An Experimental Comparison of Feature-Based 3D Retrieval Methods

Benjamin Bustos Daniel Keim Dietmar Saupe Tobias Schreck Dejan Vranić
Department of Computer and Information Science, University of Konstanz
{bustos|keim|saupe|schreck|vranic}@informatik.uni-konstanz.de

Abstract

3D objects are an important type of multimedia data with many promising application possibilities. Defining the aspects that constitute the similarity among 3D objects, and designing algorithms that implement such similarity definitions is a difficult problem. Over the last few years, a strong interest in methods for feature-based 3D similarity search has arisen, and a growing number of competing algorithms for content-based retrieval of 3D objects have been proposed. We present an extensive experimental evaluation of the retrieval effectiveness and efficiency of a large part of the current state-of-the-art feature-based methods for 3D similarity search, giving a contrasting assessment of the different approaches.

1 Introduction

The development of effective and efficient similarity search methods for multimedia data is an important research issue due to the growing amount of digital audio-visual information that is becoming available. In digital libraries that are built from heterogenous data sources, typically consistent annotations are not available in order to organize and access the objects. Therefore, automatic content-based methods for similarity estimation of multimedia objects are required. In the case of 2D images along with the growth of available data volumes, a wealth of similarity notions and retrieval systems has evolved. A similar development can be expected for 3D data, as 3D objects are powerful means for information dissemination with applications in such important fields as design and construction, education, simulation and entertainment.

Similarity search methods for 3D objects have to address a number of problems in order to achieve desirable invariance properties (i.e., position, scale and rotation). They also have to select suitable object characteristics for similarity estimation. Usually, a *feature vector* (FV) approach is used for performing similarity search. Already, there exist a variety of proposed methods that can be used to implement 3D similarity search systems. As these methods are rather new, to date few comprehensive experimental or theoretical studies contrasting the different methods exist. We have developed a retrieval system that implements many different 3D FVs from our own as well as other researchers' work. In this paper, we present a survey of the FVs we have implemented in our system, and empirically evaluate their retrieval performance based on extensive similarity search experiments conducted on a large classified database. The database is representative for the variety of 3D models one may find on the World Wide Web today in VRML or similar formats.

2 Similarity search of 3D objects

Given that it is not clear how to use geometry directly for similarity search, in most methods for similarity search the 3D data is transformed in some way to obtain a *numerical description* for indexing and retrieval, also referred to as *feature vectors*, or *FVs*. The basic idea is to extract numeric values that describe the objects under a certain geometric aspect, and to infer the similarity of the models from the distance of these FVs in some vector space.

2.1 Feature vector paradigm

The usage of feature vectors is the standard approach in multimedia retrieval. Based on the real valued vectors describing the objects in a database, a similarity query for a query object q is usually executed as a k-NN query, returning the k objects whose FVs have the smallest distance to q under a certain distance metric, sorted by increasing distance to the query.

An important family of such distance metrics in vector spaces is the *Minkowski* (l_s) family of distances, defined as $l_s(\vec{x},\vec{y}) = \left(\sum_{1\le i\le d}|x_i-y_i|^s\right)^{1/s}$, $\vec{x},\vec{y}\in I\!R^d, s\ge 1$. Examples of these distance functions are l_1, which is called *Manhattan distance*, l_2, which is the *Euclidean distance*, and $l_\infty = \max_{1\le i\le t}|x_i-y_i|$, which is called the *maximum distance*. Several extentions to the Minkowski distances have been studied, like the weighted Minkowski distance, where a weighting vector is assigned to the vector

component distances, or the Mahalanobis distance [7, 13], which engages a weight matrix to reflect cross-component similarity relationships between FVs.

2.2 Invariance requirements and the Principal Component Analysis

Several requirements that suitable methods for 3D similarity search should fulfill can be identified. The methods should be *invariant* to changes in the orientation, translation, reflection, and scale of 3D models in their reference coordinate frame. They should also be *robust* with respect to changes of the level-of-detail and to small changes of the geometry and topology of the models. Invariance and robustness properties can be achieved implicitly by those methods that consider relative object properties or that integrate a similarity measure over the space of transformations [12]. Otherwise, these properties can be approximated by a preprocessing normalization step, which transforms the objects so that they are represented in a canonical reference frame. In such a reference frame, directions and distances are comparable between different models. The predominant method for finding this reference coordinate frame is pose estimation by principal components analysis (PCA) [10, 19]. The basic idea is to align a model by considering its center of mass as the coordinate system origin, and its principal axes as the coordinate axes. While the majority of proposed methods employs PCA in some form or another, some authors have stability concerns with respect to the PCA as a tool for 3D retrieval. On the other hand, omitting orientation information also omits valuable object information. For a more detailed discussion see [2, 15, 8, 14].

3 Implemented 3D feature extraction algorithms

Geometric 3D moments. Statistical moments μ are scalar values that describe a distribution f. Parameterized by their order, moments represent a spectrum from coarse-level to detailed information of the given distribution [10]. In the case of 3D solid objects, which may be interpreted as a density function $f(x, y, z)$, the moment $\mu_{i,j,k}$ of order $n = i + j + k$ in continuous form is defined by $\mu_{ijk} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x, y, z)x^i y^j z^k dxdydz$. As is well known, the complete (infinite) set of moments uniquely describes a distribution and vice versa. In its discrete form, the moment formula becomes $\mu_{ijk} = \sum_{p\in P} {x_p}^i {y_p}^j {z_p}^k$ over all considered points from P in the distribution. In [10], it is proposed to use the centroids of all triangles of a triangulated model (weighted with the mass of the respective triangle) as input to moment calculation (*moments FV*), while in [16] object points found by an uniform ray-based scanning scheme serve as the input (*ray-moments FV*). Because moments are not invariant with respect to translation, rotation and scale, PCA and scale normalization have to be applied prior to moment calculation. A FV can then be constructed by concatenating certain statistical moments of an object, e.g., all moments of order up to some value n.

Cords based FV. A FV that combines information about the spatial extent and orientation of a 3D object is given in [10] (*cords FV*). The authors define a "cord" as a vector that runs from an object's center of mass to the centroid of a bounded region of the object, usually a triangle. For all object surfaces, such a cord is constructed. The FV is then built by calculating two histograms for the angles between the cords and the object's first two principal axes each (measuring orientation), and one histogram for the distribution of the cord length (measuring spatial extention). All three histograms are normalized by the number of cords and together make up the feature vector. Using the principal axes, the methods is invariant to rotation and translation. It is also invariant to scale, as the length distribution is binned to the same number of bins for all objects. It can be inferred that the methods is not invariant to non-uniform tessellation changes. No quantitative retrieval results were given for this FV in [10].

Shape distribution with D_2. In [9], it is proposed to characterize the shape of a 3D object as a probability distribution sampled from a shape function, which reflects geometric properties of the object. The algorithm calculates histograms called *shape distributions*, and estimates similarity between two shapes by any metric that measures distances between distributions (e.g., Minkowski distances). The authors state that, depending on the shape function employed, shape distributions possess rigid transformation invariance, robustness against small model distortions, independence of object representation, and provide for efficient computation. The shape functions studied by the authors include the distribution of angles between three random points on the surface of a 3D object, and the distribution of Euclidean distances between one fixed point (specifically, the centroid of the boundary of the object was taken) and random points on the surface. Furthermore, they propose to use the Euclidean distance between two random points on the surface, the square root of the area of the triangle between three random points on the surface, or the cube root of the volume of the tetrahedron between four random points on the surface. Where necessary, a normalization step is applied for differences in scale. As the analytic computation of distributions is feasible only for certain combinations of shape functions and models, the authors perform appropriate random sampling of many values from an object, and construct a histogram from these samples to de-

scribe the object shape. The authors perform retrieval experiments and report that the best experimental results are achieved using the distance function (distance between two random points on the surface), and using the l_1 norm of the probability density functions, which are normalized by aligning the mean of each two histograms to be compared (*D2 shape distribution FV*).

Method based on surface geometry. A methods for 3D retrieval proposed within the MPEG-7 framework for multimedia content description and reflecting curvature properties of 3D objects is presented in [20]. The *shape spectrum FV* is defined as the distribution of the *shape index* for points on the surface of a 3D object, which in turn is a function of the two principal curvatures at the respective surface point. The shape index gives the angular coordinate of a polar representation of the principal curvature vector, and it is implicitly invariant with respect to rotation, translation and scale. Because the shape index is not defined for planar surfaces, but 3D objects are usually approximated by polygon meshes, the authors suggest approximating the shape index by fitting quadratic surface patches to all mesh faces based on the respective face and all adjacent faces, and using this surface for shape index calculation. To compensate for potential estimation unreliability due to (near) planar surface approximations and (near) isolated polygonal face areas, these are excluded from the shape index distribution based on a threshold criterion, but their relative area is cumulated in two other attributes named *planar surface* and *singular surface*. These attributes together with the shape index histogram form the final FV.

Silhouette FV. A method called *silhouette FV* [4] characterizes 3D objects in terms of their silhouettes that are obtained from canonical renderings. The objects are first PCA-normalized and scaled into a unit cube that is axis-parallel to the principal axes. Then, parallel projections onto three planes, each orthogonal to one of the principal axes, are calculated. The authors propose to obtain FVs by concatenating Fourier approximations of the three resulting contours. To obtain such approximations, a silhouette is sampled by placing a certain number of equally-spaced sequential points on the silhouette, and regarding the Euclidean distance between the image center and the consecutive contour points as the sampling values. These sampling values in turn constitute the input to the Fourier transform. The concatenation of the magnitudes of low-frequency Fourier coefficients of the three contour images then gives the silhouette object description.

Depth buffer FV. Also in [4], another image-based FV is proposed. The so-called *depth buffer FV* starts with the same setup as the silhouette FV: The model is oriented and scaled into the canonical unit cube. Instead of three silhouettes, six gray-scale images are rendered using parallel projection, each two for one of the principal axes. A pixel attribute represents the distance between the object and the viewing plane measured along a corresponding direction that is perpendicular to the viewing plane. These images correspond to the concept of z- or depth-buffers in computer graphics. After rendering, the 6 images are transformed using the standard 2D discrete Fourier transform, and the magnitudes of low-frequency coefficients of each image contribute to the depth buffer FV.

Extension-based methods. In [18, 16] the authors propose a FV extraction framework that is based on taking samples from a PCA-normalized 3D object by means of rays emitted from the center of mass O of an object in uniformly distributed directions $\mathbf{u}$ (directional unit vectors). For all such rays in direction $\mathbf{u}$, starting from O the last intersection point $i(\mathbf{u})$ with a polygon p of the object is found, if such a point exists. If this point exists, the distance $r(\mathbf{u}) = ||i(\mathbf{u}) - O||$ is calculated, as well as the scalar product $x(\mathbf{u}) = |\mathbf{u} \cdot n(\mathbf{u})|$, where $n(\mathbf{u})$ is the normal vector of the respective polygon. In the first proposed FV, which compares spatial extent, the distances $r(\mathbf{u})$ make up the components of the so-called *ray-based* feature vector. A second FV is obtained by setting the scalar products $x(\mathbf{u})$ as the feature components (called *shading-based FV*). The sample values taken by these functions from an object can be seen as instantiations of a function on the sphere. The authors propose, instead of using the absolute sample values, to apply the 3D Fourier transform on the samples, and to take the magnitudes of low-frequency coefficients as an embedded multi-resolution object description. *Spherical harmonics* [3] are proposed as the basis functions for the transform, and the corresponding FVs are called *rays-SH* and *shading-SH FV*, respectively. These complex coefficients come as complex conjugate pairs with equal magnitude. Thus, for each pair only one magnitude is used in the feature vector. In extention to using either $r(\mathbf{u})$ or $x(\mathbf{u})$, also the combination of both measures in a complex function $y(\mathbf{u}) = r(\mathbf{u}) + i \cdot x(\mathbf{u})$ is considered by the authors. The FV based on the spherical harmonics transform of $y(\mathbf{u})$ is called the *complex FV*.

Rotation invariant point cloud FV. In [5], the authors present a method that relies on PCA registration but also is invariant to rotations of 90 degrees along the found principal axes. To construct the FV, an object is centered and oriented into the canonical coordinate frame using PCA, and scaled into the unit cube with origin at the center of mass of the object and axis parallel to the principal axes. The unit cube is then partitioned, e.g. into $7 \times 7 \times 7$ equally sized cubic cells, and for each cell the frequency of some large number

of points sampled uniformly from the surface and that lie in the respective cell is determined. To reduce the size of the FV, which until now consists of 343 values, all grid cells are associated with one of 21 equivalence classes based on their location in the grid. Therefore, all cells that coincide when performing arbitrary rotations of 90 degrees along the principal axes are grouped together in one of the classes. For each equivalence class, the frequency data contained in the cells belonging to the respective equivalence class is aggregated, and the final FV of dimensionality 21 is obtained. The authors present retrieval performance results on a 3D database, on which $7 \times 7 \times 7$ is found to be the best grid dimensionality , but state that in general the optimal size of the FV may depend on the database chosen. In this paper, we refer to this method as the *rotational invariant FV*.

Methods based on surface voxelization. In [17], a FV based on the rasterization of a model into a voxel grid structure is presented, and the representation of this FV in either spatial or frequency domain is experimentally evaluated. The *voxel FV* is obtained by first subdividing the bounding cube of an object (after pose normalization) into $n \times n \times n$ equally sized voxel cells. Each of these voxel cells $v_{ijk}, i, j, k \in \{1, \ldots, n\}$ then stores the fraction $p_{ijk} = \frac{S_{ijk}}{S}$ of the object surface area S_{ijk} that lies in voxel v_{ijk}, where $S = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} S_{ijk}$ is the total surface area. The object's voxel cell occupancies constitute the FV of dimension n^3.

For similarity estimation with this FV, a metric can be defined in the spatial domain (*voxel FV*), or after a 3D Fourier-transform in the frequency domain (*3DDFT FV*). Then, only k low-frequency coefficients are used, enabling multi-resolution search.

Volume-based FVs. In the preceding method, triangle occupancies of a voxel grid made up a FV for object description. This approach is appropriate when dealing with polygon meshes without further conditions. Such meshes typically come from heterogenous sources, e.g., from the Internet (informally referred to as "polygon soups"). On the other hand, if the 3D models are known to bound a solid object, then also volumetric occupancies of the corresponding solid can be considered for FV construction. Several methods for similarity estimation based on voxelized volume data of normalized models have been proposed [10, 11, 6]. Another volume based FV is presented in [4]. Here, the six surfaces of an object's bounding cube are equally divided into n^2 squares each. Adding the object's center of mass to all squares, a total of $6n^2$ pyramid-like segments in the bounding cube is obtained. Assume that the polygon mesh bounds a solid object. The net proportion of volume occupied by the solid object in each segment of the bounding cube gives the components of the so-called *volume FV*.

Rotation invariant spherical harmonics FV. In [2], a FV based also on the spherical harmonics approximation of an object is proposed. An important characteristic of this technique is that more than just phase information is discarded in the spherical harmonics representation, in order to obtain complete rotation invariance without requiring pose estimation. This is possible since the energy in each frequency band of the spherical transform is rotation invariant [3]. Input to the transform is a binary voxelization of a polygon mesh into a grid with dimension $2R \times 2R \times 2R$, where each occupied voxel indicates the intersection of the mesh with the respective voxel. To construct the voxelization, the object's center of mass is translated into grid position (R, R, R) (grid origin), and the object is scaled so that the average distance of occupied voxels to the center of mass amounts to $\frac{R}{2}$, that is $\frac{1}{4}$ of the grids edge length. By using this scale instead of scaling it so that the bounding cube fits into the grid, it is possible to lose some object geometry. On the other hand, sensitivity with respect to outliers is expected to be reduced. The $8R^3$ voxels give rise to a binary function on the corresponding cube, which is written in spherical coordinates as $f_r(\theta, \phi)$ with the origin ($r = 0$) placed at the cube center. The binary function is sampled for radii $r = 1, \ldots, R$ and sufficiently many angles θ, ϕ to allow computation of the spherical harmonics representation of the spherical functions f_r. The FV consists of low frequency band energies of the functions $f_r, r = 1, \ldots, R$. By construction, it is invariant with respect to rotation about the center of mass of the object. In this paper, we refer to this FV as the *harmonics 3D FV*.

4 Experimental comparison of 3D feature vectors

4.1 Evaluation approach

Our evaluation experiments are based on a test database we have built, which contains 1,837 3D objects collected from the Internet. From this set, 472 objects were classified by shape into 55 different model classes (e.g., cars, planes, cups), and the rest of them were left as "unclassified". Each classified object of each model class was used as a query object, and the objects belonging to the same model class, excluding the query, were taken as the relevant objects. Figure 1 shows two examples of model classes: Formula 1 (F-1) cars, which contains 9 models, and sea animals, which contains 13 models (we omitted one model in the last model class for space reasons).

For comparing the effectiveness of the search algorithms, we use *precision vs. recall figures*, a standard evaluation technique for retrieval systems [1]. *Precision* is the fraction of the retrieved objects which are relevant to a given query, and *recall* is the fraction of the relevant objects which have

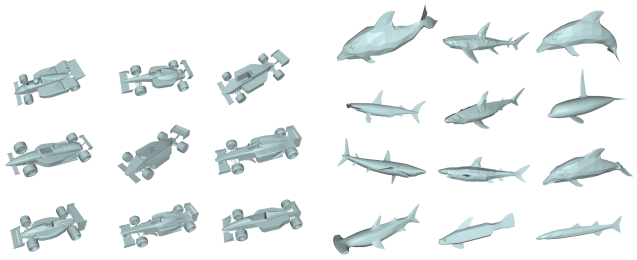


Figure 1. Formula 1 cars and sea animals model classes.

been retrieved from the database. We average the precision figures over all test queries at each of the 11 standard recall levels. In addition to the precision at multiple recall points, we also employ the widely used *R-precision* measure [1] (also known as *first tier*) for each query, which is defined as the precision when retrieving only the first N objects, where N is the number of relevant objects. The R-precision gives a single number to rate the retrieval performance.

We tested all FVs using different levels of resolution, from 3 up to 512 dimensions, selecting every possible dimensionality allowed by the parameters of each FV. The resulting values are averaged over all queries, except when otherwise stated. We used l_1 as the metric for distance computation, as this metric produced the best average retrieval results compared to the l_2 and l_{max} metric in our experiments. We apply our variant of the PCA [17] for those FVs that require object orientation normalization.

4.2 Effectiveness comparison between FVs

Average results. Figure 2 shows the precision vs. recall figures for all the implemented FVs. The average R-precision for each FV and its corresponding best dimensionality are also included in the chart. The most effective method in this experiment is the depth buffer with 366 dimensions. The difference of the average R-precision values between the best performing methods is small, which implies that in practice these FVs should all be suited equally well for retrieval of general polygonal objects. As a contrast, the effectiveness difference between the worst and the best FV is significant (up to a factor of 3). We observed that FVs that rely on consistent polygon orientation like shape spectrum or volume exhibit low retrieval rates, as consistent orientation is not guaranteed for many of the models retrieved from the Internet. Also, moment-based FVs seem to offer only limited discrimination capabilities.

Specific query classes. Most of the individual query classes from our database reflect the effectiveness ranking

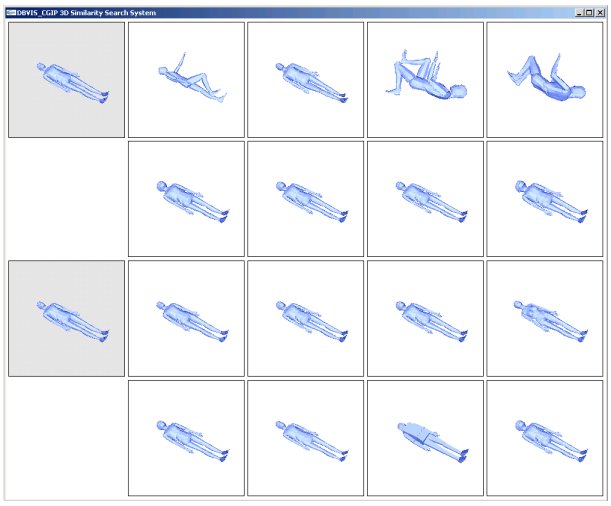
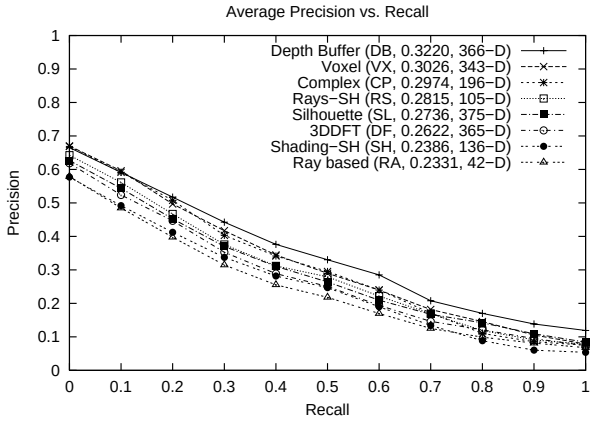


Figure 5. Example query in the humans class. The first query was conducted using the shape spectrum method and retrieves different articulations of the same model on the first ranks. The second query was conducted using the depth buffer method, and retrieves human models having the same articulation as the query object.

obtained from the database average, while certain shifts in the rankings are possible. Figures 3 and 4 illustrate two such query classes, namely one class with planes and one class with swords. The charts give the effectiveness results obtained with the methods for these query classes.

While the shape spectrum FV scores the worst on average, interestingly it achieves the best retrieval performance in a query class containing 56 models of humans (34% R-precision). As this FV considers the distribution of local curvature, it is able to retrieve human models that have different postures, while the other FVs retrieve only those models where model posture is roughly the same. Figure 5 illustrates a representative query in the humans class, showing the first retrieval results for the shape spectrum and the depth buffer FVs, in the first and second rows, respectively.

Level-of-detail. Robustness of the retrieval with respect to the level-of-detail in which models are given in a database is an important FV property. We test for this property using a query class that contains 7 different versions of the same model, in varying levels of resolution (specifically, models of a cow with 88 up to 5,804 faces). Except shape spectrum and cords, all FVs manage to achieve good to perfect retrieval results. Figure 6 illustrates.



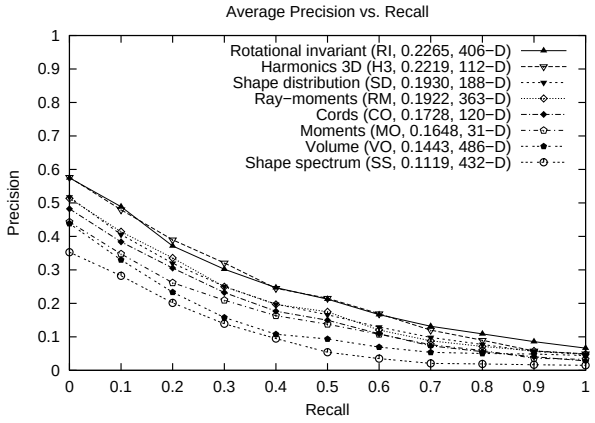
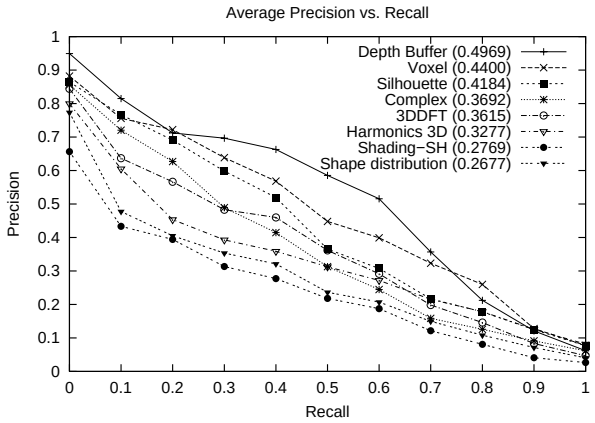


Figure 2. Average precision vs. recall with best dimensionality.



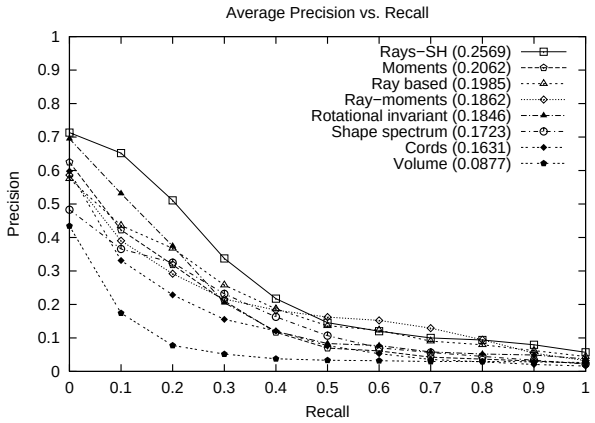
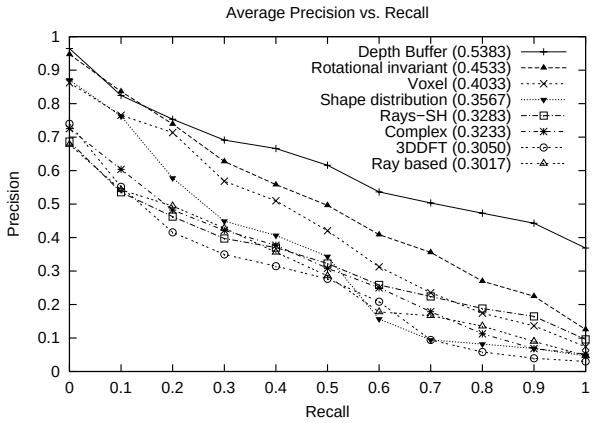


Figure 3. Average precision vs. recall, planes model class.



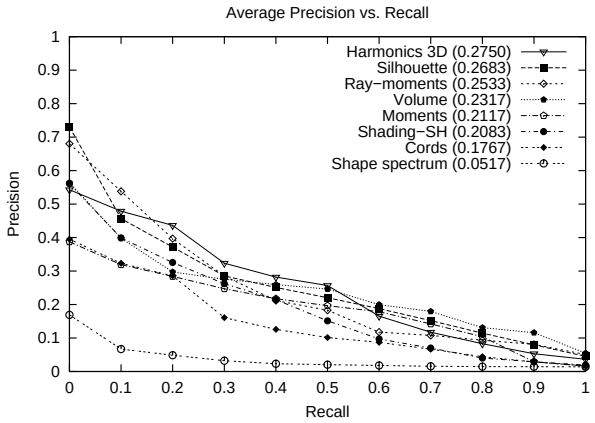


Figure 4. Average precision vs. recall, swords model class.

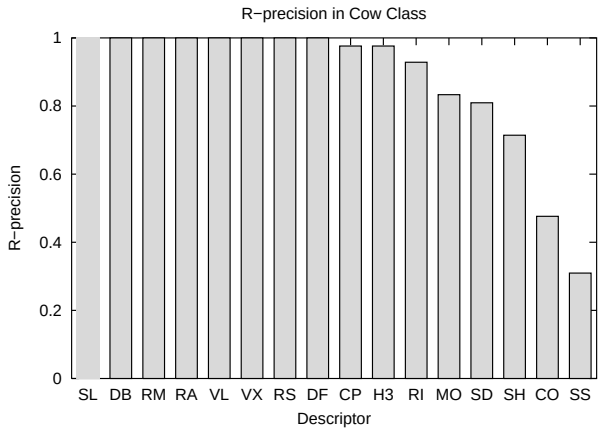


Figure 6. Robustness of the methods within the cow model class.

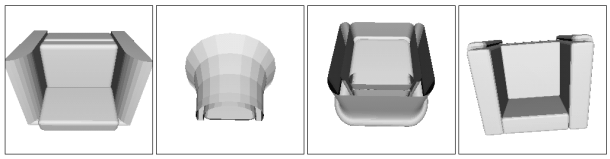


Figure 7. Alignment problem of PCA in the arm chais class. All objects are rendered with the camera looking at the center of mass along the least important principal axis, while the first principal axis is aligned horizontally.

Principal axes. PCA normalization is required by a large fraction of methods. For some model classes, the PCA gives alignment results that are not in accordance with the alignment a user would intuitively expect based on semantic knowledge of the objects. For example, we have defined a query class with 4 arm chairs, for which we observe that PCA alignment results are counterintuitive (cf. Figure 7). While we do not want to give an in-depth discussion of the PCA here, we note that in this query class an inherently rotational-invariant method (harmonics 3D) provides the best class-specific retrieval performance (see Figure 8).

Effects of dimensionality on retrieval. It is possible to calculate feature vectors at different resolutions, e.g., by specifying the number of rays with which to scan the objects, by specifying the number of Fourier coefficients to consider, etc. We are therefore interested in assessing the effect of FV resolution over the retrieval effectiveness. Figure 9 shows the effect of the FV dimensionality on the overall effectiveness. The figure shows that the effectiveness

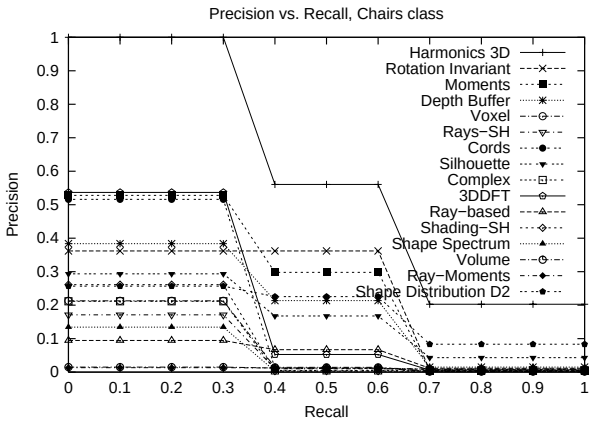


Figure 8. Precision-recall curves for the chairs model class. The rotation-invariant method harmonics 3D shows the best retrieval performance.

of the FVs first increases with dimensionality, but the improvement rate diminishes quickly for roughly more than 64 dimensions for most FVs (except for 3DDFT). It is interesting to note that the saturation effect is reached for most FVs at roughly the same dimensionality level. This is an unexpected result, considering that different FVs describe different characteristics of 3D objects.

5 Conclusions

We experimentally compared a wide variety of 3D FVs on a classified database of 3D objects, formed by models collected from the Internet, and we compared their retrieval performance using standard effectiveness measures from the Information Retrieval domain (precision vs. recall diagrams and the R-precision values). Our experimental comparison shows that there is a number of them that have good average effectiveness and work well in most cases (e.g., depth buffer, voxel and complex FVs). Other methods work well with some specific model classes, and some of them are effective when the normalization step using PCA is not effective (e.g., harmonics 3D with the chair model class). Regarding to the level-of-detail, the experimental results show that, with a few exceptions like shape spectrum and volume, all FVs can be considered robust, as they can retrieve similar objects with different level of detail.

Acknowledgments

This work was partially funded by the Deutsche Forschungsgemeinschaft (DFG), Projects No. KE 740/6-1

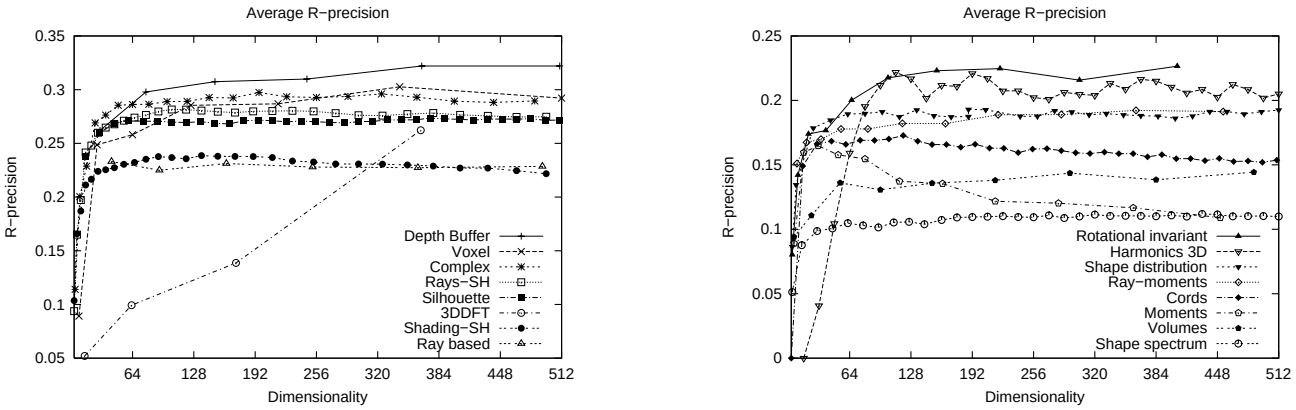


Figure 9. Dimensionality vs. R-precision.

and No. SA 449/10-1, within the strategic research initiative "Distributed Processing and Delivery of Digital Documents" (V3D2), SPP 1041. It was also partially funded by the Information Society Technologies program of the European Commission, Future and Emerging Technologies under the IST-2001-33058 PANDA project (2001–2004). The first author is on leave from the Department of Computer Science, University of Chile.

References

[1] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley, 1999.

[2] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, and D. Jacobs. A search engine for 3D models. *ACM Trans. on Graphics*, 22(1):83–105, 2003.

[3] D. Healy, D. Rockmore, P. Kostelec, and S. Moore. FFTs for the 2-sphere - Improvements and variations. *Journal of Fourier Analysis and Applications*, 9(4):341–385, 2003.

[4] M. Heczko, D. Keim, D. Saupe, and D. Vranić. Methods for similarity search on 3D databases. *Datenbank-Spektrum*, 2(2):54–63, 2002. In German.

[5] T. Kato, M. Suzuki, and N. Otsu. A similarity retrieval of 3D polygonal models using rotation invariant shape descriptors. In *Proc. IEEE Intl. Conf. on Systems, Man, and Cybernetics*, pages 2946–2952, 2000.

[6] D. Keim. Efficient geometry-based similarity search of 3D spatial databases. In *Proc. ACM Intl. Conf. on Management of Data (SIGMOD'99)*, pages 419–430. ACM Press, 1999.

[7] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin. The QBIC project: Querying images by content, using color, texture, and shape. In *Storage and Retrieval for Image and Video Databases*, pages 173–187, 1993.

[8] M. Novotni and R. Klein. A geometric approach to 3D object comparison. In *Proc. Intl. Conf. on Shape Modeling and Applications*, pages 167–175. IEEE CS Press, 2001.

[9] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Shape distributions. *ACM Trans. on Graphics*, 21(4):807–832, 2002.

[10] E. Paquet, M. Murching, T. Naveen, A. Tabatabai, and M. Rioux. Description of shape information for 2-D and 3-D objects. *Signal Processing: Image Communication*, 16:103–122, 2000.

[11] E. Paquet and M. Rioux. Nefertiti: A tool for 3-D shape databases management. *Image and Vision Computing*, 108:387–393, 2000.

[12] O. Ronneberger, H. Burkhardt, and E. Schultz. General-purpose object recognition in 3D volume data sets using gray-scale invariants - Classification of airborne pollen-grains recorded with a confocal laser scanning microscope. In *Proc. Intl. Conf. on Pattern Recognition (ICPR'02)*, volume 2, pages 290–295, 2002.

[13] T. Seidl and H.-P. Kriegel. Efficient user-adaptable similarity search in large multimedia databases. In *Proc. 23rd Intl. Conf. on Very Large Databases (VLDB'97)*, pages 506–515. Morgan Kaufmann, 1997.

[14] J. Tangelder and R. Veltkamp. Polyhedral model retrieval using weighted point sets. *Intl. Journal of Image and Graphics*, 3(1):209–229, 2003.

[15] D. Vranić. An improvement of rotation invariant 3D-shape descriptor based on functions on concentric spheres. In *Proc. IEEE Intl. Conf. on Image Processing (ICIP'03), Volume III*, pages 757–760, 2003.

[16] D. Vranić and D. Saupe. 3D model retrieval with spherical harmonics and moments. In *Proc. DAGM-Symp.*, LNCS 2191, pages 392–397. Springer, 2001.

[17] D. Vranić and D. Saupe. 3D shape descriptor based on 3D Fourier transform. In *Proc. EURASIP Conf. on Digital Signal Processing for Multimedia Communications and Services (ECMCS'01)*, pages 271–274, 2001.

[18] D. Vranić and D. Saupe. Description of 3D-shape using a complex function on the sphere. In *Proc. IEEE Intl. Conf. on Multimedia and Expo*, pages 177–180, 2002.

[19] D. Vranić, D. Saupe, and J. Richter. Tools for 3D-object retrieval: Karhunen-Loeve transform and spherical harmonics. In *Proc. IEEE 4th Workshop on Multimedia Signal Processing*, pages 293–298, 2001.

[20] T. Zaharia and F. Prêteux. 3D shape-based retrieval within the MPEG-7 framework. In *Proc. SPIE Conf. on Nonlinear Image Processing and Pattern Analysis XII*, volume 4304, pages 133–145, 2001.