{\eta}_j)\}$,

where $j = 1, \ldots, N_s$, $\omega(\boldsymbol{\eta}_j) = F^{\rho}(\boldsymbol{\eta}_j)$, ω' its derivative, and $N_s = \frac{N_R}{N_c}$, where N_c is the total number of concentric spheres, N_s is the total number of sampled points on a sphere S_{ρ} with radius ρ, and N_R is the total number of sampled points.

5. The amplitudes of the first L harmonics of the Spherical Fourier Transform (SFT).

The fifth above T functional is generated using spherical harmonics. Spherical harmonics are special functions on the sphere, generally denoted by $Y_{lm}(\boldsymbol{\eta})$, where $l \geq 0$ and $|m| \leq l$ [22].

Since spherical harmonics form a complete orthonormal set on the unit sphere, if a function σ, parameterized by the spherical coordinates $(\boldsymbol{\eta})$, can be expanded as an infinite Fourier series of spherical harmonics:

$$\sigma(\boldsymbol{\eta}_i) = \sum_{l=0}^{\infty}\sum_{m=-l}^{l} \alpha_{lm} Y_{lm}(\boldsymbol{\eta}_i), \qquad i = 1, \ldots, N_s, \quad (14)$$

then the expansion coefficients α_{lm} are uniquely determined by:

$$\alpha_{lm} = \sum_{i=1}^{N_s} \sigma(\boldsymbol{\eta}_i) Y_{lm}(\boldsymbol{\eta}_i) \frac{4\pi}{N_s}. \quad (15)$$

In our case:

$$\sigma(\boldsymbol{\eta}) = \begin{cases} F1^{\rho}_{km}(\boldsymbol{\eta}) \\ F2^{\rho}_{km}(\boldsymbol{\eta}). \end{cases} \quad (16)$$

The expansions (14) are strictly convergent in the sense that the error of the expansion reduces monotonically as l tends to infinity. Hence, the leading terms of the series are those with small values of l and m, which implies that, upon truncation, the series at a sufficiently large value of l, L, most of the detail of the function $\sigma(\boldsymbol{\eta})$ will be captured.

Further, if $\sigma(\boldsymbol{\eta})$ is rotated ($\sigma'(\boldsymbol{\eta})$ with expansion coefficients α'_{lm}), then, as is easily proven [22], the overall vector length of α'_{lm} coefficients with the same l is preserved under rotation:

$$A_l^2 = \sum_m \alpha'^2_{lm} = \sum_m \alpha^2_{lm}, \quad (17)$$

where the quantities A_l are known as the rotationally invariant shape descriptors. In the proposed method, for each l, the corresponding A_l is a spherical functional T. Therefore, the total number of spherical functionals T used is $L+4$ for each concentric sphere.

3.3 Descriptor Extraction

3.3.1 Geometrical Descriptor Extraction

In order to avoid possible sampling errors caused by using the lines of latitude and longitude (since they are concentrated too much toward the poles), each concentric sphere is simulated by an icosahedron where each of the 20 main triangles is iteratively subdivided into q equal parts to form subtriangles. The vertices of the subtriangles are the sampled points B_t. Their total number N_s, for each concentric sphere (icosahedron) C_s, with radius ρ_s, $s = 1, \ldots, N_c$, where N_c is the total number of concentric spheres, is easily seen to be:

$$N_s = 10 \cdot q^2 + 2. \quad (18)$$

Then, following the procedure described earlier, for each functional F, the descriptor vectors $\mathbf{D1}_F(l_1) = T(F^{\rho_t}(\boldsymbol{\eta}_t))$ are produced, where $l_1 = 1, \ldots, (L+4) \cdot N_c$.

3.3.2 Structural Descriptor Extraction

Besides the geometric descriptor vectors, features that characterize the primary and secondary structure of a

TABLE 2
Structural Features and Their Weights

Secondary structure features	Weight
No of HELICES	1%
No of SHEETS	1%
No of TURNS	1%
Primary structure features	**Weight**
Hydrophobic residue ratio	6%
Helix Type	1%
Residue ratio	90%

protein are also extracted [16]. More specifically, concerning the primary structure, the ratio of the amino acids' occurrences relative to the total number of amino acids (20 descriptors), the hydrophobic amino acids ratio (one descriptor), and the ratio of the helix types' occurrences (10 descriptors) contained in a protein are calculated. Concerning the secondary structure, the number of Helices (one descriptor), Sheets (one descriptor), and Turns (one descriptor), contained in a protein are also calculated. These features are listed in Table 2. All the aforementioned information is included in each PDB file. A part of a PDB file is depicted in Fig. 5.

The descriptor vector, $\mathbf{D2}$, is then produced, with length 34. Thus, the length of the compound descriptor vector $\mathbf{D} = \mathbf{D1} \bigcup \mathbf{D2}$ is $N_c \cdot (L+4) + 34$.

Our experiments presented in the sequel were performed using the values: $N_s = 2,562$, $N_c = 20$, $L = 26$, and $N = 64$, where N is the number of sampled points for each dimension of each tangential plane $\Pi(\boldsymbol{\eta}, \rho)$. The total number of sampled points on each tangential plane is $N \times N$.

4 Classification

4.1 Matching Algorithm

Let A, B be two 3D models (proteins). Also, let

$$\mathbf{D}^A(k) = [\mathbf{D}^{A1}(k_1), \mathbf{D}^{A2}(k_1), \mathbf{D}^{A3}(k_2)]^T,$$
$$\mathbf{D}^B(k) = [\mathbf{D}^{B1}(k_1), \mathbf{D}^{B2}(k_1), \mathbf{D}^{B3}(k_2)]^T$$

be two descriptor vectors, where $A1, B1$ denotes the descriptor vector extracted using Polar-Fourier Transform, $A2, B2$ denotes the descriptor vector extracted using Krawtchouk moments, $A3, B3$ denotes the descriptor vector extracted taking into account the primary and secondary structure of each protein, $k_1 = N_c \cdot (L+4)$, and $k_2 = 34$. The geometrical descriptors are compared in pairs using their L1-distance:

$$D1_{similarity} = \sqrt{\sum_{k1=1}^{N_c \cdot (L+4)} |\mathbf{D}^{A1}(k1) - \mathbf{D}^{B1}(k1)|} \qquad (19)$$

```
HEADER    IMMUNOGLOBULIN                          03-MAR-97   2PSK
TITLE     THEORETICAL MODEL OF AN FAB FRAGMENT COMPLEXED WITH THE
TITLE    2 MELANOMA-ASSOCIATED GD2 GANGLIOSIDE
COMPND    MOL_ID: 1;
COMPND   2 MOLECULE: ANTIBODY;
COMPND   3 CHAIN: L, H;
COMPND   4 FRAGMENT: FAB
...
AUTHOR    S.L.PICHLA,R.MURALI,R.M.BURNETT
REVDAT   1   04-SEP-97 2PSK    0
...
JRNL        AUTH   S.L.PICHLA,R.MURALI,R.M.BURNETT
SEQRES   1 L  213  GLN ILE VAL LEU THR GLN SER PRO ALA ILE MET SER ALA
SEQRES   2 L  213  SER PRO GLY GLU LYS VAL THR ILE THR CYS SER ALA SER
SEQRES   3 L  213  SER SER VAL SER ASN ILE HIS TRP PHE GLN GLN LYS PRO
...
HELIX    1   1 SER L  121  SER L  126  1                            6
HELIX    2   2 LYS L  182  TYR L  185  1                            4
...
SHEET    1   A 4 LEU L   4  SER L   7  0
SHEET    2   A 4 VAL L  19  ALA L  25 -1  N  SER L  24   O  THR L   5
SHEET    3   A 4 SER L  69  ILE L  74 -1  N  ILE L  74   O  VAL L  19
...
ATOM  1  N   GLN L   1  40.444   0.114  53.530  1.00 44.06      L    N
ATOM  2  CA  GLN L   1  39.136  -0.460  53.239  1.00 38.84      L    C
ATOM  3  C   GLN L   1  39.210  -1.920  52.815  1.00 33.95      L    C
ATOM  4  O   GLN L   1  39.943  -2.274  51.886  1.00 34.91      L    O
...
HETATM 3854 O HOH  1    -1.229  -1.762   5.590  1.00 15.50      W    O
HETATM 3855 O HOH  3    23.399 -21.858  56.848  1.00 10.79      W    O
HETATM 3856 O HOH  4     6.748  17.422  37.138  1.00 28.29      W    O
...
CONECT 1815 1196 1814
CONECT 2209 2208 2944
CONECT 2944 2209 2943
...
END
```

Fig. 5. A PDB file.

and

$$D2_{similarity} = \sqrt{\sum_{k1=1}^{N_c \cdot (L+4)} |\mathbf{D}^{A2}(k2) - \mathbf{D}^{B2}(k2)|}. \quad (20)$$

The overall geometrical similarity measure is determined by:

$$D_{Gsimilarity} = a_1 \cdot D1_{similarity} + a_2 \cdot D2_{similarity}, \quad (21)$$

where a_1, a_2 are descriptor vector percentage factors, which are calculated as follows: Let us assume that A belongs to a class C, which contains N_C models. Also let N_{total} be the total number of models contained in the database. Then, the factor a_1 is calculated as:

$$a_1 = \frac{\sum_{i=1}^{N_C} d_i}{\sum_{j=1}^{N_{total}-N_C} d_j}, \quad (22)$$

where d_i is the L1-distance of the descriptor vector $\mathbf{D}^{A1}$ of each model A from the descriptor vector $D^{A1'}$ of a model A' which also belongs to C and d_j is the L1-distance of the descriptor vector $\mathbf{D}^{A1}$ of the model A from the descriptor vector $\mathbf{D}^{A1''}$ of a model A'' which does not belong to C. Descriptor vectors $\mathbf{D}^{A1}$ with small values of d_i and large values of d_j are clearly appropriate for class C, in terms of successful retrieved results. The percentage factor a_2 is calculated similarly, taking into account the descriptor vector $\mathbf{D}^{A2}$. Then, a_1 and a_2 are normalized so that $1/a_1 + 1/a_2 = 100$.

Following the above approach, the discriminant power of each descriptor vector per different class is taken into account.

The structural similarity is evaluated using:

$$D_{Ssimilarity} = \sqrt{\sum_{k2=1}^{34} |\mathbf{D}^{A3}(k2) - \mathbf{D}^{B3}(k2)|}. \quad (23)$$

The overall similarity measure is determined by:

$$D_{similarity} = b_1 \cdot D_{Gsimilarity} + b_2 \cdot D_{Ssimilarity}. \quad (24)$$

The weights assigned to the different kind of descriptors are $b_1 = 90\%$ for the geometrical descriptors and $b_2 = 10\%$ for the structural descriptors. The weight allocation regarding the latest formula is listed in Table 2.

4.2 Classification Methods

In order to evaluate the classification accuracy of the proposed method, three classification schemes were used. A description of these schemes is given below.

Let $\mathbf{D}^i(j) = [D^i(1), \ldots, D^i(N_d)]$ be a compound descriptor vector, where $i = 1, \ldots, N_{total}$. N_{total} is the total number of proteins and N_d is the total number of descriptors per descriptor vector ($N_d = N_c \cdot (L+4) + 34$). Also, let C be a class with descriptor vectors:

$$M_C = \begin{bmatrix} D^1(1), & \ldots, & D^1(k), & \ldots, & D^1(S) \\ \ldots, & \ldots, & \ldots, & \ldots, & \ldots, \\ D^i(1), & \ldots, & D^i(k), & \ldots, & D^i(S) \\ \ldots & \ldots, & \ldots, & \ldots, & \ldots, \\ D^{N_C}(1), & \ldots, & D^{N_C}(k), & \ldots, & D^{N_C}(S) \end{bmatrix},$$

where N_C is the number of 3D models which belong to class C. Then, the **feature vectors** $\mathbf{f}_{C1}, \ldots, \mathbf{f}_{Ck}, \ldots, \mathbf{f}_{CS}$ are formed, where $C = 1, \ldots, N_{class}$, $\mathbf{f}_{Ck} = [D^1(k) \ldots D^i(k) \ldots D^{N_C}(k)]^T$, and N_{class} is the total number of classes.

For each $\mathbf{f}_{Ck}$, the mean,

$$\mu_{\mathbf{f}_{Ck}} = \frac{1}{N_C} \sum_{i=1}^{N_C} D^i(k), \quad (25)$$

and the variance,

$$\sigma^2_{\mathbf{f}_{Ck}} = \frac{1}{N_C} \sum_{i=1}^{N_C} (D^i(k))^2 - (\mu_{\mathbf{f}_{Ck}})^2, \quad (26)$$

are calculated. Finally, let $\mathbf{U} = [U(1), \ldots, U(N_d)]$ be a descriptor vector of an unclassified protein U.

4.2.1 Euclidean Distance Measure

The first metric of "similarity" is based on the Euclidean distance between the descriptor vectors, which is defined as:

$$M_1(\mathbf{D}, \mathbf{U}) = \left[\sum_{j=1}^{N_d} (D(j) - U(j))^2 \right]^{1/2}. \quad (27)$$

For an unclassified U, the pairwise Euclidean distances $M_1(\mathbf{D}^i, \mathbf{U})$, $i = 1, 2, \ldots, N_{total}$, are rank ordered and U is assigned to the class corresponding to the minimum distance.

4.2.2 Mean Euclidean Distance Measure

As a second metric, the Euclidean distances between a feature vector Ck and an unclassified vector $\mathbf{U}$ are used:

$$M_2(\mathbf{X}, \mathbf{U}) = \left[\sum_{j=1}^{N_d} (\mu_{\mathbf{X}_{Ci}}(j) - U(j))^2 \right]^{1/2}. \quad (28)$$

As before, the pairwise Euclidean distances $M_2(\mathbf{X}_i, \mathbf{U})$, $i = 1, 2, \ldots, N_{class}$, are rank ordered and the class with the minimum distance to U is chosen.

4.2.3 Naive Bayesian Classifier

For each class Ci, $i = 1, \ldots, N_{class}$, the mean $\mu_{\mathbf{X}_{Ci}}(j)$ and the standard deviation σ_{Ci} are calculated for each feature vector Cj. For each descriptor $U(j)$ of the unclassified protein U, the validity of the following inequality is tested:

$$\mu_{\mathbf{X}_{Ci}}(j) - a \cdot \sigma_{Ci} \le U(j) \le \mu_{\mathbf{X}_{Ci}}(j) + a \cdot \sigma_{Ci}, \quad (29)$$

where $a \in [3, 4]$. For each class Ci, the following measure is calculated:

$$B(Ci) = \sum_{j=1}^{N_d} w_{U(j)}, \quad (30)$$

where $w_{U(j)} = 1$ when $U(j)$ satisfies (29) and $w_{U(j)} = 0$, otherwise. U is assigned to the class Ci with the maximum $B(Ci)$.

5 Experimental Results

In order to evaluate the performance of the proposed method, a portion of the FSSP database [23] was used. This

TABLE 3
Protein Classes Used as Ground Truth Database

Class	No. of Proteins
12asA	4
153l	4
16pk	11
19hcA	3
1a04A	3
1a0aA	2
1a0cA	10
1a0fA	11
1a6m	189
1abwA	387
1abbzA	29
1cnzA	32
1ctqA	88
1daaA	28
1fmk	65
1igtB	335
1l92	387
1pgtA	130
1ucyE	54
1wgjA	19
1ycc	97
2cba	180
3chy	57
3lzt	449
3nul	14
4icb	33
6mhtA	14
5ptp	561
7rsa	179
8fabA	361

database was constructed according to the DALI algorithm [6], [7] and consists of 3,732 proteins classified into 30 classes (Table 3). Care was taken to include classes with different cardinalities, varying from 2 to 561 proteins. In order to get reliable results, the 3,732 proteins were randomly selected. The database can be downloaded from: ftp://ftp.iti.gr/pub/incoming/proteins.zip.

The performance of the method was evaluated in terms of overall classification accuracy [15]. More specifically, for each molecule in the database, one of the three classification methods described above is applied after removing that element from the database ("leave-one-out" experiment). A class label is then assigned to the query protein as the output of the classification method. The overall classification accuracy is the percentage of the correctly predicted class labels among all 3,732 proteins of the database and is given by:

$$Overall\ Classification\ Accuracy = \frac{Number\ of\ correctly\ predicted\ proteins}{Total\ number\ of\ proteins\ in\ the\ database}. \quad (31)$$

The overall classification accuracy can also be derived from the confusion matrix, which is widely used in classification problems [32]. The overall classification accuracy is the sum of the diagonal elements of the confusion matrix divided by the total number of classified objects.

Let FT_{km} and $Kraw_{km}$ be the descriptor vectors produced after applying the spherical functionals T to the initial functionals $F1^{\rho}_{km}(\boldsymbol{\eta})$ and $F2^{\rho}_{km}(\boldsymbol{\eta})$, respectively.

All of the produced descriptor vectors were tested experimentally in terms of overall classification accuracy. However, only the following achieved significantly high classification accuracy and are reported in this section:

$$FT = \{FT_{00}, FT_{01}, FT_{10}, FT_{02}\}$$

and

$$K = \{Kraw_{00}, Kraw_{01}, Kraw_{02}\}.$$

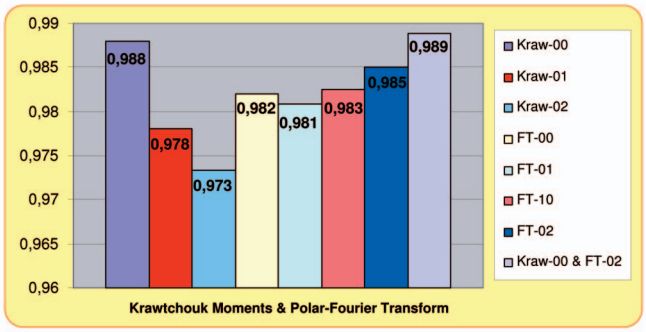

Fig. 6. Overall classification accuracy using only geometrical characteristics with the Euclidean Distance Measure method.

5.1 Evaluation of Overall Classification Accuracy Using the Euclidean Distance Measure

First, the simpler method was evaluated, which relies on the Euclidean Distance measure. The overall classification accuracy results were very satisfactory (Fig. 6 and Table 4).

As seen by Fig. 6, the use of vectors $Kraw_{00}$ and FT_{02} was found to be optimal since the percentage accuracy achieved was 98.9 percent (Fig. 6, last column).

The time needed for the extraction of the descriptor vectors of the Initial Functionals used is shown in Table 4.

In addition to the geometrical descriptors, structural descriptors are extracted as well (Table 2), which refer to the proteins' primary and secondary structure elements. The percentage of geometrical and structural features in the integrated descriptor vector was experimentally selected to be 90 percent and 10 percent, respectively. This combination

TABLE 4
Extraction Time Using Different Initial Functionals and All Spherical Functionals

Initial Functional	Descriptor extraction time
The amplitudes of the first 4 harmonics of the Polar-Fourier Transform	85 sec
First 3 Krawtchouk moments	45 sec

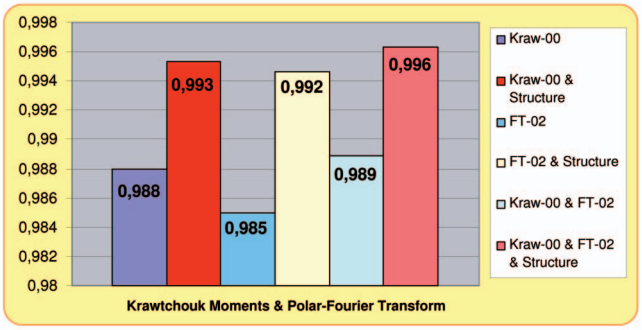

Fig. 7. Overall classification accuracy using geometrical and structural characteristics with the Euclidean Distance Measure method.

TABLE 5
The Times Needed for the Computation of the Overall Classification Accuracy Using Geometrical and Structural Characteristics with the Euclidean Distance Measure Method

Vector	Total time
$Kraw_{00}\&Struct$	225 sec
$FT_{02}\&Struct$	225 sec
$Kraw_{00}\&FT_{02}\&Struct$	395 sec

significantly increases the overall classification accuracy (Fig. 7).

The times needed for the computation of the overall classification accuracy for the entire database are shown in Table 5. These include the comparison of each query protein descriptor vector to all (3,731) descriptor vectors (all-to-all comparison). In other words, the time needed for approximately $3,731^2$ comparisons is 395 sec if the "$Kraw_{00}\&FT_{02}\&Struct$" descriptor vector is used. This is very satisfactory if we consider that the Dali algorithm requires an entire day for an all-to-all comparison of all 385 representatives of FSSP database [29].

The time needed for the complete preprocessing procedure, from the creation of the 3D structure up to the final normalization step, is approximately 3 min. Although this procedure, for a large database with thousands of proteins, may last for days, it takes place **only once** and the descriptor vectors are stored in the database along with the corresponding 3D structures.

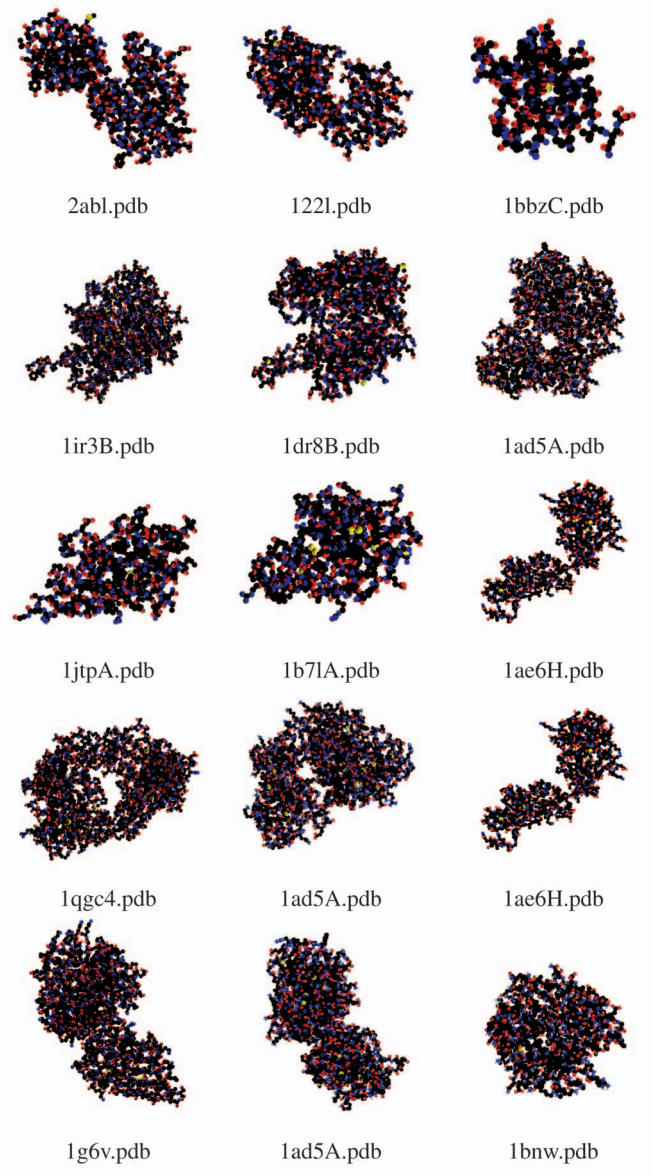

Fig. 8. Missed proteins using the Euclidean distance method. The query proteins are depicted in the first column. The second column shows the nearest neighbors, which were retrieved using the proposed method but do not belong to the same class with the query, according to the FSSP/DALI classification. The third column shows the proteins closer to the query that do belong to the same class according to the FSSP/DALI classification. It is obvious that the visual similarity between the proteins of columns 1 and 2 is greater than the similarity between the proteins of columns 1 and 3.

The FSSP/DALI database has been constructed based in part on the premise that proteins with at least 25 percent similarity in their amino acid sequence should belong to the same class even if dissimilar geometrically. Since we do not use this criterion, we do not achieve 100 percent classification accuracy. In fact, the best overall classification accuracy achieved, using the proposed method (Fig. 7, column 6), is 99.62 percent. In other words, 14 out of 3,732 proteins are misclassified. Further analysis of the misclassified proteins showed that the proposed method, which is mainly based on geometrical features (90 percent) rather than structural features (10 percent), classifies the 3D proteins differently when compared to the DALI algorithm. However, there is

TABLE 6
Classification Precision, Classification Recall, and Classification Accuracy for Each Class Using the "$Kraw_{00}\&FT_{02}\&Struct$" Descriptor Vector

Class	TP	FP	FN	TN	C_{Pre}	C_{Rec}	C_{Acc}
1	4	0	0	3728	100%	100%	100%
2	4	0	0	3728	100%	100%	100%
3	11	0	0	3721	100%	100%	100%
4	3	0	0	3729	100%	100%	100%
5	3	0	0	3729	100%	100%	100%
6	2	0	0	3730	100%	100%	100%
7	10	0	0	3722	100%	100%	100%
8	11	0	0	3721	100%	100%	100%
9	189	1	0	3542	99.47%	100%	99.96%
10	387	0	0	3345	100%	100%	100%
11	27	0	2	3703	100%	93.10%	99.92%
12	32	1	0	3699	96.97%	100%	99.96%
13	87	0	0	3645	100%	100%	100%
14	28	0	0	3704	100%	100%	100%
15	64	2	1	3665	96.97%	98.46%	99.89%
16	331	1	2	3398	99.7%	99.4%	99.89%
17	387	1	0	3344	99.74%	100%	99.96%
18	130	1	0	3601	99.24%	100%	99.96%
19	53	1	0	3678	98.15%	100%	99.96%
20	19	0	0	3703	100%	100%	100%
21	96	0	1	3635	100%	98.97%	99.96%
22	177	0	3	3552	100%	98.33%	99.89%
23	56	0	1	3675	100%	98.25%	99.96%
24	449	2	0	3281	99.56%	100%	99.92%
25	14	0	0	3718	100%	100%	100%
26	33	0	0	3699	100%	100%	100%
27	14	0	0	3718	100%	100%	100%
28	559	0	2	3171	100%	99.64%	99.95%
29	178	0	1	3553	100%	99.44%	99.97%
30	360	1	1	3370	99.72%	99.72%	99.95%

no clear answer as to which method is "more" correct. Fig. 8 depicts five missed proteins (column 1), their nearest neighbors using the proposed method (column 2), and the closest to the query proteins that belong to the same class with them according to the FSSP classification (column 3). The structures in the first column are seen to be geometrically far more similar to those in the second column than those in the third.

A more detailed view of the classification results demonstrates the high performance of the method in

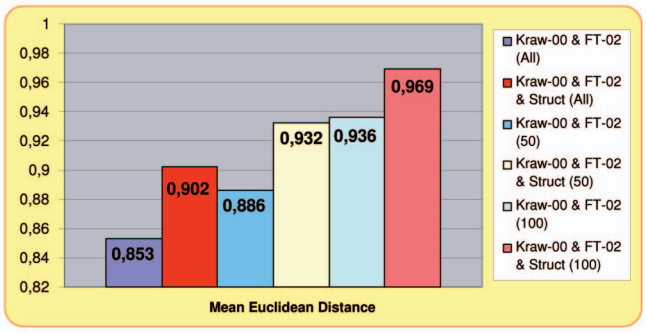

Fig. 9. Overall classification accuracy using geometrical and structural characteristics with the Mean Euclidean Distance Measure method.

application to both small and large classes. In order to evaluate the classification performance of each class, the measures of *Classification Precision* (C_{Pre}), *Classification Recall* (C_{Rec}), and *Classification Accuracy* (C_{Acc}) were used [31]. These are given by the following equations:

$$C_{Pre} = \frac{TP}{TP + FP}, \tag{32}$$

$$C_{Rec} = \frac{TP}{TP + FN}, \tag{33}$$

$$C_{Acc} = \frac{TP + TN}{TP + FP + FN + TN}, \tag{34}$$

where:

- *TP*: The number of correctly included (True Positive) class objects.
- *FP*: The number of incorrectly included (False Positive) objects.
- *TN*: The number of correctly excluded (True Negative) objects.
- *FN*: The number of incorrectly excluded (False Negative) objects.

The values of TP, FP, FN, and TN, along with the values of C_{Pre}, C_{Rec}, C_{Acc} for each class, when the "$Kraw_{00}\&FT_{02}\&Struct$" descriptor vector is used, are presented in Table 6.

Table 6 illustrates the effectiveness of the proposed method, showing its high performance in terms of *Classification Precision*, *Classification Recall*, and *Classification Accuracy* for each class.

As the protein database increases, the time needed for a one-to-all comparison and classification of an unknown protein increases dramatically. For such use, other faster classification methods, based on statistical features extraction, were evaluated. A detailed description of these methods was given in Section 4.

5.2 Evaluation of Overall Classification Accuracy Using the Mean Euclidean Distance Measure

In Fig. 9 and in Table 7, the results of the Mean Euclidean Distance method are presented: The first two columns depict the overall classification accuracy of the method with all classes included, with ($Kraw_{00}\&FT_{02}\&Struct$ All column) or without ($Kraw_{00}\&FT_{02}$ All column) structural

TABLE 7
The Times Needed for the Computation of the Overall Classification Accuracy with the Mean Euclidean Distance Measure Method

Vector	Total time
$Kraw_{00}\&FT_{02}(All)$	28 sec
$Kraw_{00}\&FT_{02}\&Struct(All)$	31 sec
$Kraw_{00}\&FT_{02}(50)$	56 sec
$Kraw_{00}\&FT_{02}\&Struct(50)$	59 sec
$Kraw_{00}\&FT_{02}(100)$	105 sec
$Kraw_{00}\&FT_{02}\&Struct(100)$	109 sec

features. The next four columns present the results when the Mean Euclidean Distance method is applied only to classes with a relatively large number of proteins. The class that best fits the query protein is then included in the Euclidean Distance algorithm, which is applied to the remaining small classes. The key reason for this fused algorithm selection is that statistical measures are more reliable when applied to large classes (over 50 or 100 proteins) since the higher the number of proteins in a class, the more reliable the statistical measures. In the third and fourth column, the Mean Euclidean method is applied to classes with a number of proteins larger than 50, while, in the last two columns, the number of proteins is larger than 100. Experiments proved that the overall classification accuracy in large classes with more than 100 proteins is very satisfactory, while the time needed for the classification procedure is four times smaller than that of the Euclidean Distance method.

5.3 Evaluation of Overall Classification Accuracy Using the Naive Bayesian Classifier

Finally, similar experiments, based on the Naive Bayesian Classifier (Section 5.2.3), were performed. The results are presented in Fig. 10 and in Table 8. It is obvious that, like the previous method, Naive Bayesian Classifier achieves satisfactory classification results as well as low computational complexity without, however, outperforming the methods presented in the previous paragraphs.

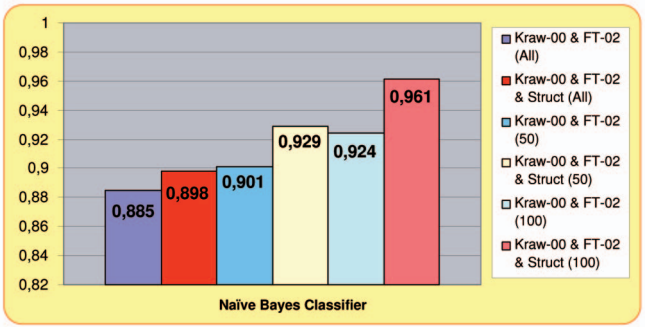

Fig. 10. Overall classification accuracy using geometrical and structural characteristics with the Naive Bayesian Classifier.

TABLE 8
The Times Needed for the Computation of the Overall Classification Accuracy with the Naive Bayesian Classifier

Vector	Total time
$Kraw_{00}\&FT_{02}(All)$	29 sec
$Kraw_{00}\&FT_{02}\&Struct(All)$	33 sec
$Kraw_{00}\&FT_{02}(50)$	57 sec
$Kraw_{00}\&FT_{02}\&Struct(50)$	60 sec
$Kraw_{00}\&FT_{02}(100)$	107 sec
$Kraw_{00}\&FT_{02}\&Struct(100)$	111 sec

5.4 Evaluation of Information Retrieval Performance

Apart from the classification performance, the efficiency of the proposed shape comparison method was evaluated in terms of information retrieval performance. In this case, each model of the database is used as query and the retrieved proteins are ranked in terms of shape similarity to the query. For the presentation of the results, the *Information Retrieval Precision-Recall* curve was used, where precision is the proportion of the retrieved models that are relevant to the query and recall is the proportion of relevant models in the entire database that are retrieved as a result of the query. More precisely, precision and recall are defined as:

$$Precision = \frac{N_{detection}}{N_{detection} + N_{false}}, \tag{35}$$

$$Recall = \frac{N_{detection}}{N_{detection} + N_{miss}}, \tag{36}$$

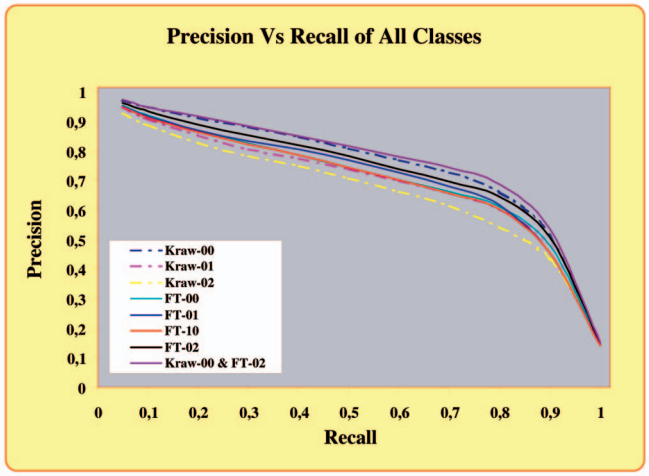

Fig. 11. Precision-recall curve for the geometrical descriptor vectors.

TABLE 9
Protein Classes to Be Compared

$Class$	$1a6m$	$1l92$	$2cba$
$Number\ \ of\ \ protein\ \ structures$	189	387	180

where:

- $N_{detection}$ = number of relevant models retrieved,
- N_{false} = number of irrelevant models retrieved,
- N_{miss} = number of relevant models not retrieved.

Fig. 11 depicts the Information Retrieval Precision-Recall curve for all geometrical descriptor vectors used.

5.5 Comparison with Existing Methods

It must be emphasized that the goal of the proposed method is not to introduce a new classification scheme, but to provide a fast geometric filtering so as to achieve a first quick classification of a new protein sequence. Thus, comparison with classification schemes, such as DALI, SCOP, CATH, etc., or with methods that focus on finding biologically relevant sequence similarities, such as BLAST, PSI-BLAST [34], etc., is clearly not meaningful. However, comparison with the methods presented in [16], [15], which are also based on the geometrical similarity of proteins, is fully meaningful and is presented in the sequel.

First, the proposed method is compared with the method [16] in terms of retrieval performance. In [16], three classes are chosen from the Dali server, which are listed in Table 9. Then, the "precision versus recall" is calculated for each class.

Fig. 12a depicts the Information Retrieval Precision-Recall curve of the three classes by using $Kraw_{00}\&FT_{02}$ descriptors. In the next three diagrams, the precision-recall curve of each class is compared with the respective curve of the method presented in [16]. It can be inferred that the proposed method demonstrates a slight improvement in the last values of recall, while it retains high performance in the first values of recall.

The proposed method is also compared with the one presented in [15] in terms of overall classification accuracy. Since the experiments in [15] were conducted on a different set of protein structures, an extra effort in developing this method for our protein data set was required. The results are presented in Fig. 13, where it is obvious that the proposed method outperforms the one presented in [15] when applied to single domain chains. For multidomain proteins, however, the experimental results are inconclusive.

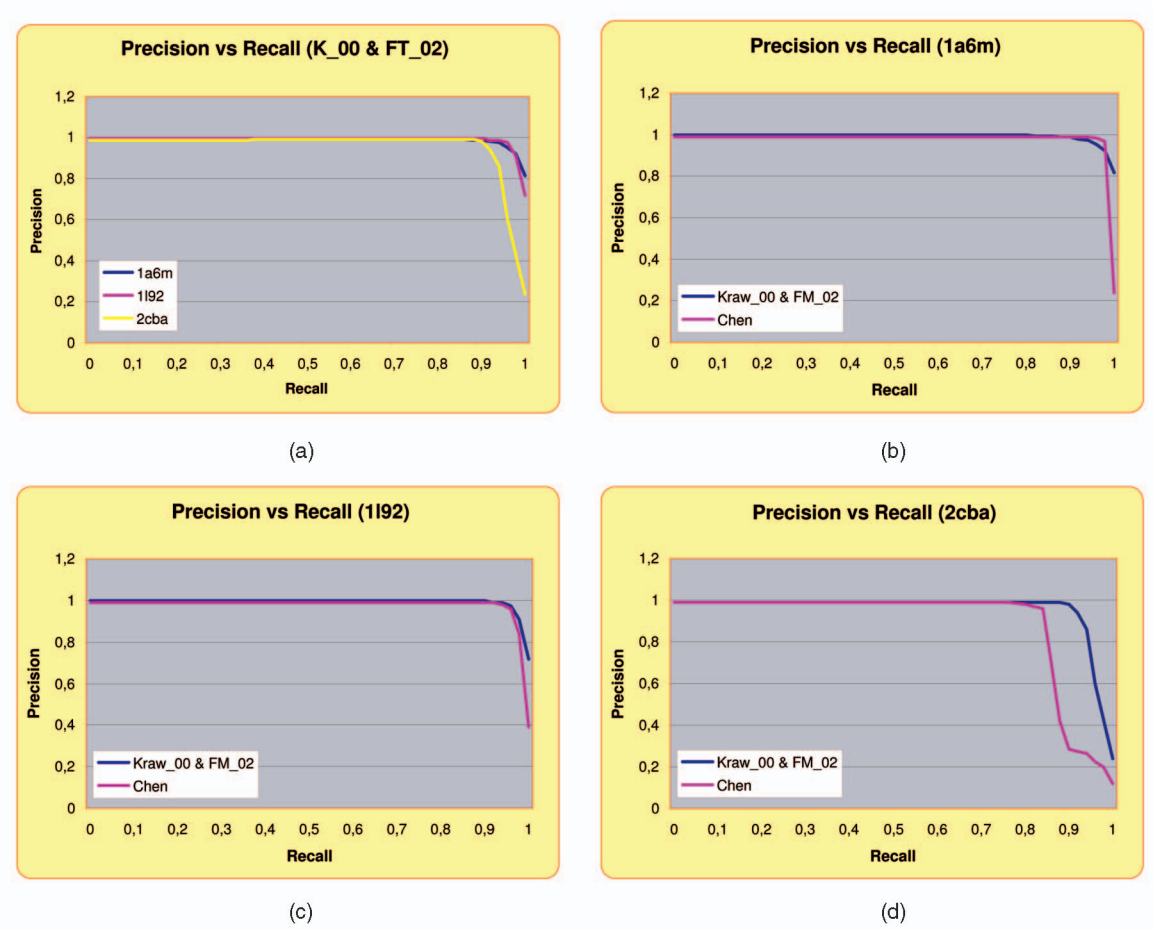

Fig. 12. (a) Precision-recall curve of classes 1a6m, 1l92, and 2cba by using $Kraw_{00}\&FT_{02}$ descriptors. (b), (c), and (d) Comparison of precision-recall curve for each class with the method presented in [16].

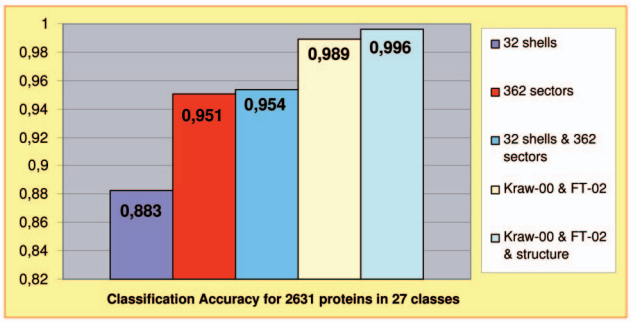

Fig. 13. Comparison of the proposed method with the one presented in [15] in terms of overall classification accuracy.

6 Conclusions

In this paper, a novel approach for the comparison of 3D protein structures is proposed. The approach consists of an offline and an online step. In the offline step, the protein, which is taken from a PDB file, is preprocessed in terms of visualization and triangulation. Next, the protein is translated, scaled, and voxelized. A set of functionals are applied to the volume of the 3D structure producing a new domain of concentric spheres. In this domain, a new set of functionals is applied, resulting in a completely rotation invariant descriptor vector. Additionally, descriptor vectors which correspond to the protein's primary and secondary structure are extracted as well. All these descriptor vectors are stored, along with the corresponding proteins. In the online step, a classification algorithm is followed for the descriptor vectors.

Experiments were performed evaluating the efficiency of the proposed method using as ground truth a portion of the FFSP/DALI database, in terms of overall classification accuracy and precision-recall. The proposed method, far less complex than the DALI algorithm, was seen to produce results very close to the ground truth when applied to single domain chains. For multidomain proteins, however, the experimental results are inconclusive.

Acknowledgments

This work was supported by the ALTAB23D project funded by the Greek Secretariat of Research and Technology and by the SIMILAR, CATER, and 3DTV EC IST projects.

References

[1] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne, "The Protein Data Bank," *Nucleic Acids Research,* vol. 28, pp. 235-242, 2000.

[2] J.L. Sussman, D. Ling, J. Jiang, N.O. Manning, J. Prilusky, O. Ritter, and E.E. Abola, "Acta Crystallogr.," vol. 54, pp. 1078-1084, 1998.

[3] C.B. Anfinsen, "Principles that Govern the Folding of Protein Chains," *Science,* vol. 181, pp. 223-230, 1973.

[4] 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," *J. Molecular Biology,* vol. 247, pp. 536-540, 1995.

[5] C.A Orengo, A.D. Michie, 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.

[6] L. Holm and C. Sander, "The FSSP Database: Fold Classification Based on Structure-Structure Alignment of Proteins," *Nucleic Acids Research,* vol. 24, pp. 206-210, 1996.

[7] L. Holm and C. Sander, "Touring Protein Fold Space with Dali/FSSP," *Nucleic Acids Research,* vol. 26, pp. 316-319, 1998.

[8] The European Bionformatics Institute, http://www.ebi.ac.uk/, 2006.

[9] A. Bairoch and R. Apweiler, "The SWISS-PROT Protein Sequence Databank and Its Supplement TrEMBL in 1998," *Nucleid Acids Research,* vol. 26, pp. 38-42, 1998.

[10] L. Falquet, M. Pagni, P. Bucher, N. Hulo, C.J. Sigrist, K. Hofmann, A. Bairoch, "The PROSITE Database, Its Status in 2002," *Nucleid Acids Research,* vol. 30, pp. 235-238, 2002.

[11] http://www.expasy.ch/prosite/, 2006.

[12] http://www.rcsb.org, 2006.

[13] F. Psomopoulos, S. Diplaris, P.A. Mitkas, "A Finite State Automata Based Technique for Protein Classification Rules Induction," *Proc. Second European Workshop Data Mining and Text Mining in Bioinformatics,* 2004.

[14] W.N. Grundy, T.L. Bailey, C.P. Elkan, and M.E. Baker, "Meta-MEME: Motif-Based Hidden Markov Models of Protein Families," *IEEE Trans. Computational and Applied Bioscience,* vol. 13, no. 4, pp. 397-406, Aug. 1997.

[15] M. Ankerst, G. Kastenmuller, H.P. Kriegel, and T. Seidl, "Nearest Neigbor Classification in 3D Protein Databases," *Proc. Seventh Int'l Conf. Intelligent Systems for Molecular Biology (ISMB '99),* 1999.

[16] C. Zhang and T. Chen, "Retrieval of 3D Protein Structures," *Proc. Int'l Conf. Image Processing,* Sept. 2002.

[17] C. Zhang and T. Chen, "Efficient Feature Extraction for 2D/3D Objects in Mesh Representation," *Proc. Int'l Conf. Image Processing,* vol. 3, pp. 935-938, Oct. 2001.

[18] C. Guerra, S. Lonardi, and G. Zanotti, "Analysis of Secondary Structure Elements of Proteins Using Indexing Techniques," *Proc. First Int'l Symp. 3D Data Processing Visualization and Transmission (3DPVT '02),* 2002.

[19] D. Zarpalas, P. Daras, D. Tzovaras, and M.G. Strintzis, "3D Model Search and Retrieval Using the Spherical Trace Transform," *IEEE Trans. Multimedia,* submitted.

[20] P.T. Yap, R. Paramesran, and S.H. Ong, "Image Analysis by Krawtchouk Moments," *IEEE Trans. Image Processing,* vol. 12, no. 11, pp. 1367-1377, Nov. 2003.

[21] M.K. Hu, "Visual Pattern Recognition by Moment Invariants," *IRE Trans. Information Theory,* vol. 8, pp. 179-197, 1962.

[22] D.W. Ritchie, "Parametric Protein Shale Recognition," PhD thesis, Univ. of Aberdeen, 1998.

[23] http://www.bioinfo.biocenter.helsinki.fi:8080/dali/index.html, 2006.

[24] P. Koehl, "Protein Structure Similarities," *Current Opinion in Structural Biology,* vol. 11, no. 3, pp. 348-353, June 2001.

[25] I.-G. Choi, J. Kwon, and S.-H. Kim, "Local Feature Frequency Profile: A Method to Measure Structural Similarity in Proteins," *Proc. Nat'l Academy of Science,* vol. 101, no. 11, pp. 3797-3802, Mar. 2004.

[26] S. Cheek, Y. Qi, S. SriKrishna, L.N. Kinch, and N.V. Grishin, "SCOPmap: Automated Assignment of Protein Structures to Evolutionary Superfamilies," *BMC Bioinformatics,* vol. 5, p. 197, 2004.

[27] J. Huan, W. Wang, A. Washington, J. Prins, R. Shah, and A. Tropsha, "Accurate Classification of Protein Structural Families Using Coherent Subgraph Analysis," *Proc. Pacific Symp. Biocomputing (PSB),* 2004.

[28] A. Dubey, S. Hwang, C. Rangel, C.E. Rasmussen, Z. Ghahramani, and D.L. Wild, "Clustering Protein Sequence and Structure Space with Infinite Gaussian Mixture Models," *Proc. Pacific Symp. Biocomputing,* 2004.

[29] L. Holm and C. Sander, "3-D Lookup: Fast Protein Structure Database Searches at 90% Reliability," *Proc. Third Int'l Conf. Intelligent Systems for Molecular Biology (ISMB),* pp. 179-187, 1995.

[30] S. Dua and N. Kandiraju, "A Novel Computational Framework for Structural Classification of Proteins Using Local Geometric Parameter Matching," *Proc. 2004 IEEE Computational Systems Bioinformatics Conf. (CSB 2004),* pp. 710-711, 2004.

[31] Y. Sun, M. Robinson, R. Adams, A.G. Rust, P. Kaye, and N. Davey, "Integrating Binding Site Predictions Using Meta Classification Methods," *Proc. Seventh Int'l Conf. Adaptive and Natural Computing Algorithms (ICANNGA 2005),* Mar. 2005.

[32] S. Tiwari and S. Gallager, "Machine Learning and Multiscale Methods in the Identification of Bivalve Larvae," *Proc. Ninth IEEE Int'l Conf. Computer Vision (ICCV 2003),* pp. 13-16, Oct. 2003.
[33] P. Daras, D. Zarpalas, D. Tzovaras, and M.G. Strintzis, "3D Model Search and Retrieval Based on the Spherical Trace Transform," *Proc. IEEE Int'l Workshop Multimedia Signal Processing (MMSP),* 2004.
[34] S.F. Altschul, T.L. Madden, A.A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D.J. Lipman, "Gapped Blast and PSI-Blast: A New Generation of Protein Database Search Programs," *Nucleic Acids Research,* vol. 25, no. 17, pp. 3389-3402, 1997.

Petros Daras received the Diploma in electrical and computer engineering, the MSc degree in medical informatics, and the PhD degree in electrical and computer engineering from the Aristotle University of Thessaloniki, Greece, in 1999, 2002, and 2005, respectively. He is an associate researcher at the Informatics and Telematics Institute. His main research interests include computer vision, search and retrieval of 3D objects, the MPEG-4 standard, peer-to-peer technologies, and medical informatics. He has been involved in more than 10 European and National research projects. Dr. Daras is a member of the Technical Chamber of Greece.

Dimitrios Zarpalas received the Diploma in electrical and computer engineering from the Aristotle University of Thessaloniki, Greece, in 2003. He is an associate researcher at the Informatics and Telematics Institute. His main research interests include search and retrieval of 3D objects and medical image processing. He is a member of the Technical Chamber of Greece.

Apostolos Axenopoulos received the Diploma in electrical and computer engineering from the Aristotle University of Thessaloniki, Greece, in 2003. Currently, he is pursuing the MSc degree in advanced computing systems at the Aristotle University of Thessaloniki. He is an associate researcher at the Informatics and Telematics Institute. His main research interests include 3D content-based search and retrieval. He is a member of the Technical Chamber of Greece.

Dimitrios Tzovaras received the Diploma in electrical engineering and the PhD degree in 2D and 3D image compression from Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1992 and 1997, respectively. He is a senior researcher in the Informatics and Telematics Institute of Thessaloniki. Prior to his current position, he was a senior researcher on 3D imaging at the Aristotle University of Thessaloniki. His main research interests include virtual reality, assistive technologies, 3D data processing, medical image communication, 3D motion estimation, and stereo and multiview image sequence coding. His involvement with those research areas has led to the coauthoring of more than 35 papers in refereed journals and more than 80 papers in international conferences. He has served as a regular reviewer for a number of international journals and conferences. Since 1992, he has been involved in more than 40 projects in Greece funded by the EC and the Greek Secretariat of Research and Technology. He is an associate editor of the *EURASIP Journal of Applied Signal Processing* and a member of the Technical Chamber of Greece.

Michael Gerassimos Strintzis (M'70-SM'80-F'04) received the Diploma in electrical engineering from the National Technical University of Athens, Athens, Greece, in 1967, and the MA and PhD degrees in electrical engineering from Princeton University, Princeton, New Jersey, in 1969 and 1970, respectively. He then joined the Electrical Engineering Department at the University of Pittsburgh, where he served as an assistant professor (1970-1976) and an associate professor (1976-1980). Since 1980, he has been a professor of electrical and computer engineering at the University of Thessaloniki, Thessaloniki, Greece, and, since 1999, director of the Informatics and Telematics Research Institute, Thessaloniki. His current research interests include 2D and 3D image coding, image processing, biomedical signal and image processing, and DVD and Internet data authentication and copy protection. Dr. Strintzis has served as associate editor for the *IEEE Transactions on Circuits and Systems for Video Technology* since 1999. In 1984, he was awarded one of the Centennial Medals of the IEEE. He is a fellow of the IEEE.