

# Content Based Retrieval of 3D Data

Alberto Del Bimbo and Pietro Pala

Dipartimento Sistemi e Informatica  
Università di Firenze  
via S.Marta 3, 50139 Firenze, Italy  
{delbimbo,pala}@dsi.unifi.it

**Abstract.** Along with images and videos, 3D models have recently gained increasing attention for a number of reasons: advancements in 3D hardware and software technologies, their ever decreasing prices and increasing availability, affordable 3D authoring tools, and the establishment of open standards for 3D data interchange.

The ever increasing availability of 3D models demands for tools supporting their effective and efficient management. Among these tools, those enabling content-based retrieval play a key role.

In this paper we report on our experience in developing models to support retrieval by content of 3D objects. Particularly, we present three different models for representing and comparing the content of 3D objects. A comparative analysis is carried out to evidence the actual potential of the proposed solutions.

## 1 Introduction

Digital multimedia information is nowadays spreading through all sectors of society and collections of multimedia documents are being created at an increasing pace in several domains. However, in order to exploit the valuable assets contained in these ever growing collections, some tool should be available to support users in the process of finding information out of these data. In recent years, as a result of the efforts spent in the attempt of finding solutions to this problem, many systems have been developed that enable effective retrieval from digital libraries, covering text, audio, images, and videos.

Beside image and video databases, archives of 3D models have recently gained increasing attention for a number of reasons: advancements in 3D hardware and software technologies, their ever increasing availability at affordable costs, and the establishment of open standards for 3D data interchange (e.g. VRML, X3D).

Acquisition of the 3D model of an object, capturing both object geometry and its visual features (surface color and texture), can be achieved through many different techniques, including CAD, 3D laser scanners, structured light systems and photogrammetry. The selection of the most appropriate technique depends on application specific quality requirements. Furthermore, these techniques result in a large variety of models, differing in terms of their representation (e.g. point clouds, voxels, analytical functions), of their resolution and size, of the presence, nature, and amount of noise and artifacts.

Thanks to the availability of technologies for their acquisition, 3D models are being employed in a wide range of application domains, including medicine, computer aided design and engineering, and cultural heritage. In this framework the development of techniques to enable retrieval by content of 3D models assumes an ever increasing relevance.

This is particularly the case in the fields of cultural heritage and historical relics, where there is an increasing interest in solutions enabling preservation of relevant artworks (e.g. vases, sculptures, and handicrafts) as well as cataloguing and retrieval by content. In these fields, retrieval by content can be employed to detect commonalities between 3D objects (e.g. the “signature” of the artist) or to monitor the temporal evolution of a defect (e.g. the amount of bending for wooden tables).

Methods addressing retrieval of 3D models can be distinguished based on different aspects, such as the type of representation used for geometry, the use of information about models’ aspect (i.e. colour and/or texture), the need for manual annotation.

Description and retrieval of 3D objects based on description and retrieval of 2D views has been addressed in [1] and [2]. However, the effectiveness of these solutions is limited to description and retrieval of simple objects. In fact, as complex objects are considered, occlusions prevent to capture distinguishing 3D features using 2D views.

Description of 3D surface data for the purpose of recognition or retrieval has been addressed for some time. A few authors have investigated analytical 3D models, but this is not always a viable solution, as there are many limitations in providing parameterizations of arbitrary models. In [3] retrieval of 3D objects based on similarity of surface segments is addressed. Surface segments model potential docking sites of molecular structures. The proposed approach develops on the approximation error of the surface. However, assumptions on the form of the function to be approximated limit applicability of the approach to special contexts.

Much attention has been recently devoted to free-form (i.e. polygonal) meshes. While this representation of 3D models poses major hurdles to development and implementation of algorithms, it is indeed the most appealing field of application. The system developed within the Nefertiti project supports retrieval of 3D models based on both geometry and appearance (i.e. colour and texture) [4]. Also Kolonias et al. have used dimensions of the bounding box (i.e. its aspect ratios) and a binary voxel-based representation of geometry [5]. They further relied on a third feature, namely a set of paths, outlining the shape (*model routes*). In [6] a method is proposed to select feature points which relies on the evaluation of Gaussian and median curvature maxima, as well as of torsion maxima on the surface. In [7], Elad et al. use moments (up to the 4-7th order) of surface points as basic features to support retrieval of 3D models. Differently from the case of 2D images, evaluation of moments is not affected by (self-)occlusions.

In this paper we report on the use of three models for representing the content of a 3D object for the purpose of supporting retrieval by object similarity.

The three models are based on projection of surface curvature information and spin images. Projection of surface curvature information is obtained by warping object surface until it becomes a function on the sphere. Then, information about curvature is projected onto a 2D curvature map that retains information about curvature distribution on the original object surface. Content of the 2D curvature map is described using two different techniques: histograms of map tiles and weighted walkthroughs of map regions.

The third model for content representation is based on spin images. These capture geometric properties of 3D objects regardless of surface curvature. Since object description based on spin images entails a huge amount of information, feature extraction and clustering techniques are used to meet the specific storage and efficiency requirements of content-based retrieval.

The paper is organized as follows: Sec.2 describes some pre-processing steps that are necessary to apply both curvature maps and spin images techniques; Sec.3 expounds on computation of curvature maps from 3D object meshes; Sec.4 describes how to use curvature maps for description of 3D objects content; Sec.5 introduces extraction of spin images from 3D objects; Sec.6 describes how to use spin images for description of 3D objects content; finally in Sec.7 a comparative analysis among the proposed techniques is presented.

## 2 Preprocessing

High resolution 3D models obtained through scanning of real world objects are often affected by high frequency noise, due to either the scanning device or the subsequent registration process. Hence, smoothing is required to cope with such models for the purpose of extracting their salient features.

Selection of a smoothing filter is a critical step, as application of some filters entails changes in the shape of the models. For instance, mean or Laplacian smoothing cause shrinking of the model (a known problem, which has been pointed out – for example – in [6]). In Laplacian smoothing, every vertex  $x$  is moved from its original location by an offset  $\Delta(x)$ ; the offset is determined as a function of the neighbouring vertices of  $x$ , and a parameter  $\lambda$  controls the strength of the filter. To avoid shrinking, we adopted the filter first proposed by Taubin [8]. This filter, also known as  $\lambda|\mu$  filter, operates iteratively, and interleaves a Laplacian smoothing weighed by  $\lambda$  with a second smoothing weighed with a negative factor  $\mu$  ( $\lambda > 0$ ,  $\mu < -\lambda < 0$ ). This second step is introduced to preserve the model's original shape.

An additional pre-processing step is employed to reduce the complexity of the model (in terms of the number of vertices). To this end, an algorithm performing an iterative contraction of vertex pairs (i.e. edges) is used: first, all edges are ranked according to a cost metric; then, the minimum cost vertex pair is contracted; finally, the costs are updated [9]. The algorithm is iterated until a predefined stop criterion is met: In our experiments, the stop criterion was set in terms of the number of polygons of the final model.

## 2.1 Curvature Estimation

Estimation of surface curvature at a generic vertex  $v_i$  of the mesh is accomplished by considering variations of surface normal over the *platelet*  $V^{v_i}$  of vertex  $v_i$ . This guarantees less sensitivity to noise and acquisition errors.

In particular, surface curvature in correspondence with the  $i$ -th vertex  $v_i$  of the mesh  $\mathcal{M}$  is estimated by considering versor  $v_i^\perp$ , that is, the normal to  $\mathcal{M}$  at point  $v_i$ . Then, the *platelet*  $V^{v_i}$  of vertex  $v_i$  is considered. This is defined as the set of all mesh vertices around  $v_i$ . Given a generic vertex of the platelet  $v_j \in V^{v_i}$  let  $v_j^\perp$  be the normal to  $\mathcal{M}$  at point  $v_j$ . Mesh curvature  $\gamma_{v_i}$  at vertex  $v_i$  is estimated as:

$$\gamma_{v_i} = \frac{1}{2} \frac{\sum_{v_j \in V^{v_i}} |v_i^\perp - v_j^\perp|}{|V^{v_i}|} \quad (1)$$

It can be shown that with this definition, the value of  $\gamma_{v_i}$  is always in  $[0, 1]$ .

## 3 Curvature Maps

Given a 3D object, construction of its curvature map relies on warping object surface until it becomes a function on a sphere and then projecting curvature information onto a 2D image. Mesh deformation is obtained by iteratively applying a smoothing operator to the mesh. In general, application of a smoothing operator is accomplished by updating the position of each vertex of the mesh according to the following formula:

$$\mathcal{M}(v_i) \otimes \omega = \frac{\mu}{\sum_{v_j \in V^{v_i}} w_j} \sum_{v_j \in V^{v_i}} w_j * v_i - v_i \quad (2)$$

being weights  $\omega = \{w_j\}$  characteristic of each operator and  $\mu$  a parameter used to control the amount of motion of each vertex and to guarantee stability and continuity of the smoothing process.

Under the assumption of low  $\mu$  values, the iterative application of the smoothing operator to every vertex of the mesh is equivalent to an elastic deformation process. During the deformation process each vertex of the mesh should be moved in order to satisfy two sometimes opposite requirements: mesh regularization and curvature minimization. As demonstrated in previous work [10], application of *Laplacian Smoothing*, *Taubin Smoothing*, or *Bilaplacian Flow* operators increases mesh regularization but may result in unnatural deformations of the original mesh. Differently, application of *Mean Curvature Flow* operator doesn't guarantee mesh regularization.

To achieve both regularization and smoothing of the original mesh, the proposed solution develops on the application of two distinct operators at each step of the iterative deformation process. In particular, Laplacian and Gaussian smoothing operators are used in combination to achieve both mesh smoothing and regularization. Application of the two operators is iterated until the average value of vertex motion falls below a predefined threshold  $\tau$ .

### 3.1 Mapping

Projection of a curved surface is a well known problem in cartography [11]. There are many different projections used to map (a part of) the globe onto a plane, but their description is far beyond the scope of this paper. In our approach, we have selected the Archimedes projection (also known as the Lambert equal-area projection). Similarly to the Mercator projection, the Archimedes projection is a cylindrical projection. In particular, it is the projection along a line perpendicular to the axis connecting the poles and parallel to the equatorial plane. Thus, a point on the sphere with latitude  $\Theta$  and longitude  $\Phi$ , is mapped into the point on the cylinder with the same longitudinal angle  $\Theta$  and height  $\sin(\Phi)$  above (or below) the equatorial plane.

A major advantage of the Archimedes projection is that it is an area preserving projection: all regions on the surface of the sphere are mapped into regions on the map having the same area. This guarantees that, regardless of the position on the sphere, the relevance of any region is the same both on the sphere and on the map.

## 4 From Curvature Maps to Content Descriptors

Ideally, once a 3D model is represented through a 2D curvature map, any approach supporting image retrieval by visual similarity could be used to evaluate the similarity between two 3D models. In fact, this can be achieved by computing the similarity of the corresponding maps.

In the proposed approach, information about curvature maps is captured at two distinct levels: tiles obtained by a uniform tessellation of the map, and homogeneous regions obtained by segmenting the map. In the former case, we use histograms to capture global properties of map tiles, whereas in the latter case we rely on weighted walkthroughs to describe spatial arrangement and local properties of regions on the map. Details on the two techniques are provided hereafter.

### 4.1 Histogram-Based Description of Map Tiles

A generic histogram  $H$  with  $n$  bins is an element of the histogram space  $\mathcal{H}^n \subset \mathbb{R}^n$ . Given an image and a quantization of a feature space, histogram bins count the number of occurrences of points of that quantized feature value in the image.

Histograms also support a multi-resolution description of image features. Given a partitioning of an image into  $n$  fine-grained tiles, histograms provide a representation for the content of each of these tiles.

In order to compute the similarity between two histograms, a norm must be defined in the histogram space. In our experiments the *Kolmogorov-Smirnov* distance was adopted. Thus, the distance between two histograms  $H$  and  $H'$  is computed as follows:

$$\mathcal{D}_{KS}(H, H') = \max_i(\check{h}_i, \check{h}'_i) \quad (3)$$

being  $\check{h}_i$  and  $\check{h}'_i$   $i$ -th element of the cumulated histogram of  $H$  and  $H'$ , respectively (i.e.  $\check{h}_i = \sum_{k=1}^i h_k$ ).

Computing the distance between two maps requires to find the best tiles correspondence function. This is defined as the permutation  $p : \{1, \dots, n\} \rightarrow \{1, \dots, n\}$  that minimizes the sum of distances between corresponding tiles.

The solution  $p$  is approximated through a *greedy search* approach that requires to scan all tiles in the first map in a predefined order and associate to each tile the most similar tile not yet associated in the second map. This pairwise NN association yields a suboptimal solution.

## 4.2 Weighted Walkthroughs Description of Map Tiles

Description of map content through histograms is not able to capture neither the spatial arrangement nor the local properties of individual regions of the map. In some cases this can be a limitation, since information about individual regions and their spatial arrangement in the map is strictly related to information about shape and structure of the original 3D mesh. To overcome these limitations, the coarse description of map content is complemented with a local approach capturing local properties of individual regions in the map as well as their spatial arrangement.

Local description of map content is based on weighted walkthroughs technique [12]. In particular, description of map content is accomplished by segmenting the map into regions characterized by uniform curvature values. For each region, information about region area and average curvature is retained. Furthermore, for each pair of regions, their relative position is captured through a  $3 \times 3$  array corresponding to the weighted walkthroughs for the two regions.

The use of weighted walkthroughs enables description of map content in the form of an attributed relational graph. Graph vertices correspond to regions of the map and are labelled with the region's area and average curvature. Graph edges retain information about the relative position of regions they link and are labelled with the corresponding  $3 \times 3$  weighted walkthroughs.

The descriptor of content of a generic map can be represented as  $\langle R, f, w \rangle$ , being  $R$  the set of regions in the map,  $f$  the set of visual features capturing the appearance of each region (in our case region area and average curvature), and  $w$  the set of weighted walkthroughs capturing the relative position of each region pair.

Computation of the similarity between two descriptors of map local content is equivalent to an error correcting subgraph isomorphism problem [13], which is an NP-complete problem with exponential time solution algorithms [14].

In the proposed approach, identification of the optimal node association function is accomplished through the technique presented in [15]. This is based on a look-ahead strategy that extends classical state-space search approaches.

## 5 Spin Images

Spin images were introduced by Johnson and Hebert to support recognition of single objects in complex scenes [16]. Basically, spin images encode the density

of mesh vertices projected onto an object-centred space: the three-dimensional mesh vertices are first mapped onto a two-dimensional space defined w.r.t. to the object itself; the resulting coordinates are then used to build a two-dimensional histogram.

More precisely, let  $O = \langle \mathbf{p}, \mathbf{n} \rangle$  an *oriented point* on the surface of the object, where  $\mathbf{p}$  is a point on the surface of the object and  $\mathbf{n}$  the normal of the tangent plane in  $\mathbf{p}$ . For a generic oriented point  $O$ , a *spin map* can be defined, which maps any point  $\mathbf{x}$  in the three-dimensional space onto a two-dimensional space according to the following formula (see also Fig. 1 for notation):

$$S_O(\mathbf{x}) \rightarrow [\alpha, \beta] = [\sqrt{\|\mathbf{x} - \mathbf{p}\|^2 - (\mathbf{n} \cdot (\mathbf{x} - \mathbf{p}))^2}, \mathbf{n} \cdot (\mathbf{x} - \mathbf{p})]$$

In other words, the oriented point defines a family of cylindrical coordinate systems, with the origin in  $\mathbf{p}$ , and with the axis along  $\mathbf{n}$ . The spin map projection of  $\mathbf{x}$  retains the radial distance ( $\alpha$ ) and the elevation ( $\beta$ ), while it discards the polar angle.

To produce a *spin image* of an object, a spin map is applied to points comprising the surface of the object. Hence, given a mesh representation of the object, the spin image can be obtained by applying the map to the vertices comprising the mesh. A simple binary image can be obtained by discretizing the projected coordinates and by setting the corresponding point on the image. However, more refined grey-level spin images encoding a measure of the density of vertices that insist upon the same image point are usually employed. To construct such an image, the projected coordinates  $\alpha$  and  $\beta$  of each mesh vertex are used to update the two-dimensional histogram  $I(i, j)$  (i.e. the spin image) according to a bi-linear interpolation scheme that spreads the contribution of each vertex over the nearest points on the grid induced by the quantization of the image space.

Most outstanding characteristics of spin images are invariance to rigid transformations (as a consequence of the adoption of an object-centred coordinate system), limited sensitivity to variations of position of mesh vertices (which might result from the adoption of different sampling schemes), flexibility (since no hypotheses are made on the surface representation), and ease of computation.

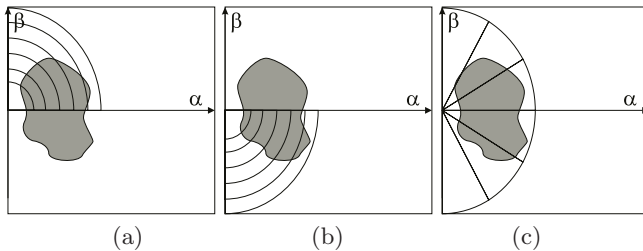


**Fig. 1.** Given an oriented point  $\langle \mathbf{p}, \mathbf{n} \rangle$  on the object surface, a generic point  $\mathbf{x}$  is mapped on point  $[\alpha, \beta]$  on the spin map, being  $[\alpha, \beta]$  the radial distance and the elevation of  $\mathbf{x}$  w.r.t. to  $\langle \mathbf{p}, \mathbf{n} \rangle$ . a) the object centred 3D coordinates system, and b) the spin map coordinate system.

## 6 From Spin Images to Content Descriptors

Spin images provide a powerful means to describe three-dimensional objects. However, the fact that many spin images are typically produced for a single object, and the fact that each image implies considerable storage requirements, prevent us to use them directly as descriptors for retrieval purposes. Therefore, we decided to rely on more compact descriptions extracted from spin images, synthesizing the content of each spin image. Our descriptor for spin images was inspired by region descriptions such as grid-based techniques or the shape matrix [17]. Instead of sampling the shape at the intersection between radial and circular lines, we decided to measure the relative density encompassed by each of the regions defined by those lines, so as to provide a more precise description of the spin image. We have defined three independent sets of regions for the spin image: sectors of circular crowns for both the half-plane with  $\beta > 0$  and the half-plane with  $\beta < 0$ , and circular sectors. Each of these sets defines a descriptor ( $C^p = \langle cp_1, \dots, cp_{np} \rangle$ ,  $C^n = \langle cn_1, \dots, cn_{nn} \rangle$ , and  $S = \langle s_1, \dots, s_{ns} \rangle$ , respectively), whose components represent the amount of surface points (or vertices) whose projections fall within the corresponding crown/sector.

Based on results of some preliminary experiments we chose  $np = nn = ns = 6$  as these represent a satisfactory trade-off between compactness and selectivity of the representation. Hence, a 18-dimensional descriptor  $D = \langle C^p, C^n, S \rangle$  is evaluated for each spin image.



**Fig. 2.** Compound object descriptors comprise descriptors for a)  $np$  crowns in the half-plane  $\beta > 0$ , b)  $nn$  crowns in the half-plane  $\beta < 0$ , c)  $ns$  sectors. In our experiments  $np = nn = ns = 6$ .

In order to avoid use of one spin image descriptor for each mesh vertex, spin image descriptors are subject to clustering. For this purpose we relied on fuzzy clustering [18], which is an extension to *c*-means procedure that avoids partitioning feature vectors into *hard* or *crisp* clusters. Through clustering, we represent the original set of spin image descriptors  $\{SI_1, \dots, SI_m\}$  ( $SI_i \in \mathbb{R}^{18}$ ) – each descriptor being associated with one mesh vertex – with a compact set represented by the clusters’ centers.

Computation of the similarity between two 3D objects is accomplished by comparing their descriptors, each descriptor being in the form of a set of cluster centers  $\mathcal{D} = \{D_i, i = 1 \dots\}$ .





**Fig. 3.** Retrieval of Mercur statues using spin images. All models of the Mercur statues are retrieved first, followed by models of other statues which display similar shapes.

Computing the distance  $\Delta$  between two descriptors  $\mathcal{D}$  and  $\hat{\mathcal{D}}$  requires to find the best cluster-center correspondence function. This is defined as the permutation  $p : \{1, \dots, l\} \rightarrow \{1, \dots, k\}$  that minimizes the sum of distances between corresponding cluster centers.

## 7 Comparative Analysis

Approximately 300 models were collected to build the test database. These comprise four classes of models: taken from the web, manually authored (with a 3D CAD software), high quality 3D scans from the De Espona 3D Models Encyclopedia<sup>1</sup>, and variations of the previous three classes (obtained through geometric deformation or application of noise, which caused points surface to be moved from their original locations).

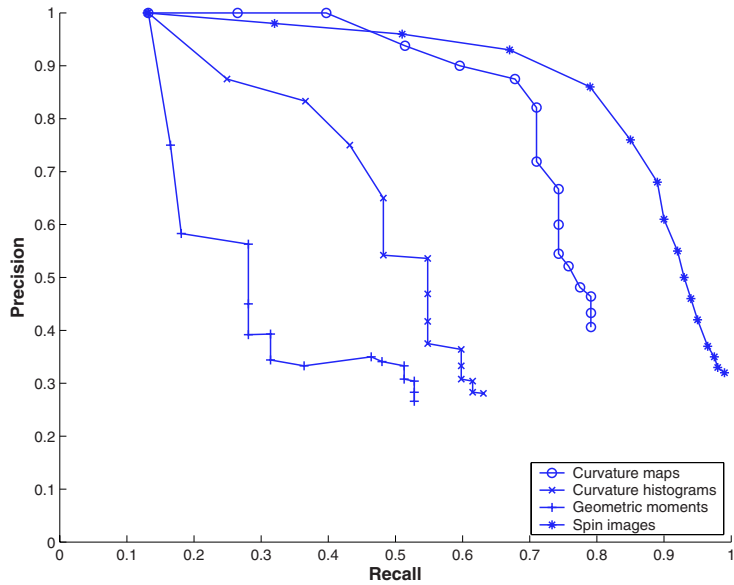
Fig.3 shows a retrieval example using spin images. The query is the model of a statue portraying Mercur. The result set displays all models of the Mercur statue in the first five positions. In general, all retrieved models feature similar shapes, characterized by a main body and protrusions that resemble Mercur's elongated arm and leg. Fig.4 shows a retrieval example where the model of a Satyr's bust is used as a query. The result set displays all models of the Satyr's bust in the first five positions. Other models reproducing busts are also retrieved.

Among the different techniques reviewed in Section 1, we selected the curvature histograms [19] and moments of surface points [7] for a comparative as-

<sup>1</sup> <http://www.deespona.com>



**Fig. 4.** A retrieval example, using the model of a Satyr's bust as the query. Other models of the Satyr's bust are retrieved first, followed by models of other busts.



**Fig. 5.** Comparison of precision/recall figures for the four methods: curvature histograms, moments, weighted walkthroughs of curvature maps and spin images.

assessment. Performance comparison is assessed through four sample queries that were submitted to each of the four retrieval engines. Average precision vs. recall curves are shown in Fig. 5.

The comparative evaluation shows that retrieval based on spin images performs better than all the other three approaches. In particular, performance of approaches based on curvature histograms and 3D moments is particularly critical. This may be accounted to the fact that these two methods only provide a global description of the object, and this is often inappropriate for discrimination of different models.

## References

1. S. Mahmoudi, M. Daoudi, "3D models retrieval by using characteristic views", in *Proc. of 16th Int'l Conf. on Pattern Recognition*, Vol.2, pp.457-460, 11-15 Aug, 2002.
2. R. Ohbuchi, M. Nakazawa, T. Takei, "Retrieving 3D Shapes based on Their Appearance", in *Proc. of MIR'03*, Berkeley, CA, USA, Nov. 2003, pp.39-46.
3. H.P. Kriegel, T. Seidl, "Approximation-Based Similarity Search for 3D Surface Segments", *GeoInformatica Journal*, 2(2):113-147, Kluwer Academic Publisher, 1998.
4. E. Paquet, M. Rioux, "Nefertiti: a query by content system for three-dimensional model and image database management", *Image Vision and Computing*, 17(2):157-166, 1999.
5. I. Kolonias, D. Tzovaras, S. Malassiotis, M. G. Strintzis, "Content-Based Similarity Search of VRML Models Using Shape Descriptors", in *Proc. of International Workshop on Content-Based Multimedia Indexing*, Brescia (I), September 19-21, 2001.
6. F. Mokhtarian, N. Khalili, P. Yeun, "Multi-scale free-form 3D object recognition using 3D models", *Image and Vision Computing*, 19(5):271-281, 2001.
7. M. Elad, A. Tal, S. Ar, "Content Based Retrieval of VRML Objects - An Iterative and Interactive Approach", *EG Multimedia*, September 2001, 97-108.
8. G. Taubin, "A Signal Processing Approach to Fair Surface Design", *Computer Graphics (Annual Conference Series)*, 29:351-358, 1995.
9. M. Garland, "Multiresolution Modeling: Survey & Future Opportunities", in *Proc. of Eurographics'99*, September 1999.
10. A.G. Belyaev, I.A. Bogaevski, Y. Ohtake, "Polyhedral Surface Smoothing with Simultaneous Mesh Regularization", in *Proc. of Geometric Modeling and Processing 2000. Theory and Applications*, Hong Kong, China, April 10-12, 2000, pp.229-237.
11. J.P. Snyder, L.M. Bugayevski, *Map projections - a reference manual*, Taylor & Francis, 1995.
12. A. Del Bimbo, E. Vicario, "Using weighted spatial relationships in retrieval by visual contents", in *Proc of the IEEE Workshop on Content-Based Access of Image and Video Libraries (CBAIVL'98)*, Santa Barbara, CA, pp.35-39, June 1998.
13. M.A. Eshera, K.-S. Fu, "A Graph Measure for Image Analysis", *IEEE Trans. on Systems, Man and Cybernetics*, 14(3):398-407, May/June 1984.
14. M.R. Garey, D. Johnson, "Computer and Intractability: A Guide to the Theory of NP-Completeness", Freeman, San Francisco, 1979.
15. S. Berretti, A. Del Bimbo, E. Vicario, "Efficient Matching and Indexing of Graph Models in Content-Based Retrieval", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(10):1089-1105, October 2001.

16. A. E. Johnson, M. Hebert, "Using Spin-Images for efficient multiple model recognition in cluttered 3-D scenes", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):433-449, 1999.
17. A. Goshtasby, "Description and discrimination of planar shapes using shape matrices", *IEEE Transactions on PAMI*, 7:738-743, 1985.
18. J.C.Bezdek, J.Keller, R.Krishnapuram, N.R.Pal, *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*, Kluwer Academic Publisher, Boston, 1999.
19. J.-Ph. Vandeborre, V. Couillet, M. Daoudi, "A Practical Approach for 3D Model Indexing by combining Local and Global Invariants", in *Proc. of the 1st Int'l Symp. on 3D Data Processing, Visualization, and Transmission (3DPVT'02)*, 2002.