Available online at www.sciencedirect.com
ScienceDirect

Pattern Recognition 40 (2007) 2437–2452

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/pr

Efficient 3D shape matching and retrieval using a concrete radialized spherical projection representation

Panagiotis Papadakis[a,b], Ioannis Pratikakis[a,*], Stavros Perantonis[a], Theoharis Theoharis[b]

[a]Computational Intelligence Laboratory, Institute of Informatics and Telecommunications, National Center for Scientific Research "Demokritos", 15310 Agia Paraskevi, Greece

[b]Computer Graphics Group, Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, 15784 Panepistimiopolis, Athens, Greece

Received 14 July 2006; received in revised form 13 December 2006; accepted 19 December 2006

Abstract

We present a 3D shape retrieval methodology based on the theory of spherical harmonics. Using properties of spherical harmonics, scaling and axial flipping invariance is achieved. Rotation normalization is performed by employing the continuous principal component analysis along with a novel approach which applies PCA on the face normals of the model. The 3D model is decomposed into a set of spherical functions which represents not only the intersections of the corresponding surface with rays emanating from the origin but also points in the direction of each ray which are closer to the origin than the furthest intersection point. The superior performance of the proposed methodology is demonstrated through a comparison against state-of-the-art approaches on standard databases.

Keywords: 3D shape retrieval; Spherical harmonics; Descriptor's invariance

1. Introduction

3D models have become an important part in modern computer graphics applications, such as computer-aided design, game development and film production while in several scientific domains, such as molecular biology, medicine, computational geometry and computer vision, a large part of scientific data is 3D data. The rapid evolution in graphics hardware and software development which has greatly facilitated 3D model acquisition, creation and manipulation, has given the opportunity to experience applications using 3D models not only to specialized users of the scientific community and the industrial domain, but also to common users. As the number of 3D models is continuously growing and thousands of models are already available from public and proprietary databases, the problem in creating new 3D models may shift to the problem of searching for existing 3D models. Instead of creating 3D models from scratch, a more convenient and profitable approach is the use of existing 3D models. Thereupon, the development of efficient search mechanisms is needed for the retrieval of 3D models from large repositories.

3D model search and retrieval could be performed by using a textual description of the user's target which identifies the semantic meaning of the desired model or class of models. In this case, the user would explicitly describe the target, but such an approach is sensitive to the user's subjectivity factor which is not necessarily in agreement with the textual information which has been annotated to the target. Furthermore, this method is problematic as it requires individually annotating every model of a repository which is impractical due to the huge and continuously increasing number of existing 3D models. Therefore, content-based 3D shape retrieval methods are suited for search since they do not require any annotation while. They only require robust 3D shape feature extraction that can be applied automatically. In these methods, a *shape descriptor* is computed which represents the model and is consequently

[*] Corresponding author. Tel.: +30 210 650 3183; fax: +30 210 653 2175.
E-mail addresses: ppapadak@iit.demokritos.gr, ppapadak@di.uoa.gr (P. Papadakis), ipratika@iit.demokritos.gr (I. Pratikakis), sper@iit.demokritos.gr (S. Perantonis), theotheo@di.uoa.gr (T. Theoharis).

0031-3203/$30.00 © 2007 Pattern Recognition Society. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2006.12.026

used at the matching stage. When 3D model comparison is performed, it is desired that shape descriptors are invariant under geometrical transformations such as translation, scaling, rotation and reflection. Thereafter, the discriminative power of these methods is highly affected by the invariance to these transformations, while extraction and comparison time also affect the performance, especially for real-time applications.

In this paper, we describe a 3D shape descriptor based on the theory of spherical harmonics. Translation invariance is achieved by using continuous principal component analysis (CPCA). The rotation invariance problem is alleviated by applying PCA on the face normals of the model (NPCA) and CPCA on the surface points of the model. The 3D model is decomposed into a set of spherical functions which represents not only the intersection points of the model's surface with rays emanating from the origin but also all points in the direction of each ray that are closer to the origin than the furthest intersection point. Spherical functions are then expressed by spherical harmonic coefficients by applying the spherical harmonics transform individually to each spherical function. Scaling and axial flipping invariance is achieved in the last stage of the shape descriptor's extraction, by using properties of spherical harmonic coefficients. The novelty of the proposed approach consists of the following: (i) Application of PCA for rotation normalization by using the face normals of the model and parallel use of CPCA on the surface points of the model; (ii) Scaling invariance is achieved by scaling the spherical functions proportionally to the model's size and by scaling the corresponding spherical harmonic coefficients to the unit L_1 norm; (iii) A new shape descriptor using spherical functions that represent not only the intersection points of the model's surface with rays emanating from the origin but also all points in the direction of each ray that are closer to the origin than the furthest intersection point.

The remaining of this paper is organized as follows: in Section 2, the state-of-the-art in the area of 3D model retrieval methods is presented. We summarize the approaches used to achieve rotation invariance in Section 3 and describe the proposed 3D shape descriptor. In Section 4, we present performance results and a comparison against start-of-the-art retrieval methods. Finally, in Section 5 conclusions and future work are discussed.

2. Related work

In this section we give an overview of existing start-of-the-art 3D shape retrieval methods. We categorize these methods into the following: (i) spherical function based; (ii) 2D image-projection based; (iii) volumetric features based; (iv) hierarchical structure based; (v) shape distributions based and (vi) hybrid methods.

In the first category, 3D space is parameterized using spherical functions. A popular example is the shape histograms descriptor proposed by Ankerst et al. [1], which was developed primarily to classify proteins. In this method, 3D space can be divided into concentric shells, sectors, or both. Dividing 3D space into both sectors and shells gives the best results. For each part of the subdivision the model's shape distribution is computed, giving a sum of histogram bins which comprises the final shape descriptor. In [2] Koertgen et al. make use of 3D shape contexts which are histograms each one identifying a surface point by the relative coordinates of the remaining surface points. Vranic et al. [3] proposed the ray-based descriptor which characterizes a 3D model by a spherical extent function capturing the furthest intersection points of the model's surface with rays emanating from the origin. Spherical harmonics and moments representation of the spherical extent function were tested in [4], indicating better performance in the first case. In the same context, a more general approach was introduced by Vranic et al. [5,6] known as the layered depth spheres descriptor (also known as radialized spherical extent function). Here, the 3D model is described by a spherical function which decomposes the model into a sum of concentric shells and gives the maximal distance of the model from the center of mass as a function of angle and the radius of the equivalent shell. The spherical function is represented by spherical harmonics coefficients. The extended Gaussian image (EGI) proposed by Horn et al. [7] has also been used as a shape representation method, where a spherical function correlates the faces of a 3D model with values on a sphere of radius 1 (Gaussian sphere). Specifically, the Gaussian sphere is discretized into a finite set of points. Then, each point is assigned a value proportional to the surface area of the model having the same direction as the unit vector defined by the specific point. A variation of this basic idea is the complex EGI proposed by Kang et al. [8] where each EGI cell is described by a complex number which captures both rotation and translation information. Novotni et al. [9] presented the 3D Zernike descriptor, generalizing the 2D Zernike descriptor. The descriptor is an extension of spherical harmonics descriptors because Zernike functions are spherical harmonic functions modulated by appropriate radial functions. Norms of vectors consisting of Zernike moments are used to comprise the shape descriptor in order to achieve rotation invariance. In [10], Daras et al. introduces the generalized radon transform, where the descriptor is extracted by combining the radial integration transform which captures the radial information of a 3D model and the spherical integration transform which captures its spherical information.

In the second category of 3D shape retrieval methods, 2D images-projections of a 3D model are used for feature extraction and comparison. Chen et al. [11] proposed the light field descriptor, which is presented in the literature as the method with the best discriminative power so far, but having long extraction and comparison times as a disadvantage. The shape descriptor is generated by projecting the 3D model uniformly from the vertices of a dodecahedron and the distance between two models is computed by the minimal L_1 distance between all pairs of vertices under all possible rotations of the two dodecahedrons. Each projection is composed of a combined feature vector of Zernike moments and Fourier coefficients. Another method where 2D shape matching can support 3D shape matching is the depth-buffer descriptor presented in [5] by Vranic, where six projections of the model are created, one for each side of a cube which encloses the model, storing the maximum and minimum depth values of the model at each side.

The six depth buffers can then be used to produce shape descriptors in the spatial domain or in the spectral domain. In the same work, the silhouette-based descriptor is proposed, which is produced by projecting the 3D model to the Cartesian planes, leading to three silhouettes. The contours of the silhouettes are computed by two alternative approaches and are used accordingly at the matching stage. Ohbuchi et al. [12] proposed the multiple orientation depth Fourier transform descriptor. In this method, the model is projected from 42 viewpoints to cover all possible view aspects of a 3D model. The respective depth buffers are then transformed to the r–θ coordinate system and a Fourier transform is applied, producing a set of coefficients for each depth buffer. To compare two models, all possible pairs of coefficients are compared which inevitably increases comparison time. Zarpalas et al. [13] introduced a 3D shape descriptor called the spherical trace transform, which is an extension to 3D shape matching of the 2D trace transform. Among a variety of 2D features which are computed over a set of planes intersecting the volume of a 3D model, one of the components of the shape descriptor are 2D Krawtchouk moments which improve significantly the overall performance of the proposed method.

In the case of methods that belong to the third category, initially the 3D model is preprocessed to obtain a volumetric representation which is then used at the feature extraction stage. This is done by dividing 3D space into volume elements (voxels), and then assigning to each voxel a value according to a function which considers the model's relative position. Many variations exist concerning the method used to decide what value is to be assigned to each voxel. The Gaussian (exponentially decaying) Euclidean distance transform proposed by Kazhdan et al. [14] (GEDT) is one of best performing variations. The GEDT descriptor can be expressed by norms of spherical harmonic frequencies which give a rotation invariant representation, reducing the shape descriptor's storage cost and comparison time. The reflective symmetry descriptor proposed by Kazhan et al. [15], computes a voxelized representation of a 3D model which is then used to compute the reflective symmetries of the model with respect to all planes passing through its center of mass. Suzuki et al. [16], introduced a shape descriptor based on equivalence classes. After enclosing the model in a 3D grid, cells which follow the same paths under 90° rotations are organized into sets called equivalence classes. Each of these classes is represented by a value giving the density of the initial point cloud inside the cells of the given class. Tangelder et al. [17], make use of weighted point sets. The set of points is selected by a 3D grid which encloses the model and three methods are proposed to select a point at each non-empty grid cell and assign a weight to the specific point. For the comparison of two shape signatures a variation of the earth mover's distance is adopted.

The fourth category consists of methods which use hierarchical 3D model representations based on skeleton graphs. Hilaga et al. [18] introduced the multiresolution Reeb graph, which represents a 3D model's topology and skeletal structure at various levels of detail. In [19] Zhang et al. consider the use of medial surfaces to compute an equivalent directed acyclic graph of a voxelized model. Sundar et al. [20] also proposed skeleton based shape matching. They first obtain the volume of a 3D model which passes through a thinning process producing a set of skeletal points, which finally form a directed acyclic graph by applying the minimum spanning tree algorithm. Cornea et al. [21], propose the use of curve skeletons produced by the application of the generalized distance field to the volumetric representation of a 3D model. Matching between two skeletons is performed by using the earth mover's distance. Generally, graph-based methods for 3D model matching are mostly useful for matching articulated models but are problematic in the use of efficient similarity measures.

Methods based on statistical measures belong to the last category. The shape distributions proposed by Osada et al. [22,23] is a shape descriptor where several shape characteristics can be measured for a random set of points belonging to the model, according to the selection of an appropriate shape function. There are variations of this method according to the characteristic that is measured, e.g. in the D2 shape distribution descriptor the distance between two random surface points is computed. Ohbuchi et al. [24], proposed enhanced shape functions, namely the angle–distance histogram or the absolute angle–distance histogram for inconsistently oriented meshes, which are extensions of the D2 shape distribution descriptor and have better performance. In [25], the same author proposes a combination of the absolute angle–distance function with the alpha shape representation of 3D models. Zaharia et al. [26] presented the 3D shape spectrum descriptor which is within the MPEG-7 framework. This shape representation is the histogram of the shape index values calculated for the entire mesh of a 3D model, where each shape index value is the angular representation of the vector composed of the first and second principal curvature at a specific surface point.

Finally, the hybrid descriptor proposed by Vranic [5] is a combination of three different methods. The depth buffer, silhouette and ray-based descriptors described earlier, are used together into one descriptor trying to capture as much discriminating information for a 3D model as possible.

3. The proposed methodology

3.1. The overall scheme

In this section, we provide an overview of the proposed methodology that will be detailed in Sections 3.2–3.4. The main steps for the 3D shape descriptor's extraction procedure as well as the matching scheme are depicted in Fig. 1.

Initially, the model is translated so that its center of mass coincides with the origin, as shown in Fig. 1(a). Then we apply in parallel two different alignment methods, namely CPCA and the PCA on the normals of the model's faces (NPCA), to alleviate the problem of rotation invariance. This gives two aligned versions of the translated model (Fig. 1(b)) which are hereafter processed in the same way and will finally give two 3D shape descriptors. In the next stage, each alignment variant is expressed by a set of spherical functions $f_1, f_2, \ldots, f_N$ with increasing radius giving the intersections of the model's

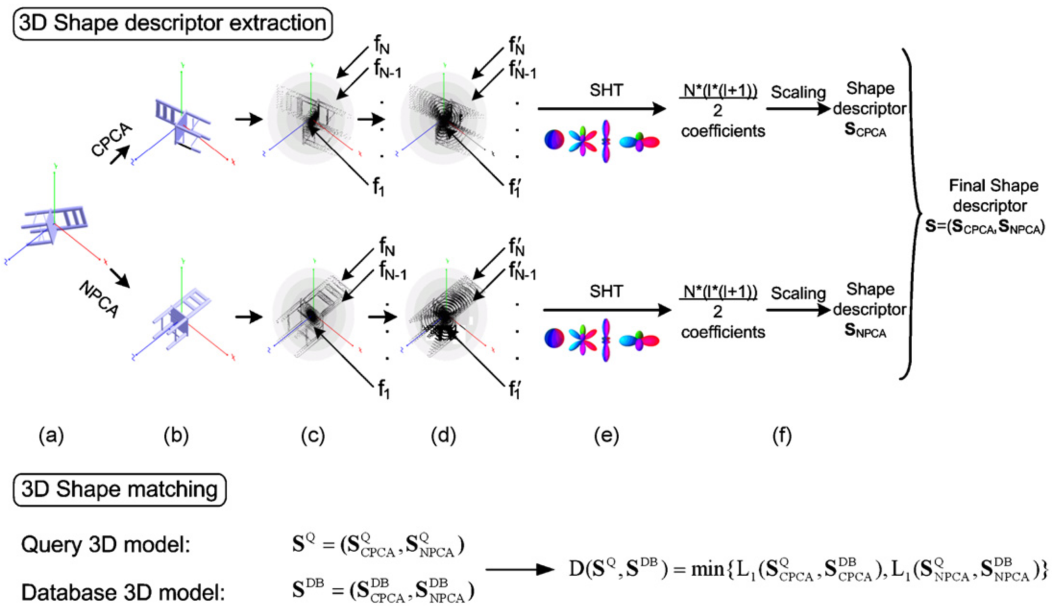
3D Shape descriptor extraction
CPCA
NPCA
f_N
f_{N-1}
f_1
f'_N
f'_{N-1}
f'_1
SHT
$\frac{N*(l*(l+1))}{2}$ coefficients
Scaling
Shape descriptor S_{CPCA}
Shape descriptor S_{NPCA}
Final Shape descriptor $S=(S_{CPCA}, S_{NPCA})$
(a)
(b)
(c)
(d)
(e)
(f)
3D Shape matching
Query 3D model: $S^Q = (S^Q_{CPCA}, S^Q_{NPCA})$
Database 3D model: $S^{DB} = (S^{DB}_{CPCA}, S^{DB}_{NPCA})$
$D(S^Q, S^{DB}) = \min\{L_1(S^Q_{CPCA}, S^{DB}_{CPCA}), L_1(S^Q_{NPCA}, S^{DB}_{NPCA})\}$

Fig. 1. The stages of the proposed 3D shape matching scheme for retrieval.

surface with rays emanating from the origin (Fig. 1(c)). This is equivalent to producing spherical projections of approximately equidistant parts of the model at different radii. In the sequel, the set of spherical functions is processed to find the furthest intersection points from the origin on each ray. If the model is viewed in the direction of a specific ray, then the furthest intersection point obscures the part of the model along the ray which is closer to the origin. We may assume that the obscured part belongs to the 3D model because we perceive a 3D model as a concrete entity and not as a 3D surface composed of a set of faces. This may not be true for concave objects but our experiments showed that this does not influence the discriminative power of the shape descriptor. Thereafter, these parts are considered to belong to the model giving a new 3D model representation (Fig. 1(d)). For each spherical function, the spherical harmonics transform is applied producing a set of complex coefficients (Fig. 1(e)). In the last stage, spherical harmonic coefficients are scaled to the unit L_1 norm (Fig. 1(f)) which guarantees scaling invariance and properties of spherical harmonics are used to achieve axial flipping invariance. From here on, we will call the proposed descriptor the *concrete radialized spherical projection* (*CRSP*) descriptor.

Each 3D model is represented by the concatenation two shape descriptors corresponding to the aligned versions using CPCA and NPCA and of a set of complex spherical harmonic coefficients. To compare two shape descriptors, the L_1 distance between the corresponding parts is computed and the minimum distance is taken as the similarity measure between the two models. The notion of taking the minimum distance is based on the expectation that the best establishment of correspondences between two models is achieved, when the difference between the corresponding shape descriptors is minimum.

3.2. Translation and rotation normalization

In this section, we detail how we address the problem of translation and rotation invariance. Prior to the translation and rotation normalization procedures, a preprocessing step is applied to obtain a representation of a 3D model, consisting only of triangles. A double triangle check is also performed to find identical triangles within the polygonal model. These triangles are recognized as having the same vertices but appearing in different order within a 3D model file (e.g. the triangles with vertex indices (10, 422, 87) and (422, 10, 87) are identical). We found a large number of such triangles within the 3D model datasets tested in Section 4. These triangles cannot be identified when a 3D model is rendered, but if they are taken into account they may affect the model's translation, rotation and scale normalization.

The model is translated to its center of mass $\mathbf{m}$ which is computed using CPCA [5,27]. If $\mathbf{v} = [v_x, v_y, v_z]^{\mathrm{T}}$ is a surface point, then using barycentric coordinates, $\mathbf{v}$ is given by

$$\mathbf{v} = a \cdot \mathbf{A}_i + b \cdot \mathbf{B}_i + (1 - a - b) \cdot \mathbf{C}_i, \tag{1}$$

where $\mathbf{A}_i$, $\mathbf{B}_i$ and $\mathbf{C}_i$ are the vertices of triangle i, $i = 1, 2, \ldots, N$, where N is the number of triangles of the model and a, b are the barycentric coordinates. Thus $\mathbf{m}$ is equal to:

$$\mathbf{m} = \frac{1}{E} \int\int_{\mathbf{v} \in T_i} \mathbf{v}\, \mathrm{d}s = \frac{1}{E} \sum_{i=1}^{N} E_i \cdot \frac{(\mathbf{A}_i + \mathbf{B}_i + \mathbf{C}_i)}{3}, \tag{2}$$

where E is the total surface area of the model and E_i is the surface area of triangle T_i. The detailed derivation of Eq. (2) is given in Appendix A.

When a shape descriptor is based on the establishment of correspondences between models, the rotation invariance property is of great importance. Translation and scaling invariance are also important and a shape descriptor can be invariant under these transformations using efficient methods. However, to achieve rotation invariance using a single method for a wide range of model classes, has proven to be a very complicated problem. This is because it is not obvious which shape characteristic we should choose and the way it should be used when normalizing for rotation, so that it is suitable for every class. Several approaches have been used belonging to one of the following categories: (i) alignment and (ii) rotation invariance by definition.

In category (i), each model is normalized for rotation by rotating the model into a canonical coordinate frame with the expectation that when comparing models of the same class these models will be similarly aligned. However, as mentioned previously, 3D models are spread over a wide variety of classes, making it very difficult to find an alignment method for models of every possible class. The prominent alignment method is PCA, also known as the Karhunen–Loeve transform [28]. The PCA algorithm generates mutually uncorrelated features taking a set of vectors as input. The set of vertices of the polygonal model [3] or the centroids of the model's triangles [29] have been used as input to the PCA. In [5,27] the so-called CPCA is presented. Another approach to normalize a 3D model for rotation (similar to PCA), is the use of singular value decomposition (SVD) [28]. In [22,30], the SVD of the covariance matrix of the 3D model is computed and the unitary matrix is applied to the model for rotation normalization.

In the category (ii), rotation invariance is achieved by the definition of the shape descriptor. These descriptors are invariant under rotation, but they may discard information regarding the 3D model which can be discriminative. Descriptors based on spherical harmonics, Zernike moments and shell histograms are examples of representation methods able to achieve rotation invariance by definition.

Instead of just using one method to achieve rotation invariance, combinations of different approaches may be considered. This may improve the performance of a shape descriptor at the cost of increased complexity.

To alleviate the rotation invariance problem, we apply two alternative alignment procedures in parallel. In the first procedure, we use CPCA to compute the covariance matrix $\mathbf{C}$ of the model's surface, as in the following:

$$\mathbf{C} = \frac{1}{E}\int\int_{\mathbf{v}\in M}(\mathbf{v}-\mathbf{m})(\mathbf{v}-\mathbf{m})^{\mathrm{T}}\,\mathrm{d}s$$
$$= \frac{1}{E}\sum_{i=1}^{N}\int\int_{\mathbf{v}\in T_i}(\mathbf{v}-\mathbf{m})(\mathbf{v}-\mathbf{m})^{\mathrm{T}}\,\mathrm{d}s$$
$$\Rightarrow \mathbf{C} = \frac{1}{12E}\sum_{i=1}^{N}E_i\cdot[\mathbf{f}(\mathbf{A}_i)+\mathbf{f}(\mathbf{B}_i)+\mathbf{f}(\mathbf{C}_i) + 9\cdot\mathbf{f}((\mathbf{A}_i+\mathbf{B}_i+\mathbf{C}_i)/3)], \qquad (3)$$

where M is the set of vertices of the polygonal model and $\mathbf{f}(\mathbf{v}) = (\mathbf{v}-\mathbf{m})(\mathbf{v}-\mathbf{m})^{\mathrm{T}}$. A detailed derivation of Eq. (3) is given in Appendix A; this is not included in [5,27] where CPCA was introduced.

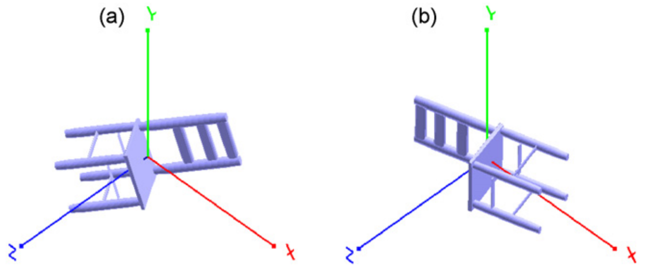

Fig. 2. (a) Initial model; (b) aligned model using the continuous principal component analysis.

The eigenvectors of $\mathbf{C}$ are then computed, which represent the principal directions of the model's surface area and form an orthonormal basis for $\mathscr{R}^3$ space. These vectors are then sorted according to the order of descending (or ascending) eigenvalues and are finally normalized to the Euclidean unit norm. A rotation matrix $\mathbf{R}$ is formed having as columns the scaled eigenvectors in decreasing (or ascending) order and $\mathbf{R}$ is applied to the model so that the principal directions of the model coincide with the coordinate axes. After the application of $\mathbf{R}$, the model is rotated to a canonical position as shown in Fig. 2.

In the second alignment procedure we compute the covariance matrix by using NPCA. If $\mathbf{n}_i$ is the normal vector of triangle T_i and $\mathbf{m}_{Nl}$ is the average face normal, then the covariance matrix is computed as

$$\mathbf{C} = \frac{1}{2E}\sum_{i=1}^{N}(E_i(\mathbf{n}_i-\mathbf{m}_{Nl})\cdot(\mathbf{n}_i-\mathbf{m}_{Nl})^{\mathrm{T}} + E_i(-\mathbf{n}_i-\mathbf{m}_{Nl})\cdot(-\mathbf{n}_i-\mathbf{m}_{Nl})^{\mathrm{T}}). \qquad (4)$$

Each triangle is counted twice, using its normal and opposite to the normal direction. This is done to avoid possible variations in vertex ordering at the model triangles, that affects the direction's sign and because we may assume that every triangle has two sides with opposite directions. Adopting this, we observe that the average face normal $\mathbf{m}_{Nl}$ becomes

$$\mathbf{m}_{Nl} = \frac{1}{2E}\sum_{i=1}^{N}(E_i\cdot\mathbf{n}_i + E_i\cdot(-\mathbf{n}_i)) = \mathbf{0}. \qquad (5)$$

The above equation indicates that $\mathbf{m}_{Nl}$ is always the zero vector, thus $\mathbf{C}$ is now given by

$$\mathbf{C} = \frac{1}{2E}\sum_{i=1}^{N}(E_i\cdot\mathbf{n}_i\cdot\mathbf{n}_i^{\mathrm{T}} + E_i\cdot(-\mathbf{n}_i)\cdot(-\mathbf{n}_i^{\mathrm{T}}))$$
$$= \frac{1}{E}\sum_{i=1}^{N}E_i\cdot\mathbf{n}_i\mathbf{n}_i^{\mathrm{T}}. \qquad (6)$$

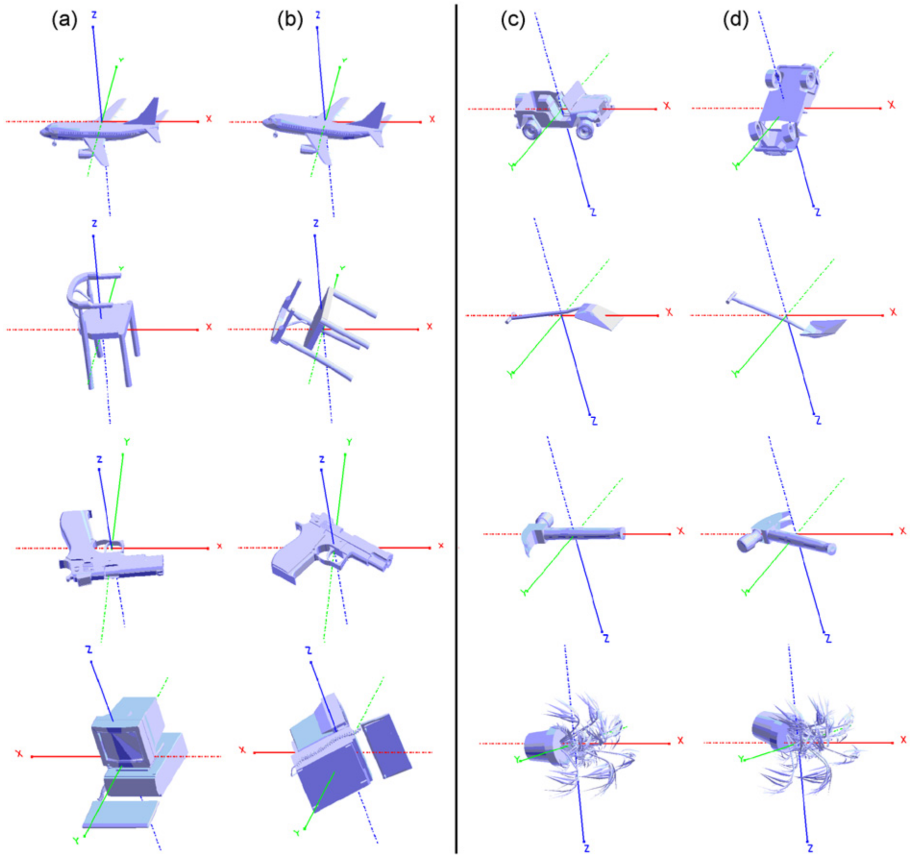

Fig. 3. (a), (d) columns: alignment using NPCA; (b), (c) columns: alignment using CPCA. The objects of columns (a), (b) are better aligned using NPCA while those of columns (c), (d) are better aligned using CPCA.

After the computation of **C**, the PCA algorithm is applied to find and align the principal axes of the model with the coordinate axes. Our experiments indicate that there are models where NPCA gives a more consistent alignment than CPCA and vice versa. To justify this, we give some examples in Fig. 3. Motivated by this observation, we use both methods, expecting to achieve better overall results. This is justified by the experimental results given in Section 4. After completion of this stage we have two alignment variants of the initial model which will finally lead to two 3D shape descriptors.

3.3. Representation of a polygonal model as a set of spherical functions

After alignment, the model's surface is represented by a set of spherical functions.

A spherical function describes a bounded area of the model, defined by a lower and an upper radius which delimit a spherical shell. This shell is the volume in which the underlying surface of the model is represented by the equivalent spherical function points. The region corresponding to the rth spherical function is the spherical shell defined in the interval $[l_r, l_{r+1})$ where the l_r bound is given by

$$l_r = \begin{cases} 0, & r = 1, \\ (r-0.5)\cdot M \cdot d_{\mathrm{avg}}/N, & 1 < r \leqslant N+1, \end{cases} \tag{7}$$

where M is a weighting factor which determines the radius of the largest sphere (we set $M=3$), N is the number of spherical functions (we set $N=24$) and d_{avg} is the average distance of the model's surface from the center of mass. Multiplying by $M=3$ ensures that the model is adequately enclosed by the sphere with the largest radius and only outlying parts of the model may be excluded from the furthest boundary. The average distance is computed as

$$d_{\mathrm{avg}} = \sqrt{C_{11}+C_{22}+C_{33}}, \tag{8}$$

where C_{ij} is the (i, j)th element of the covariance matrix, computed using CPCA and by definition, C_{11}, C_{22}, and C_{33} represent the variance of the model's surface from the yz, xz and xy planes, respectively. By multiplying the boundaries of the spherical function with d_{avg} in Eq. (7), the spherical functions become scaling invariant in the sense that they are scaled proportionally to the original model's size. The boundaries of the spherical functions as well as d_{avg} and $M * d_{\mathrm{avg}}$ are shown in Fig. 4.

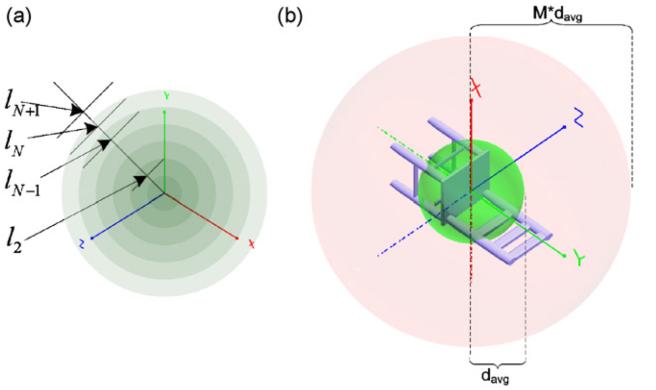

Fig. 4. (a) Visualization of the boundaries of the spherical functions and (b) visualization of d_{avg} and $M * d_{\text{avg}}$ values.

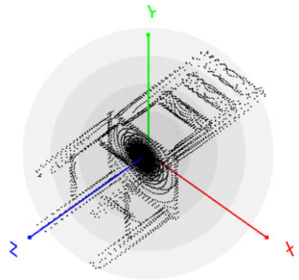

Fig. 5. Representation of a polygonal model as a set of spherical functions representing the intersections of the model's surface with rays emanating from the origin.

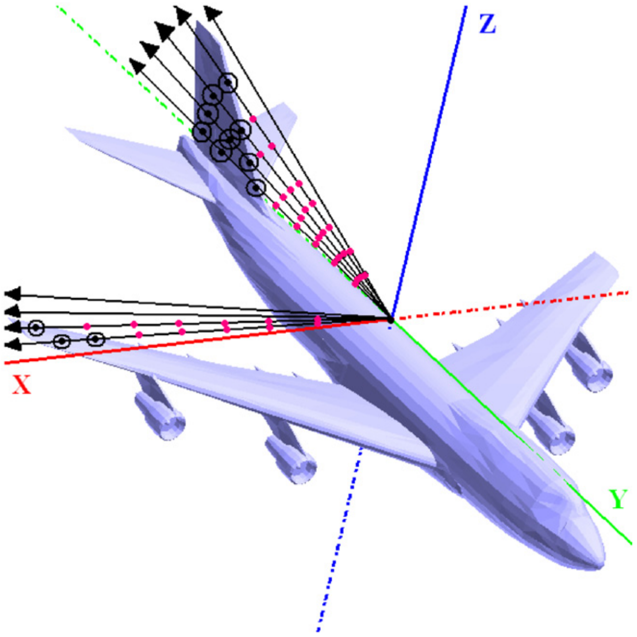

Fig. 6. Intersections of a model's surface with rays emanating from the origin (circled dots) and points closer to the origin than the furthest intersection point at each ray (single dots).

We compute the intersections of the triangles of the aligned model with rays emanating from the origin in the (θ_j, φ_k) directions denoted as ray(θ_j, φ_k), where $\theta_j = (2j+1)\pi/(4B)$, $\varphi_k = 2\pi k/(2B)$, $j, k = 0, 1, \ldots, 2B-1$ and B is the sampling bandwidth. Inters(i, θ_j, φ_k) denotes the distance from the origin of the ith intersection of ray(θ_j, φ_k) with the model's surface. If ray(θ_j, φ_k) does not intersect the model's surface then Inters$(0, \theta_j, \varphi_k) = 0$.

Let mxin(r, θ_j, φ_k) be the distance of the furthest intersection point along ray(θ_j, φ_k), within the boundaries of the rth spherical function, given by

$$\text{mxin}(r, \theta_j, \varphi_k) = \max\{\text{Inters}(i, \theta_j, \varphi_k) \,|\, l_r \leqslant \text{Inters}(i, \theta_j, \varphi_k) < l_{r+1}, i = 0, 1, \ldots\}. \tag{9}$$

The 3D model is represented as a set of N spherical functions $f_r : S^2 \rightarrow [l_r, l_{r+1}) \cup \{0\}, r = 1, 2, \ldots, N$, where S^2 denotes the points (θ_j, φ_k) on the unit sphere and f_r is given by

$$f_r(\theta_j, \varphi_k) = \text{mxin}(r, \theta_j, \varphi_k). \tag{10}$$

The decomposition of the 3D model into the spherical functions $f_r(\theta, \varphi)$ is similar to projecting approximately equidistant parts of the 3D model to concentric spheres of increasing radius. The set of spherical functions representing the 3D model as given by Eq. (10) is depicted in Fig. 5, where for each spherical function point $f_r(\theta_j, \varphi_k) \neq 0$, a black dot is used to represent the equivalent intersection.

In the following, the spherical functions are expanded to include additional information. Let mxin$(D, \theta_j, \varphi_k) \neq 0$ (corresponding to the furthest intersection point on ray(θ_j, φ_k)), be assigned to the spherical function point $f_D(\theta_j, \varphi_k)$. Then all $f_r(\theta_j, \varphi_k)$, for $r < D$ are to:

$$f_r(\theta_j, \varphi_k) = r \cdot M \cdot d_{\text{avg}}/N \quad \forall r < D. \tag{11}$$

Extra points are thus included in the spherical functions as depicted in Fig. 6. Practically, if the model is viewed from the (θ_j, φ_k) direction then the part of the model along ray(θ_j, φ_k) which is closer to the origin than mxin(D, θ_j, φ_k) is occluded by the intersection point corresponding to mxin(D, θ_j, φ_k). We may assume that the occluded part actually belongs to the model because we perceive 3D models as concrete entities. Adopting this assumption, a more concrete representation of the 3D model is produced which highly increases discriminative power as shown in Section 4. Therefore, taking into account Eqs. (10), (11), the new spherical functions f'_r are

$$f'_r(\theta_j, \varphi_k) = \begin{cases} \text{mxin}(r, \theta_j, \varphi_k), & \text{if } \text{mxin}(D, \theta_j, \varphi_k) \neq 0 \\ & \quad \text{and } r = D, \\ r \cdot M \cdot d_{\text{avg}}/N, & \text{if } \text{mxin}(D, \theta_j, \varphi_k) \neq 0 \\ & \quad \text{and } r < D, \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

These spherical functions give a model representation as shown in Fig. 7, where black dots are used to illustrate spherical function points $f'_r(\theta_j, \varphi_k) \neq 0$.

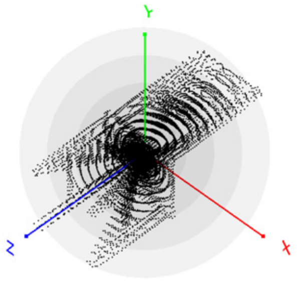

Fig. 7. Representation of a model as a set of spherical function points representing the intersections of the model's surface and occluded parts under specific viewpoints.

In the sequel, we expand the N spherical functions to their spherical harmonic representation. A detailed description of the spherical harmonics' formulation is given in Appendix B. Thus, having formed N spherical functions, the spherical harmonics transform (SHT) (Appendix B, Theorem 1) is applied for each function. For the computation of SHT, some researchers have used the SpharmonicKit [31] software setting $B=64$ for the sampling bandwidth as the best choice. However this software had the limitation of computing transformations of only $B=2^n$. Instead, we used the S2kit [31] software which enables transformations at any bandwidth and we set $B=40$ without reducing the discriminative power. As a result, the shape descriptor's extraction time dropped significantly and the number of rays dropped from $2*64*2*64=16\,384$ to $2*40*2*40=6400$.

By applying the SHT, B^2 complex coefficients are produced for each spherical function. Higher order coefficients represent finer details of the model as well as possible noise. Instead of taking into account all B^2 coefficients, we used the first L_{bands} of coefficients (up to the lth degree, we set $L_{\text{bands}}=16$). For all N spherical functions there are $N*L^2_{\text{bands}}$ coefficients, where $L_{\text{bands}} \leqslant B$. Using Theorem 1 (Appendix B) and the fact that $f'_r(\theta_j, \varphi_k)$ is a real-valued function, the following property holds between the coefficients of positive and negative order:

$$\hat{f}(l,m) = (-1)^m \overline{\hat{f}(l,-m)}, \tag{13}$$

where $\overline{x}$ denotes the complex conjugate of x. This means that only the positive order coefficients, which are $N*(L_{\text{bands}}*(L_{\text{bands}}+1))/2$ in total, are necessary to find the L_1 distance between two shape descriptors as will be proven in Section 3.4. We next show that the shape descriptor is invariant to possible axial flipping using the spherical harmonics' properties below:

(i) if the model is reflected with respect to the x-axis (yz plane) then: $\hat{f}(l,m) \Rightarrow \overline{\hat{f}(l,m)}$;
(ii) if the model is reflected with respect to the y-axis (xz plane) then: $\hat{f}(l,m) \Rightarrow (-1)^{l+m}\hat{f}(l,m)$;
(iii) if the model is reflected with respect to the z-axis (xy plane) then: $\hat{f}(l,m) \Rightarrow (-1)^m \overline{\hat{f}(l,m)}$.

These properties indicate that only the sign of the real and imaginary part of a coefficient may change, so we store the absolute values of the imaginary parts for all l, m and the absolute values of the real parts when l or m is odd. If l and m are both even, then the sign of the real part does not change and we store the actual values.

Finally, the spherical harmonic coefficients are scaled to the unit L_1 norm. From the definition of spherical harmonics, spherical harmonic coefficients are exactly proportional to the model's scale. So scaling invariance is achieved by scaling the coefficients which is done in constant time because we sample all models with the same bandwidth.

3.4. Similarity measure

Each spherical harmonic coefficient is a complex number. To compare two shape descriptors, the L_1 distance is computed.

Let $\hat{f}(l,m)=\text{Re}+j\,\text{Im}$, $\hat{f}'(l,m)=\text{Re}'+j\,\text{Im}'$ be two corresponding coefficients of two different shape descriptors. The L_1 distance between them is

$$|\hat{f}(l,m)-\hat{f}'(l,m)| = \sqrt{(\text{Re}-\text{Re}')^2+(\text{Im}-\text{Im}')^2}. \tag{14}$$

Eq. (14) is computed for all coefficients where $l \leqslant L_{\text{bands}}$ and $m \geqslant 0$, due to the property derived from Eq. (13) that leads to the following:

$$\begin{aligned} |\hat{f}(l,m)-\hat{f}'(l,m)| &= |(-1)^m \overline{\hat{f}(l,-m)} - (-1)^m \overline{\hat{f}'(l,-m)}| \\ &= |\hat{f}(l,-m)-\hat{f}'(l,-m)|. \end{aligned} \tag{15}$$

Thus only the positive order coefficients are necessary.

The descriptor consists of two sets of coefficients corresponding to the two aligned versions of the model (using CPCA and NPCA). The comparison between two models is done between the corresponding sets. Consequently, the CPCA aligned model is compared with the CPCA aligned version of the second model, and similarly for the NPCA version. The comparison giving the minimum distance sets the final difference. The notion of taking the minimum distance is based on the expectation that the best establishment of correspondences between two models is achieved, when the difference between the shape descriptors is minimum.

4. Experimental results

In this section, we give performance results for the proposed shape descriptor and compare it against state-of-the-art methods. The evaluation was run in the following 3D model databases: The Princeton Shape Benchmark (PSB) (both training and test set) [32], the classified models of the National Taiwan University database (NTU) [33], the MPEG-7 dataset [34], and the classified models of the CCCC dataset [35]. From the NTU and CCCC datasets only the classified models were used because the entire datasets contain a large number of unclassified models thus, using the whole datasets would not give

Table 1
Cardinalities of 3D model datasets and number of classes per dataset

3D model dataset	#Classes	#Models
PSB test	92	907
PSB train	90	907
NTU	47	549
MPEG-7	102	1300
CCCC	55	472

accurate results. Table 1 shows the number of categories and the total number of models for each dataset.

We compare the proposed descriptor with the following shape retrieval methods which are considered state-of-the-art with respect to their discriminative power: (i) hybrid (DSR472) [5]; (ii) depth buffer-based (DB) [5]; (iii) silhouette-based (SIL) [5]; (iv) the ray-based using spherical harmonics (RSH) [27]; (v) light-field (LF) [11]; (vi) the spherical harmonic representation of the Gaussian Euclidean distance transform descriptor (SH-GEDT) [14].

Descriptors (i)–(iv) were tested using tools provided in [35], the LF descriptor was produced by using the executable provided at [33] and the SH-GEDT descriptor was produced by using the executable provided at [36].

Our quantitative evaluation is based on the following measures:

- *Precision–Recall plots* (*P–R*): For a query model which belongs to a class C, recall is the percentage of models of class C that are retrieved and precision is the proportion of models that belong to class C to the number of retrieved models.
- *Nearest neighbor*: Given a query model, its nearest neighbor is its closest match. The nearest neighbor measure is the percentage of queries where the closest match belongs to the query model's class.
- *First tier*, *second tier*: Given a query model belonging to class C with $|C|$ number of models, the first tier measure is the proportion of the first $|C|-1$ retrieved models that belong to class C. The second tier measure is the proportion of the first $(2|C|-1)$ retrieved models that belong to class C. We average these values for all models in each dataset.
- *Discounted cumulative gain* (*DCG*): For a query model belonging to class C, a list G is produced where G_i is equal to 1 if the ith match belongs to class C, otherwise $G_i = 0$. Then DCG_i is given by

$$\mathrm{DCG}_i = \begin{cases} G_1, & i = 1, \\ \mathrm{DCG}_{i-1} + \dfrac{G_i}{\log_2 i} & \text{otherwise} \end{cases} \qquad (16)$$

and the overall score is given by

$$\mathrm{DCG} = \frac{\mathrm{DCG}_k}{1 + \sum_{i=2}^{|C|} \log_2 i}, \qquad (17)$$

where k is the number of models of the dataset.

For all the above quantitative measures, the best score is 100%.

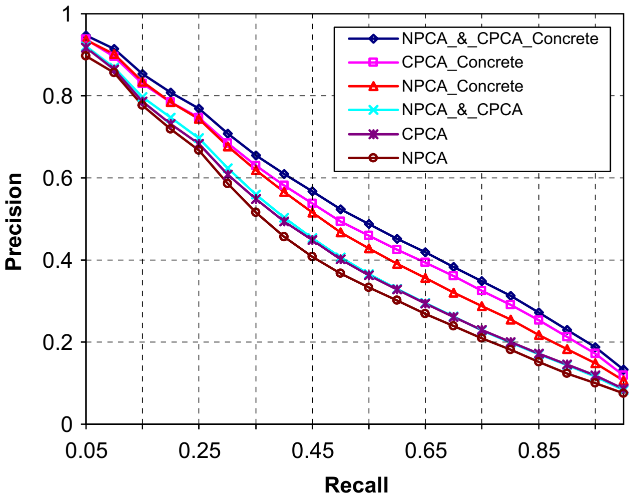

Fig. 8. P–R plot for the Princeton Shape Benchmark test dataset. A significant gain in discriminative power is achieved when NPCA is used along with CPCA to alleviate the rotation invariance problem and when the concrete representation is adopted (Eq. (11)).

Table 2
Comparison of the quantitative measures for the *CRSP*, LF and SH-GEDT descriptors on the PSB test dataset

PSB test	Nearest neighbor (%)	First tier (%)	Second tier (%)	DCG (%)
CRSP	67.9	40.5	52.8	66.8
LF	65.7	38.0	48.7	64.3
SH-GEDT	55.6	30.9	41.1	58.4

In the P–R plot of Fig. 8, we show the significant gain in discriminative power when Eq. (11) is adopted to obtain a more concrete representation of a 3D model and when CPCA and NPCA are used together to alleviate the rotation invariance problem.

In Fig. 9 the P–R plot comparing all descriptors on all datasets is given. Precision changes along the vertical axis and recall along the horizontal axis. For each method, $\overline{p}_{50}$ and $\overline{p}_{100}$ values are given in the legend of each figure, indicating average precision for recall 5–50% and 5–100%, respectively.

Comparing the plots as well as the $\overline{p}_{50}$ and $\overline{p}_{100}$ values, we conclude that the *CRSP* descriptor has an overall superior performance. The difference is greater for recall up to 50%, which is the portion of the models of a class that a user is mostly interested in. In addition, the gradient of the precision curve does not seem to change significantly for recall higher than 50%.

In Table 2, we give the corresponding scores of the proposed descriptor for the PSB test dataset and compare it to the LF and SH-GEDT descriptors. Their scores were taken from the Princeton Shape Benchmark [37] and we used the tools provided in [32] to produce the scores of our descriptor. The top performance of the *CRSP* descriptor is thus justified by the quantitative measures as well.

We believe that a difference over 2% is not trivial when top performing methods are compared, thus the *CRSP* de-

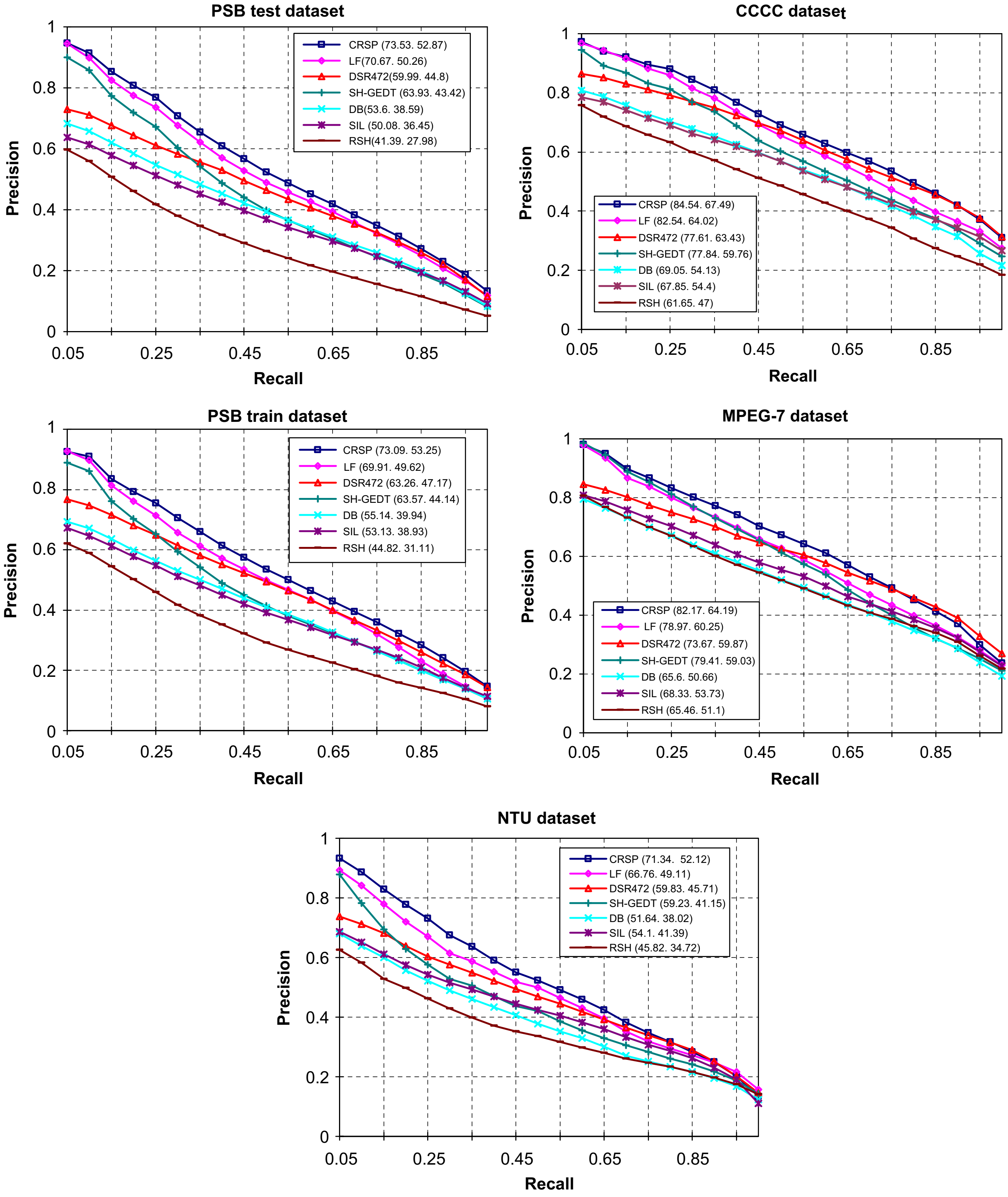

Fig. 9. Precision–Recall plots for the *CRSP*, LF, DSR472, SH-GEDT, DB, SIL and RSH descriptors for all datasets.

scriptor has a significant gain over the LF and the SH-GEDT descriptors.

In Table 3 the scores of the *CRSP* descriptor are given for the remaining datasets.

In Table 4, we give the quantitative measure scores of *CRSP* for each class of the “coarse2Test” and “coarse2Train” classifications of the Princeton Shape Benchmark and the respective DCG scores for the SH-GEDT and LF descriptors.

Table 3
Quantitative measure scores for the proposed *CRSP* shape descriptor on various datasets

	Nearest neighbor (%)	First tier (%)	Second tier (%)	DCG (%)
PSB train	70.3	41.8	55.3	69.0
NTU	72.1	42.0	55.3	70.1
MPEG-7	84.8	54.9	66.2	79.6
CCCC	81.1	54.8	68.2	78.6

Table 4
Quantitative measure scores for the *CRSP* descriptor on the “coarse2Test” and “coarse2Train” classifications of the Princeton Shape Benchmark in comparison to the DCG score of the SH-GEDT and LF descriptors

CRSP					SH-GEDT	LF
	Nearest neighbor (%)	First tier (%)	Second tier (%)	DCG (%)	DCG (%)	DCG (%)
coarse2Test						
Vehicle (245)	91.8	40.5	67.0	84.9	83.1	81.8
Animal (155)	84.5	39.0	60.5	82.3	79.6	81.2
Household (185)	77.3	20.5	34.6	74.0	72.4	75.6
Building (47)	34.0	13.8	22.2	54.3	50.3	59.7
Furniture (94)	78.7	22.2	32.6	70.4	66.6	72.7
Plant (60)	88.3	35.6	48.2	76.6	69.3	69.9
coarse2Train						
Vehicle (230)	94.3	44.9	73.0	87.4	86.3	84.0
Animal (123)	85.4	40.3	59.7	82.2	79.2	80.1
Household (219)	82.6	23.9	42.9	77.4	75.5	78.5
Building (53)	35.8	13.6	21.9	54.7	54.6	57.5
Furniture (104)	87.5	23.3	36.1	73.0	66.8	73.5
Plant (78)	88.5	43.7	60.6	82.3	69.4	73.6

The cardinality of each class is given in parenthesis next to the class name.

Table 5
Comparison between the *CRSP*, LF and SH-GEDT descriptors in the “plant“ class of the “coarse2Test” classification of the Princeton Shape Benchmark and its subclasses, using the DCG score

	CRSP	SH-GEDT	LF
	DCG (%)	DCG (%)	DCG (%)
Plant (60)	76.6	69.3	69.9
Subclass of plant			
Bush (9)	52.4	39.2	44.7
Flowers (4)	58.2	34.2	35.6
Potted plant (26)	68.1	56.3	59.9
Barren tree (11)	71.6	47.1	47.5
Conical tree (10)	51.0	51.2	48.2

The cardinality of each class is given in parenthesis next to the class name.

It is evident that while the performance of *CRSP* may vary among different classes, it achieves a better overall performance than LF and SH-GEDT. As it is shown in Table 4, the best performance is achieved for the class “vehicle” having the highest cardinality, while the worst performance is for the “building” class having the lowest cardinality. The higher cardinality of a class the more significant the results are.

In Table 5, we have split the “plant” class of the coares2Test classification into its five subclasses according to the test classification, to measure how performance is affected in the case of different degrees of intra-class variability. All other classes (vehicle, animal, household, building and furniture) in the coarse2Test classification keep the same granularity thus the performance of the corresponding methods in these classes remains the same.

The *CRSP* descriptor has the additional advantage of having very short extraction and comparison time. Using a computer with an AMD processor at 1.4 GHz, the average extraction time (including searching and removing identical triangles) was less than one second and the average comparison time was 0.16 ms. The average number of triangles per model was 8180 and the average number of vertices was 4427. We believe that the short extraction and comparison time is a significant advantage of our descriptor that enables very efficient online search and retrieval of 3D models. Furthermore, they enable extensions and future improvement (such as combinations with other descriptors).

In Fig. 10, we show examples of retrieval results for different queries within the PSB test dataset. For each query (top left model), the first 16 matches are shown using the CRSP and the LF descriptors.

Comparing the retrieved models for each query, we can conclude that the quality of the results depends on the cardinality of the query model’s class, the intra-class variation as well as the inter-class variation. The third factor is particularly important as there may be classes closely related to each other. In such a setting, mixed retrieved results from these classes will have little impact on the user’s dissatisfaction, while the opposite is true when mixed retrieved models belong to classes with large inter-class variation. The above query results are from the Princeton Shape Benchmark 3D model dataset where models have been classified by the designers according to similarity in the shape’s functionality. The CRSP descriptor is based on the establishment of correspondences between 3D models and measures the similarity between models by taking into account the shape’s geometry. Thus, the quality of the results is even better when models are classified according to this criterion.

5. Conclusion

We propose a new methodology for content based 3D shape retrieval. It uses CPCA along with PCA applied on the model’s face normals in order to deal with the rotation invariance problem. The model is decomposed into a set of spherical functions representing intersections of the model’s surface with rays emanating from the origin. Parts closer to the origin than the furthest intersection point at each ray are included in the spherical functions, which improves the descriptor’s discriminative power. Spherical functions are represented by spherical harmonics and their properties are used to achieve scaling invariance and axial flipping invariance.

We extensively evaluated the proposed 3D shape descriptor on various 3D model datasets and compared it against state-of-the-art methods. The evaluation results lead to the conclusion that the *CRSP* descriptor has very high discriminative power and is ranked first among the other methods that we compared

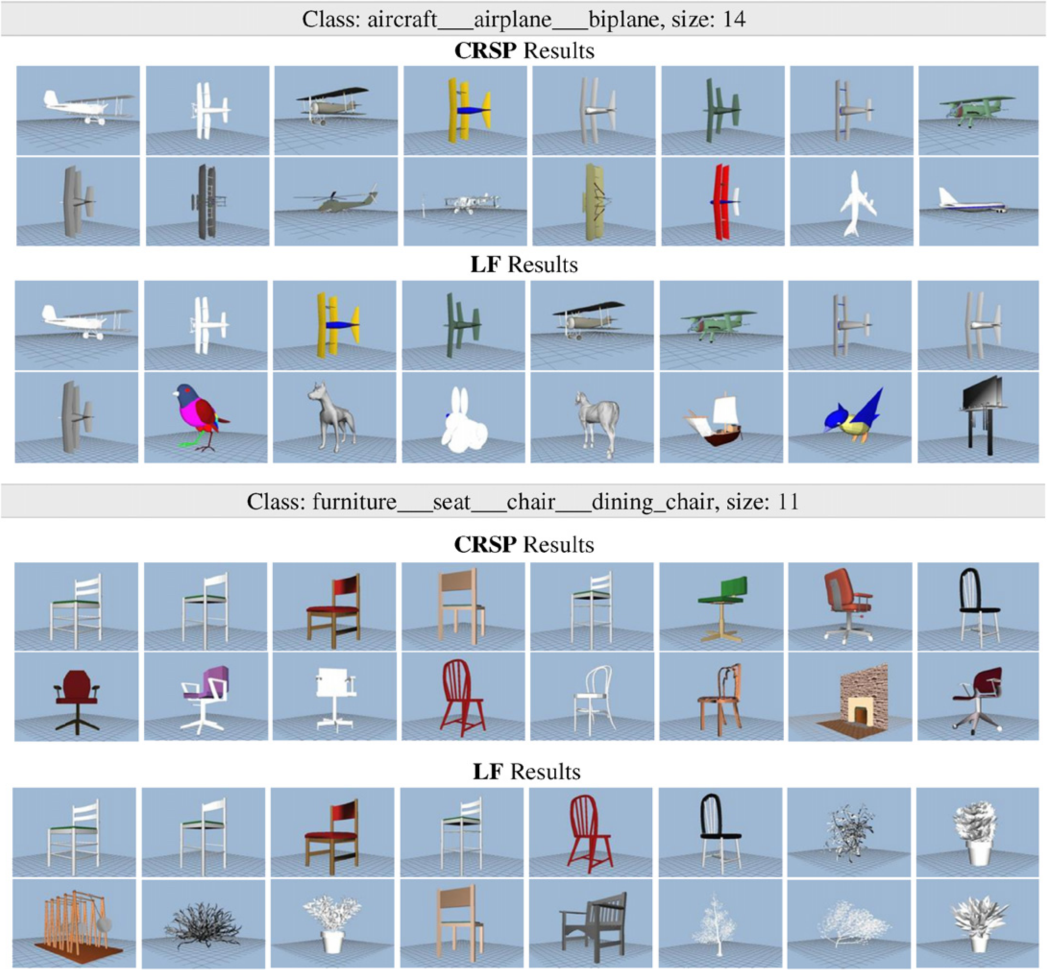
Class: aircraft___airplane___biplane, size: 14
CRSP Results
LF Results
Class: furniture___seat___chair___dining_chair, size: 11
CRSP Results
LF Results

Fig. 10. Example queries and the corresponding top 16 matches from the Princeton Shape Benchmark test dataset, using the concrete radialized spherical projection descriptor (CRSP) and the light field descriptor (LF). The top left model is the query model. The first row shows the results 1–8 while the second row shows the results 9–16.

with. The *CRSP* descriptor has the additional advantage of having very short extraction and comparison times.

In future work we aim to further improve the quality of the results by considering the user's relevance feedback. In that framework, the user would mark relevant models or assign a measure of relevance to some of the results. Using this information the similarity between the query and the models of the database would be re-evaluated and a more accurate list of results would be returned.

Acknowledgment

This research was supported by the Greek Secretariat of Research and Technology (PENED "3D Graphics search and retrieval" 03 EΔ 520).

Appendix A.

In the following a detailed description is given regarding the exact computation of a 3D model's center of mass and its surface covariance matrix, using the continuous principal component analysis [5,26].

Let $\mathbf{v} = [v_x, v_y, v_z]^{\mathrm{T}}$ be a point on the surface of a model; then using barycentric coordinates, $\mathbf{v}$ is equal to

$$\mathbf{v} = a \cdot \mathbf{A}_i + b \cdot \mathbf{B}_i + (1 - a - b) \cdot \mathbf{C}_i,$$

where $\mathbf{A}_i, \mathbf{B}_i$ and $\mathbf{C}_i$ are the vertices of triangle T_i, $i = 1, 2, \ldots, N$ where N is the number of triangles of the model and a, b are the barycentric coordinates. If E is the total surface area of the 3D model and E_i is the surface area of triangle T_i, then using CPCA the model's center of mass $\mathbf{m}$ is equal to

$$\begin{aligned}
\mathbf{m} &= \frac{1}{E} \int \int_{\mathbf{v} \in T_i} \mathbf{v}\, \mathrm{d}s \\
&= \frac{1}{E} \sum_{i=1}^{N} \int \int (a\mathbf{A}_i + b\mathbf{B}_i + (1 - a - b)\mathbf{C}_i)\, \mathrm{d}s \\
&= \frac{2}{E} \sum_{i=1}^{N} E_i \int_0^1 \left(\int_0^{1-a} (a\mathbf{A}_i + b\mathbf{B}_i + (1 - a - b)\mathbf{C}_i)\, \mathrm{d}b \right) \mathrm{d}a \\
&= \frac{2}{E} \sum_{i=1}^{N} E_i \int_0^1 \left[ab\mathbf{A}_i + \frac{b^2}{2}\mathbf{B}_i + \left(b - ab - \frac{b^2}{2} \right) \mathbf{C}_i \right]_0^{1-a} \mathrm{d}a
\end{aligned}$$

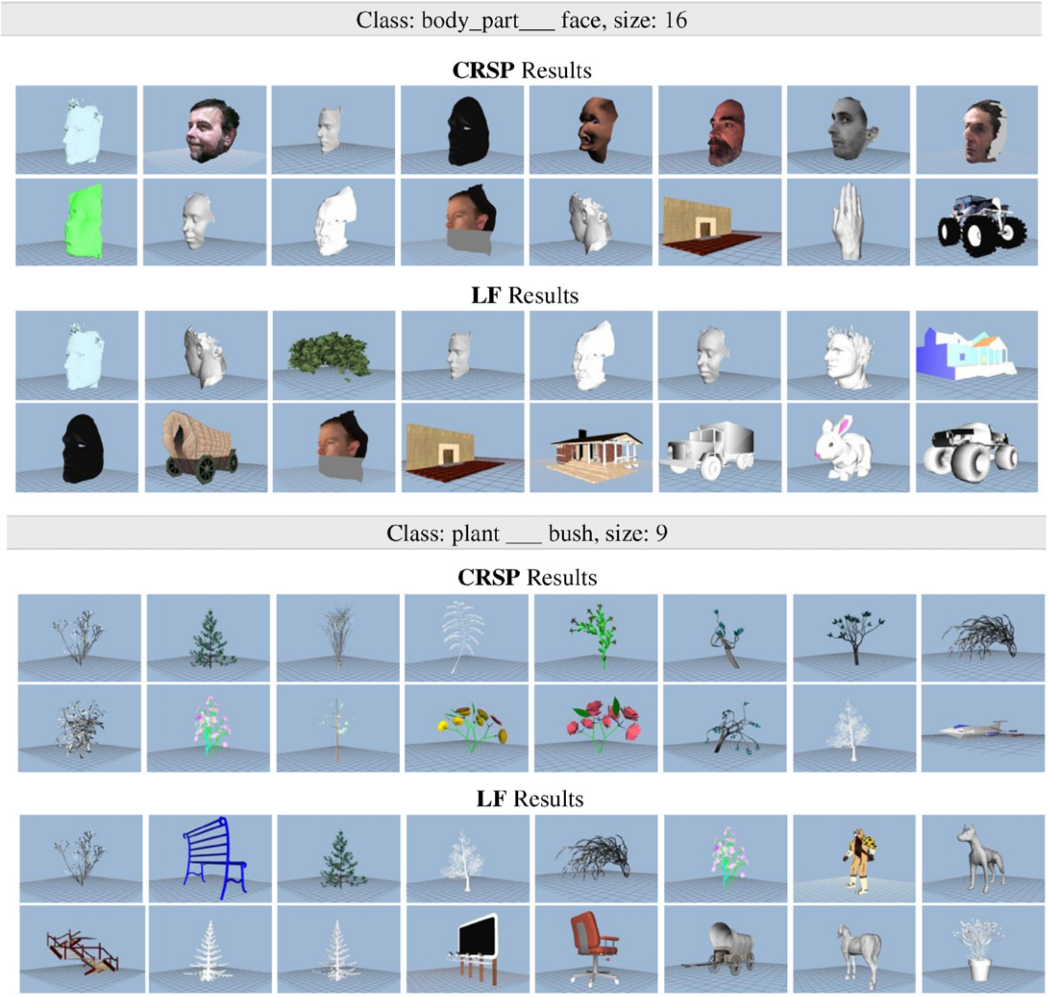
Class: body_part___ face, size: 16
CRSP Results
LF Results
Class: plant ___ bush, size: 9
CRSP Results
LF Results

Fig. 10. (*continued*).

$$= \frac{2}{E}\sum_{i=1}^{N} E_i \int_0^1 \left(a(1-a)\mathbf{A}_i + \frac{(1-a)^2}{2}\mathbf{B}_i + \left((1-a) - a(1-a) - \frac{(1-a)^2}{2}\right)\mathbf{C}_i \right) \mathrm{d}a$$

$$= \frac{2}{E}\sum_{i=1}^{N} E_i \int_0^1 \left(a\mathbf{A}_i - a^2\mathbf{A}_i + \frac{1}{2}\mathbf{B}_i - a\mathbf{B}_i + \frac{a^2}{2}\mathbf{B}_i + \frac{1}{2}\mathbf{C}_i - a\mathbf{C}_i + \frac{a^2}{2}\mathbf{C}_i \right) \mathrm{d}a$$

$$= \frac{2}{E}\sum_{i=1}^{N} E_i \left[\frac{a^2}{2}(\mathbf{A}_i - \mathbf{B}_i - \mathbf{C}_i) - \frac{a^3}{3}\left(\mathbf{A}_i - \frac{1}{2}\mathbf{B}_i - \frac{1}{2}\mathbf{C}_i\right) + \frac{1}{2}a(\mathbf{B}_i + \mathbf{C}_i) \right]_0^1$$

$$= \frac{2}{E}\sum_{i=1}^{N} E_i \left(\frac{1}{2}(\mathbf{A}_i - \mathbf{B}_i - \mathbf{C}_i) - \frac{1}{3}\left(\mathbf{A}_i - \frac{1}{2}\mathbf{B}_i - \frac{1}{2}\mathbf{C}_i\right) + \frac{1}{2}(\mathbf{B}_i + \mathbf{C}_i) \right) \Rightarrow \mathbf{m} = \frac{1}{E}\sum_{i=1}^{N} E_i \cdot \frac{(\mathbf{A}_i + \mathbf{B}_i + \mathbf{C}_i)}{3}.$$

In the sequel, we use CPCA to compute the covariance matrix $\mathbf{C}$ of the model's surface. If $\mathbf{g}(\mathbf{v}) = \mathbf{v}\mathbf{v}^{\mathrm{T}}$, then $\mathbf{C}$ is given by

$$\mathbf{C} = \frac{1}{E}\int\int_{\mathbf{v}\in M} (\mathbf{v}-\mathbf{m})(\mathbf{v}-\mathbf{m})^{\mathrm{T}}\,\mathrm{d}s$$

$$= \frac{1}{E}\sum_{i=1}^{N}\int\int_{\mathbf{v}\in T_i} (\mathbf{v}-\mathbf{m})(\mathbf{v}-\mathbf{m})^{\mathrm{T}}\,\mathrm{d}s$$

$$= \frac{1}{E}\sum_{i=1}^{N}\int\int (a\mathbf{A}_i + b\mathbf{B}_i + (1-a-b)\mathbf{C}_i - \mathbf{m}) \cdot (a\mathbf{A}_i + b\mathbf{B}_i + (1-a-b)\mathbf{C}_i - \mathbf{m})^{\mathrm{T}}\,\mathrm{d}s$$

$$= \frac{2}{E}\sum_{i=1}^{N} E_i \int_0^1 \left(\int_0^{1-a} (a\mathbf{A}_i + b\mathbf{B}_i + (1-a-b)\mathbf{C}_i - \mathbf{m}) \cdot (a\mathbf{A}_i + b\mathbf{B}_i + (1-a-b)\mathbf{C}_i - \mathbf{m})^{\mathrm{T}}\,\mathrm{d}b \right) \mathrm{d}a$$

$$= \frac{2}{E}\sum_{i=1}^{N} E_i \int_0^1 \left(\int_0^{1-a} \left(\begin{array}{l} a^2\mathbf{A}_i\mathbf{A}_i^{\mathrm{T}} + b^2\mathbf{B}_i\mathbf{B}_i^{\mathrm{T}} + (1-2a-2b+2ab+a^2+b^2)\mathbf{C}_i\mathbf{C}_i^{\mathrm{T}} + \\ ab(\mathbf{A}_i\mathbf{B}_i^{\mathrm{T}} + \mathbf{B}_i\mathbf{A}_i^{\mathrm{T}}) + (a-a^2-ab)(\mathbf{A}_i\mathbf{C}_i^{\mathrm{T}} + \mathbf{C}_i\mathbf{A}_i^{\mathrm{T}}) + \\ (b-ab-b^2)(\mathbf{B}_i\mathbf{C}_i^{\mathrm{T}} + \mathbf{C}_i\mathbf{B}_i^{\mathrm{T}}) - a\mathbf{A}_i\mathbf{m}^{\mathrm{T}} - b\mathbf{B}_i\mathbf{m}^{\mathrm{T}} - (1-a-b)\mathbf{C}_i\mathbf{m}^{\mathrm{T}} - \\ a\mathbf{m}\mathbf{A}_i^{\mathrm{T}} - b\mathbf{m}\mathbf{B}_i^{\mathrm{T}} - (1-a-b)\mathbf{m}\mathbf{C}_i^{\mathrm{T}} + \mathbf{m}\mathbf{m}^{\mathrm{T}}\,\mathrm{d}b \end{array} \right) \right) \mathrm{d}a$$

$$= \frac{2}{E}\sum_{i=1}^{N} E_i \int_0^1 \left(\begin{array}{l} (a^2-a^3)\mathbf{A}_i\mathbf{A}_i^{\mathrm{T}} + \dfrac{(1-a)^3}{3}\left(\mathbf{B}_i\mathbf{B}_i^{\mathrm{T}} + \mathbf{C}_i\mathbf{C}_i^{\mathrm{T}} + \dfrac{\mathbf{B}_i\mathbf{C}_i^{\mathrm{T}}}{2} + \dfrac{\mathbf{C}_i\mathbf{B}_i^{\mathrm{T}}}{2}\right) \\ +a\dfrac{(1-a)^2}{2}(\mathbf{A}_i\mathbf{B}_i^{\mathrm{T}} + \mathbf{B}_i\mathbf{A}_i^{\mathrm{T}} + \mathbf{A}_i\mathbf{C}_i^{\mathrm{T}} + \mathbf{C}_i\mathbf{A}_i^{\mathrm{T}}) - a(1-a)(\mathbf{A}_i\mathbf{m}^{\mathrm{T}} + \mathbf{m}\mathbf{A}_i^{\mathrm{T}}) \\ -\dfrac{(1-a)^2}{2}(\mathbf{B}_i\mathbf{m}^{\mathrm{T}} + \mathbf{C}_i\mathbf{m}^{\mathrm{T}} + \mathbf{m}\mathbf{B}_i^{\mathrm{T}} + \mathbf{m}\mathbf{C}_i^{\mathrm{T}}) + (1-a)\mathbf{m}\mathbf{m}^{\mathrm{T}} \end{array} \right) \mathrm{d}a$$

$$= \frac{1}{12E}\sum_{i=1}^{N} E_i \left(\begin{array}{l} 2\mathbf{A}_i\mathbf{A}_i^{\mathrm{T}} + 2\mathbf{B}_i\mathbf{B}_i^{\mathrm{T}} + 2\mathbf{C}_i\mathbf{C}_i^{\mathrm{T}} + \mathbf{B}_i\mathbf{C}_i^{\mathrm{T}} + \mathbf{C}_i\mathbf{B}_i^{\mathrm{T}} + \mathbf{A}_i\mathbf{B}_i^{\mathrm{T}} + \mathbf{B}_i\mathbf{A}_i^{\mathrm{T}} + \mathbf{A}_i\mathbf{C}_i^{\mathrm{T}} + \mathbf{C}_i\mathbf{A}_i^{\mathrm{T}} \\ -4\mathbf{A}_i\mathbf{m}^{\mathrm{T}} - 4\mathbf{m}\mathbf{A}_i^{\mathrm{T}} - 4\mathbf{B}_i\mathbf{m}^{\mathrm{T}} - 4\mathbf{C}_i\mathbf{m}^{\mathrm{T}} - 4\mathbf{m}\mathbf{B}_i^{\mathrm{T}} - 4\mathbf{m}\mathbf{C}_i^{\mathrm{T}} + 12\mathbf{m}\mathbf{m}^{\mathrm{T}} \end{array} \right)$$

$$= \frac{1}{12E}\sum_{i=1}^{N} E_i \left(\begin{array}{l} \mathbf{A}_i\mathbf{A}_i^{\mathrm{T}} - \mathbf{A}_i\mathbf{m}^{\mathrm{T}} - \mathbf{m}\mathbf{A}_i^{\mathrm{T}} + \mathbf{m}\mathbf{m}^{\mathrm{T}} + \mathbf{B}_i\mathbf{B}_i^{\mathrm{T}} - \mathbf{B}_i\mathbf{m}^{\mathrm{T}} - \mathbf{m}\mathbf{B}_i^{\mathrm{T}} + \mathbf{m}\mathbf{m}^{\mathrm{T}} \\ +\mathbf{C}_i\mathbf{C}_i^{\mathrm{T}} - \mathbf{C}_i\mathbf{m}^{\mathrm{T}} - \mathbf{m}\mathbf{C}_i^{\mathrm{T}} + \mathbf{m}\mathbf{m}^{\mathrm{T}} + \mathbf{A}_i\mathbf{A}_i^{\mathrm{T}} + \mathbf{A}_i\mathbf{B}_i^{\mathrm{T}} + \mathbf{A}_i\mathbf{C}_i^{\mathrm{T}} - \mathbf{A}_i 3\mathbf{m}^{\mathrm{T}} \\ +\mathbf{B}_i\mathbf{A}_i^{\mathrm{T}} + \mathbf{B}_i\mathbf{B}_i^{\mathrm{T}} + \mathbf{B}_i\mathbf{C}_i^{\mathrm{T}} - \mathbf{B}_i 3\mathbf{m}^{\mathrm{T}} + \mathbf{C}_i\mathbf{A}_i^{\mathrm{T}} + \mathbf{C}_i\mathbf{B}_i^{\mathrm{T}} + \mathbf{C}_i\mathbf{C}_i^{\mathrm{T}} - \mathbf{C}_i 3\mathbf{m}^{\mathrm{T}} \\ -3\mathbf{m}\mathbf{A}_i^{\mathrm{T}} - 3\mathbf{m}\mathbf{B}_i^{\mathrm{T}} - 3\mathbf{m}\mathbf{C}_i^{\mathrm{T}} + 3\mathbf{m}3\mathbf{m}^{\mathrm{T}} \end{array} \right)$$

$$\Rightarrow \mathbf{C} = \frac{1}{12E}\sum_{i=1}^{N} E_i \cdot \left[\mathbf{f}(\mathbf{A}_i) + \mathbf{f}(\mathbf{B}_i) + \mathbf{f}(\mathbf{C}_i) + 9 \cdot \mathbf{f}\left(\frac{(\mathbf{A}_i + \mathbf{B}_i + \mathbf{C}_i)}{3}\right) \right],$$

where $\mathbf{f}(\mathbf{v}) = (\mathbf{v}-\mathbf{m})(\mathbf{v}-\mathbf{m})^{\mathrm{T}}$.

Appendix B.

In this following, a description of the theory of spherical harmonics [38] is given.

Let S^2 denote the unit sphere, which is the sphere with radius 1 and center at the coordinates' origin. Then a point $\mathbf{v}(\theta, \varphi)$ on the surface of S^2 can be expressed as

$$\mathbf{v}(\theta, \varphi) = (\cos\varphi \sin\theta, \sin\varphi \sin\theta, \cos\theta), \quad (18)$$

where $0 \leqslant \theta \leqslant \pi$ is measured counter-clockwise on the yz plane and $0 \leqslant \varphi < 2\pi$ is measured clockwise on the xy plane.

Let $L^2(S^2)$, denote the Hilbert space of square integrable complex functions in S^2. Then the inner product of two functions $f, h \in L^2(S^2)$ is given by

$$\langle f, h \rangle = \int_0^{\pi} \left[\int_0^{2\pi} f(\theta, \varphi) \cdot \overline{h(\theta, \varphi)}\, \mathrm{d}\varphi \right] \sin\theta\, \mathrm{d}\theta. \quad (19)$$

Spherical harmonics form an orthonormal basis of the space $L^2(S^2)$. Consequently, every function $f \in L^2(S^2)$ is given by

$$f(\theta, \varphi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \hat{f}(l, m) \cdot Y_l^m(\theta, \varphi), \quad (20)$$

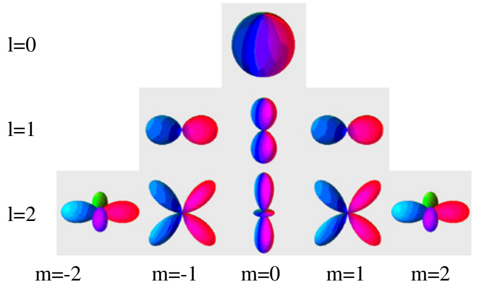

Fig. 11. Representation of $[\mathrm{Re}\{Y_l^m(\theta, \varphi)\}]^2$ for the spherical harmonics up to degree $l = 2$.

where the $\hat{f}(l, m)$ corresponds to the lth degree and mth order Fourier coefficient and $Y_l^m(\theta, \varphi)$ corresponds to the equivalent spherical harmonic function

$$Y_l^m(\theta, \varphi) = k_{l,m} \cdot P_l^m(\cos\theta)\mathrm{e}^{\mathrm{i}m\varphi}, \quad (21)$$

where $k_{l,m}$ is a normalization constant and $P_l^m(\cos\theta)$ is the associated Legendre polynomial of degree l and order m. A visual representation of the spherical harmonics up to the 2nd degree is given in Fig. 11.

Each coefficient $\hat{f}(l, m)$ equals to the projection of the spherical function $f(\theta, \varphi)$ to the equivalent spherical harmonic function, that is

$$\hat{f}(l,m) = \langle f(\theta,\varphi), Y_l^m(\theta,\varphi)\rangle = k_{l,m} \cdot \int_0^{\pi} \left[\int_0^{2\pi} f(\theta,\varphi) \cdot \mathrm{e}^{-\mathrm{i}m\,\varphi}\,\mathrm{d}\varphi \right] \cdot P_l^m(\cos\theta) \cdot \sin\theta\,\mathrm{d}\theta. \quad (22)$$

We consider band-limited functions, so the following theorem [38] to compute the SHT is used:

Theorem 1. *Let $f \in L^2(S^2)$ have bandwidth B. Then for each $|m| \leqslant l < B$:*

$$\hat{f}(l,m) = \frac{\sqrt{2\pi}}{2B} \sum_{j=0}^{2B-1} \sum_{k=0}^{2B-1} a_j^{(B)} f(\theta_j,\varphi_k) \mathrm{e}^{-\mathrm{i}m\,\varphi_k} P_l^m(\cos\theta_j), \quad (23)$$

where $\theta_j = (2j+1)\pi/(4B)$, $\varphi_k = 2\pi k/(2B)$ and weights $a_j^{(B)}$ play an analogous role with the $\sin\theta$ factor in the integrals of Eq. (22).

References

[1] M. Ankerst, G. Kastenmuller, H.P. Kriegel, T. Seidl, Nearest neighbor classification in 3D protein databases, in: Seventh International Conference on Intelligent Systems for Molecular Biology, Heidelberg, 1999

[2] M. Körtgen, G.J. Parl, M. Novotni, R. Klein, 3D shape matching with 3D shape contexts, in: Seventh Central European Seminar on Computer Graphics, 2003.

[3] D.V. Vranic, D. Saupe, 3D model retrieval, in: Spring Conference on Computer Graphics and its Applications, Comenius University Slovakia, 2000, pp. 89–93.

[4] D. Saupe, D.V. Vranic, 3d model retrieval with spherical harmonics and moments, in: Proceedings of DAGM, Munich, 2001, pp. 392–397.

[5] D.V. Vranic, 3D model retrieval, Ph.D. Dissertation, University of Leipzig, Department of Computer Science, 2004.

[6] D.V. Vranic, An improvement of rotation invariant 3D shape descriptor based on functions on concentric spheres, in: IEEE International Conference on Image Processing, vol. 3, Barcelona, 2003, pp. 757–760.

[7] B. Horn, Extended Gaussian images, Proc. IEEE 72 (12) (1984) 1671–1686.

[8] S. Kang, K. Ikeuchi, Determining 3-D object pose using the complex extended Gaussian image, in: Computer Vision and Pattern Recognition, Lahaina Maul, 1991, pp. 580–585.

[9] M. Novotni, R. Klein, 3D Zernike descriptors for content based shape retrieval, in: Eighth ACM Symposium on Solid Modeling and Applications, Seatle, 2003.

[10] P. Daras, D. Zarpalas, D. Tzovaras, M.G. Strintzis, Efficient 3-D model search and retrieval using generalized 3-D radon transforms, J. IEEE Trans. Multimedia 8 (1) (2006) 101–114.

[11] D.Y. Chen, M. Ouhyoung, X.P. Tian, Y.T. Shen, On visual similarity based 3D model retrieval, in: Computer Graphics Forum, 2003, pp. 223–232.

[12] R. Ohbuchi, M. Nakazawa, T. Takei, Retrieving 3D shapes based on their appearance, in: Fifth ACM SIGMM Workshop on Multimedia Information Retrieval, Berkeley, California, 2003, pp. 39–46.

[13] D. Zarpalas, P. Daras, A. Axenopoulos, D. Tzovaras, M.G. Strintzis, 3D model search and retrieval using the spherical trace transform, in: IEEE International Workshop on Multimedia Signal Processing, Siena, Italy, 2004.

[14] M. Kazhdan, T. Funkhouser, S. Rusinkiewicz, Rotation invariant spherical harmonic representation of 3D shape descriptors, in: Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, Aachen, 2003, pp. 156–164.

[15] M. Kazhdan, B. Chazelle, D. Dobkin, A. Finkelstein, T. Funkhouser, A reflective symmetry descriptor, in: European Conference on Computer Vision, Copenhagen, 2002, pp. 642–656.

[16] M.T. Suzuki, T. Kato, N. Otsu, A similarity retrieval of 3D polygonal models using rotation invariant shape descriptors, in: IEEE International Conference on Systems, Man, and Cybernetics, Nashville, 2000, pp. 2946–2952.

[17] J.W.H. Tangelder, R.C. Veltkamp, Polyhedral model retrieval using weighted point sets, Int. J. Image Graphics 3 (1) (2003) 209–229.

[18] M. Hilaga, Y. Shinagawa, T. Kohmura, T. Kohmura, T.L. Kunii, Topology matching for fully automatic similarity estimation of 3D shapes, in: ACM SIGGRAPH 28th Annual Conference on Computer Graphics and Interactive Techniques, Los Angeles, 2001, pp. 203–212.

[19] J. Zhang, K. Siddiqi, D. Macrini, A. Shokoufandeh, S. Dickinson, Retrieving articulated 3-D models using medial surfaces and their graph spectra, in: International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition, St. Augustine, 2005, pp. 285–300.

[20] H. Sundar, D. Silver, N. Gagvani, S. Dickinson, Skeleton based shape matching and retrieval, in: International Conference on Shape Modeling and Applications, Seoul, 2003, pp. 130–142.

[21] N.D. Cornea, M.F. Demirci, D. Silver, A. Shokoufandeh, S. Dickinson, P.B. Kantor, 3D object retrieval using many-to-many matching of curve skeletons, in: International Conference on Shape Modeling and Applications, MIT, 2005, pp. 368–373.

[22] R. Osada, T. Funkhouser, B. Chazelle, D. Dobkin, Matching 3D models with shape distributions: in: International Conference on Shape Modeling and Applications, Genova, 2001, pp. 154–166.

[23] R. Osada, T. Funkhouser, B. Chazelle, D. Dobkin, Shape distributions, J. ACM Trans. Graphics 21 (4) (2002) 807–832.

[24] R. Ohbuchi, T. Minamitani, T. Takei, Shape-similarity search of 3D models by using enhanced shape functions, Int. J. Comput. Appl. Technol. 23 (2/3/4) (2005) 70–85.

[25] R. Ohbuchi, T. Takei, Shape-similarity comparison of 3D models using alpha shapes, in: 11th Pacific Conference on Computer Graphics and Applications, Canmore, 2003, pp. 293–302.

[26] T. Zaharia, F. Petreux, 3D shape-based retrieval within the MPEG-7 framework, in: SPIE Conference 4304 on Nonlinear Image Processing and Pattern Analysis, vol. XII, San Jose, 2001, pp. 133–145.

[27] D.V. Vranic, D. Saupe, J. Richter, Tools for 3D-object retrieval: Karhunen–Loeve transform and spherical harmonics, in: IEEE Workshop on Multimedia and Signal Processing, Cannes, 2001, pp. 293–298.

[28] S. Theodoridis, K. Koutroumbas, Pattern Recognition, Academic Press, New York, 1999.

[29] E. Paquet, A. Murching, T. Naveen, A. Tabatabai, M. Rioux, Description of shape information for 2-D and 3-D objects, J. Signal Process. Image Communication 16 (2000) 103–122.

[30] M. Elad, A.Tal, S. Ar, Contend based retrieval of VRML objects—an iterative and interactive approach, in: Sixth Eurographics Workshop in Multimedia, Manchester, 2001.

[31] ⟨http://www.cs.dartmouth.edu/~geelong/sphere/⟩.

[32] ⟨http://shape.cs.princeton.edu/benchmark/⟩.

[33] ⟨http://3d.csie.ntu.edu.tw/⟩.

[34] ⟨http://www.chiariglione.org/MPEG/standards/mpeg-7/mpeg-7.htm⟩.

[35] ⟨http://merkur01.inf.uni-konstanz.de/CCCC/⟩, Dejan Vranic's 3D Search Engine.

[36] ⟨http://www.cs.jhu.edu/~misha/HarmonicSignatures/⟩.

[37] P. Shilane, P. Min, M. Kazhdan, T. Funkhouser, The Princeton Shape Benchmark, in: International Conference on Shape Modeling and Applications, Genova, 2004, pp. 167–178.

[38] D.M. Healy Jr., D. Rockmore, P. Kostelec, S. Moore, FFTs for the 2-sphere—improvements and variations, J. Fourier Anal. Appl. 9 (4) (2003) 341–385.

About the Author—PANAGIOTIS PAPADAKIS received the degree in Informatics and Telecommunications from the National Kapodistrian University of Athens, Greece in 2005. He is currently pursuing research for a D.Phil. degree at the Institute of Informatics and Telecommunications of the National Center for Scientific Research “Demokritos”, Athens, Greece. His research interests span the field of artificial intelligence and computer graphics with a special focus on 3D models search and retrieval.

About the Author—STAVROS J. PERANTONIS is the holder of a B.S. degree in Physics from the Department of Physics, University of Athens, an M.Sc. degree in Computer Science from the Department of Computer Science, University of Liverpool and a D.Phil. degree in Computational Physics from the Department of Physics, University of Oxford. Since 1992 he has been with the Institute of Informatics and Telecommunications, NCSR “Demokritos”, where he currently holds the position of a Senior Researcher and Head of the Computational Intelligence Laboratory. His main research interests are in image processing and document image analysis, OCR and pattern recognition.

About the Author—IOANNIS PRATIKAKIS received the Diploma degree in Electrical Engineering from the Demokritus University of Thrace, Xanthi, Greece in 1992, and the D.Phil. degree in Applied Sciences from Vrije Universiteit Brussel, Brussels, Belgium, in 1998. From March 1999 to March 2000, he was at IRISA/ViSTA group, Rennes, France as an INRIA postdoctoral fellow. He is currently working as a Research Scientist at the Institute of Informatics and Telecommunications of the National Center for Scientific Research “Demokritos”, Athens, Greece. His research interests include 2D and 3D image analysis, image and volume sequence analysis and content-based image/3D models search and retrieval. He has published more than 50 papers in journals, book chapters and papers in conference proceedings in the above areas. He has participated in more than 10 national and international R&D projects.

About the Author—THEOHARIS THEOHARIS received the B.Sc. degree in Computer Science from the University of London, UK in 1984, the M.Sc. Degree in Computation from the University of Oxford in 1985 and the D.Phil. degree in Computer Graphics and Parallel processing from the University of Oxford in 1988. He is currently a Associate Professor at the Department of Informatics of the National Kapodistrian University of Athens, Greece where he leads the Computer Graphics Group. His current research interests lie in the fileds of biometrics, three dimensional object retrieval and reconstruction.