

George Leifman · Ron Meir · Ayellet Tal

Semantic-Oriented 3D Shape Retrieval using Relevance Feedback

Abstract Shape-based retrieval of 3D models has become an important challenge in computer graphics. Object similarity, however, is a subjective matter, dependent on the human viewer, since objects have semantics and are not mere geometric entities. Relevance feedback aims at addressing the subjectivity of similarity. This paper presents a novel relevance feedback algorithm that is based on supervised as well as unsupervised feature extraction techniques. It also proposes a novel signature for 3D models, the sphere projection. A Web search engine that realizes the signature and the relevance feedback algorithm, is presented. We show that the proposed approach produces good results and outperforms previous techniques.

Keywords 3D retrieval · search engine · relevance feedback

1 Introduction

Given a database of 3D models and a query model, the objective of 3D retrieval is to obtain the most similar objects from the database. Usually, the problem is divided into two sub-problems. First, each model should be compactly represented by a *signature*. Second, a *distance measure* on these signatures should be defined.

Static signatures and distances are often not sufficient. After all, similarity is a subjective issue, dependent on the human viewer. Different users might have conflicting interpretations regarding the similarity of models. For instance, what is more similar to a centaur – a man or a horse?

Relevance feedback (RF) lets the user incorporate his or her perceptual feedback in the search, by iterating the following three stages. First, the system retrieves similar models and presents them to the user in descending order of sim-

This work was partially supported by European FP6 NoE grant 506766 (AIM@SHAPE), by the Israeli Ministry of Science, grant 01-01-01509 and by the Ollendorff foundation.

Department of Electrical Engineering
Technion – Israel Institute of Technology
E-mail: {gleifman@techunix,rmeir@ee,ayellet@ee}.technion.ac.il

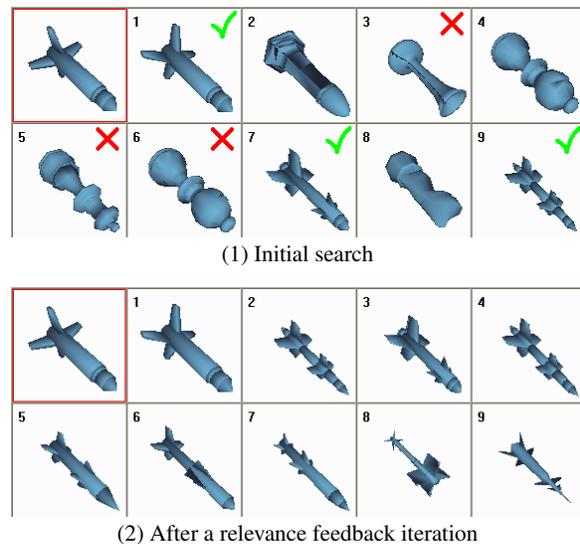


Fig. 1 Filtering out geometrically similar, but semantically dissimilar models (query model at the top-left)

ilarity. Next, the user provides feedback regarding the relevance of some of the current retrieval results. Finally, the system uses these examples to learn and improve the performance in the next iteration, as demonstrated in Figure 1.

While relevance feedback has been used in the retrieval of text and images [12,25,23,26,30], it is a relatively new area of research for 3D shapes. A learning technique based on *Support Vector Machine (SVM)* is studied in [8]. A feature space warping approach is presented in [3]. A method that ranks relevant (irrelevant) objects on top (bottom) is presented in [1]. This paper proposes a new approach that is based both on supervised and unsupervised feature extraction [23,26]. We show that our technique outperforms previous schemes.

This paper also proposes a novel signature, the *sphere projection*, which attempts to capture the global characteristics of a 3D model by computing the amount of “energy” required to deform it into a sphere. This signature is enriched by topological properties.

Various signatures for 3D shapes have been proposed [31, 10, 29]. They include Reeb graphs and other graph representations [11, 4, 35], shape distributions [21], moments [8], cords, moments and wavelets [22], reflective symmetry [16], Fourier coefficients for spherical harmonics [33, 32, 17], light-field descriptors [5], multi-scale hierarchical representations [13] and more. The signature proposed in this paper captures the global geometry and topology of the objects. It tolerates degenerated meshes and disconnected components.

The paper makes the following contributions:

1. A Web search engine for 3D models was built. The system lets the user provide perceptual feedback and updates the search using a relevance feedback algorithm.
2. A new algorithm for relevance feedback is proposed. This algorithm is general and can be applied to any signature represented as a vector. It combines several known techniques from information retrieval in a new way.
3. A novel signature that takes into account both geometric and topological properties of models, is presented.
4. Criteria for measuring the quality of signatures and relevance feedback algorithms are discussed.

This paper is organized as follows. Section 2 introduces the signature. Section 3 presents the relevance feedback algorithm. Section 4 describes the experiments performed and presents some results. Section 5 concludes this paper.

2 Signature

This section describes a new signature for representing 3D models. It consists of a geometric signature, the *sphere projection*, as well as a topological signature.

Geometric signature: The *sphere projection* signature attempts to capture the global characteristics of the model by computing the amount of “energy” required to deform it into a predefined three-dimensional shape, in our case a sphere.

Let \mathbf{F} be the applied force and *dist* be the distance between the enclosing sphere and the model surface. The energy required to deform a model into a sphere is given by $E = \int_{dist} \mathbf{F} \cdot d\mathbf{r}$. We assume that the force is constant along this distance and is also constant for all the points on the model’s surface. Therefore, the energy is proportional to the average distance between the sphere and the model.

To implement it, the sphere projection signature is defined as a concatenation of three sub-signatures: the distance from the sphere to the model D_1 , the distance from the model to the sphere D_2 and the variance of radii D_3 . While D_1 and D_2 describe global properties of the model, D_3 captures the local geometric structure.

D_1 is a bi-variate function which represents the minimal distance from a point on the enclosing sphere to the model’s surface. Let R be the radius of the enclosing sphere, $P_{(\theta, \phi, R)}$ be a point on the enclosing sphere, where (θ, ϕ) are spherical coordinates, and O be the set of points on the model’s surface. Then:

$$D_1(\theta, \phi) = \min_{o \in O} (||P_{(\theta, \phi, R)} - o||). \quad (1)$$

D_1 is not sufficient for describing non star-shaped models. For instance, D_1 for a sphere with a cylindrical hole from one pole to another is equal to D_1 for a sphere with dents on the poles.

To solve this problem, D_2 , a bi-variate function which represents the distance to a sphere, is considered. We denote the set of model points having the same spherical coordinates (θ, ϕ) by $G(\theta, \phi)$ and the size of $G(\theta, \phi)$ by $|G(\theta, \phi)|$.

$$D_2(\theta, \phi) = \frac{\sum_{(\theta, \phi, r) \in G(\theta, \phi)} (R - r)}{|G(\theta, \phi)|}, \quad (2)$$

where r is the radius of a point in $G(\theta, \phi)$. If the size of $G(\theta, \phi)$ is infinite, the sum is replaced by an integral.

In the implementation, in order to calculate the distances, the sphere’s surface is first sampled, producing a 2D mesh, M , of $m \times n$ points. The ij^{th} sample point, $1 \leq i \leq m$, $1 \leq j \leq n$, is defined as:

$$M_{ij} = \left(\frac{2\pi(i - 0.5)}{m}, -0.5\pi + \frac{\pi(j - 0.5)}{n}, R \right). \quad (3)$$

Next, a set of points, O , distributed uniformly over the model’s surface, is drawn. The entries of the distance matrix D_1 are defined by:

$$D_1^{ij} = \min_{o \in O} (||M_{ij} - o||). \quad (4)$$

Similarly, for each sampled point $o = (\theta, \phi, r) \in O$, a sample point on the sphere having the most similar angles θ and ϕ is found. Thus, for each sphere sample point, a corresponding set of model points, G_{ij} , is produced. The entries of the distance matrix D_2 are defined by:

$$D_2^{ij} = \frac{\sum_{(\theta, \phi, r) \in G_{ij}} (R - r)}{|G_{ij}|}. \quad (5)$$

Finally, D_3 , the variance of the radii, is calculated. It represents the deviation from the sphere within a specific angle. Let m_{ij} be the mean of the radii r of the points in the set G_{ij} . The entries of the matrix D_3 are defined by:

$$D_3^{ij} = \frac{\sum_{(\theta, \phi, r) \in G_{ij}} (r - m_{ij})^2}{|G_{ij}|}. \quad (6)$$

Each entry D^{ij} in the signature matrix D is a concatenation of D_1^{ij} , D_2^{ij} and D_3^{ij} into a vector. Signatures are compared using the L_2 metric.

Topological signature: The geometric signature is enriched with topological properties – the model’s Betti numbers [6, 18]. Betti zero, β_0 , is the number of connected components; Betti one, β_1 , is the number of independent tunnels; Betti two, β_2 , is the number of closed regions in space.

Comparing Betti numbers using the L_1 or L_2 metric, does not yield the desirable results. This is due to the substantial difference between the comparison of small and large Betti numbers. For instance, the difference between 2 and 3 connected components is more significant than the difference between 3298 and 3299 components. Therefore, a logarithmic function is applied to the Betti numbers prior to signature comparison.

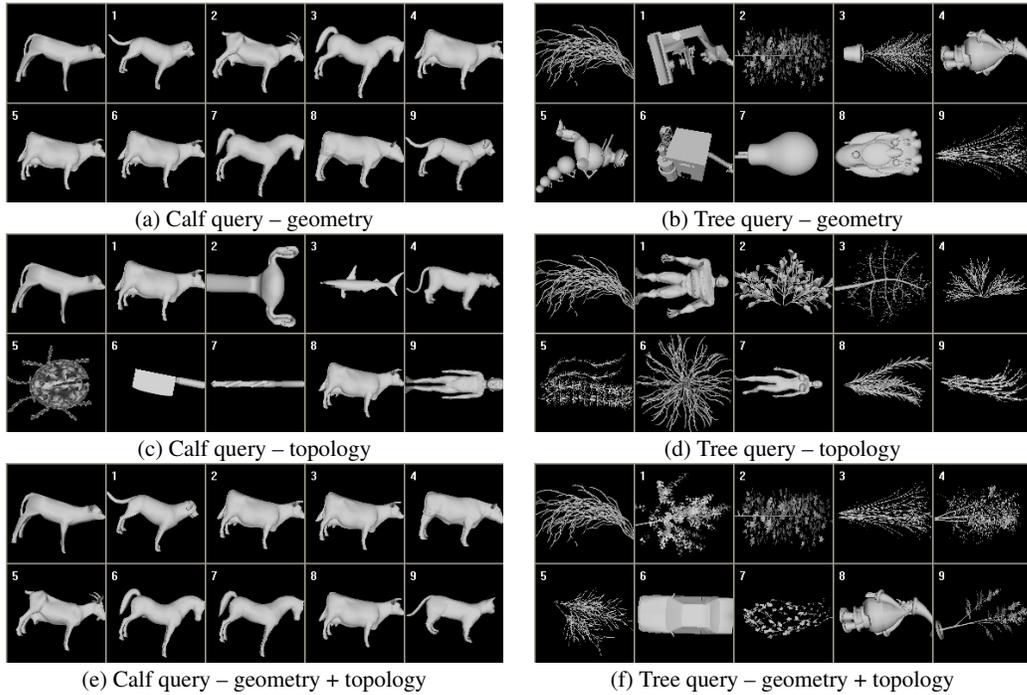


Fig. 2 Queries using geometric signatures, topological signatures and combinations

Combining geometry and topology: The topological signature is concatenated to the geometrical signature, to form the combined signature. The relative weights of the signatures are adjusted automatically, using relevance feedback, described in the next section.

Figure 2 compares the results of two queries, using only geometric signatures, only topological signatures, and combinations of the two. For a calf, the geometric signature achieves good results, but the topological signature does not. This is because all 4-legged animals are similar geometrically, but many other models resemble the calf topologically. Trees, however, are not necessarily similar geometrically. Yet, they are usually modeled similarly by designing one element (e.g., a leaf, a branch) and copying it multiple times. As a result, all the trees have many components. Therefore, in this case, the topological signature achieves good results. In both examples, the best results are achieved when the combined signatures are used.

Often, Betti numbers also help in retrieving models in different poses, such as people in different motions. This is so because the same prototype is used to model the objects, which therefore have the same topology.

3 Relevance feedback algorithm

Though relevance feedback has been a lively topic of research in text retrieval and in image retrieval, it was hardly explored in 3D shape retrieval. We are aware of only a few algorithms that specifically target 3D models [8, 3, 1].

In this section we describe our algorithm for 3D relevance feedback. It builds upon state-of-the-art algorithms in information retrieval and puts them together in a new way.

The algorithm consists of a pre-processing off-line stage and an online computation. In pre-processing, which is applied to the whole database, relevance information is not used. Reversely, relevance information is used in each online step, to improve the retrieval results.

Pre-processing: During pre-processing, *unsupervised feature extraction* is applied. Given N observations on d variables, feature extraction refers to the reduction of the dimensionality of the data by finding r new variables, $r \leq d$, and projecting the data [9]. This projection obtains an efficient combination of the features in the sense of estimation variance.

The most widespread linear mapping is the *Principal Component Analysis (PCA)* [15]. PCA finds a projection matrix W : $\mathbf{y} = W^T \mathbf{x}$, where $\mathbf{y} \in \mathcal{R}^r$ is a transformed data point, W is a $d \times r$ transformation matrix and $\mathbf{x} \in \mathcal{R}^d$ is an original data point.

Since standard PCA cannot capture nonlinear structures of the input data, we use a more advanced technique, the *Kernel Principal Component Analysis (KPCA)* [28]. KPCA is based on the computation of the standard linear PCA in a new feature space, into which the input data is mapped using a nonlinear transformation.

To avoid computationally expensive calculations of high-dimensional dot products, *kernels* are used [27]. A kernel is a function K , such that for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$, $K(\mathbf{x}, \mathbf{y}) = \langle \Phi(\mathbf{x}), \Phi(\mathbf{y}) \rangle$, where Φ is a mapping from \mathcal{X} to a high-dimens-

ional feature space. In our system, good results are obtained using the Gaussian kernel with $\sigma = 1$:

$$K(\mathbf{x}, \mathbf{y}) = e^{-|\mathbf{x}-\mathbf{y}|^2/2\sigma^2}.$$

Experimentally, decreasing the dimensionality of the signature from 219 to 100, not only decreases the running time, but also improves the retrieval results by 5%, both in the initial search and in the following relevance feedback iterations. This improvement stems from the fact that KPCA finds correlations between the original features and increases the weight of the more important features.

Relevance feedback step: The aim is to separate between the relevant and the irrelevant results. Thus, the algorithm should search for the best transformation that preserves class separability in a low dimension. This can be done using *supervised feature extraction*. We show that no single method is fully appropriate and offer a combination that outperforms the known methods.

Given N observations on d variables, divided into two subsets \mathcal{D}_1 and \mathcal{D}_2 with N_1 and N_2 samples in each subset, respectively. We aim at finding a projection onto some r -dimensional subspace, $r \leq d$, by $\mathbf{y} = W^T \mathbf{x}$, where $\{y_i\}_{i=1}^N$ are divided into the subsets \mathcal{Y}_1 and \mathcal{Y}_2 , so as to achieve the maximal separation between \mathcal{Y}_1 and \mathcal{Y}_2 . $\mathbf{y} \in \mathcal{R}^r$ is a transformed data point and W is a $d \times r$ transformation matrix.

There are various ways to address this problem. A common solution is *Linear Discriminant Analysis (LDA)* [9, 7], which is formulated as an optimization problem. Let \mathbf{m} be the mean vector of all observations and \mathbf{m}_i , $i = 1, 2$, be the sample means (relevant and irrelevant). Define two scatter matrices: the between-class scatter matrix S_B and the within-class scatter matrix S_W :

$$S_B = \sum_{i=1}^2 N_i (\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T, \quad (7)$$

$$S_W = \sum_{i=1}^2 \sum_{\mathbf{x} \in \mathcal{D}_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T, \quad (8)$$

The optimal transformation matrix W is defined as

$$W_{opt} = \operatorname{argmax}_W \left\{ \frac{W^T S_B W}{W^T S_W W} \right\}. \quad (9)$$

LDA finds an optimal linear transformation that re-weights the signature entries so that the maximal separation between the relevant and the irrelevant results, is achieved. However, LDA also aims at clustering the relevant examples and the irrelevant examples in the discriminating subspace. The set of relevant examples is likely to represent the true distribution, since the class of interest has a compact support. However, the irrelevant examples are often too sparse to represent their true distribution. Moreover, they can be heterogeneous and reside far from each other in feature space. Thus, any attempt to cluster them is not only unnecessary, but also potentially damaging.

It is thus preferred to treat the relevant and the irrelevant examples differently. *Biased Discriminant Analysis (BDA)*

addresses this asymmetry [34]. In BDA, the scatter matrices S_B and S_W , are replaced by S_z and S_x :

$$S_z = \sum_{i=1}^{N_z} (\mathbf{z}_i - \mathbf{m}_x)(\mathbf{z}_i - \mathbf{m}_x)^T, \quad (10)$$

$$S_x = \sum_{i=1}^{N_x} (\mathbf{x}_i - \mathbf{m}_x)(\mathbf{x}_i - \mathbf{m}_x)^T, \quad (11)$$

where $\{\mathbf{x}_i\}_{i=1}^{N_x}$ are the relevant examples, $\{\mathbf{z}_i\}_{i=1}^{N_z}$ are the irrelevant examples and \mathbf{m}_x is the mean vector of the relevant examples. The mean vector \mathbf{m}_x is subtracted from the observations, in order to cluster the relevant examples together, while keeping them away from the irrelevant examples.

Our experiments show that BDA fails for a small number of training examples. Fortunately, in this case, LDA achieves good results. On the other hand, LDA fails for a large number of training examples, since in this case it is difficult to cluster the irrelevant example.

In relevance feedback retrieval systems, the number of training examples provided by the user, cannot be controlled. The system should yield the best performance for any sample size. A valid solution is to use LDA for a small set of training examples and BDA otherwise. But how should the system automatically determine which is the case?

In our algorithm, Fisher's Linear Discriminant (FLD) criterion [7], a 1-dimensional LDA, is used to determine which method to utilize. The higher its value, the more likely it is that LDA successfully discriminates between the relevant and the irrelevant classes. Given a query, the FLD is first calculated. According to its value, it is determined when to switch from LDA to BDA. Figure 3 shows the average performance of LDA, BDA and our algorithm, evaluated using *Discounted Cumulated Gain (DCG)*, described in Section 4.

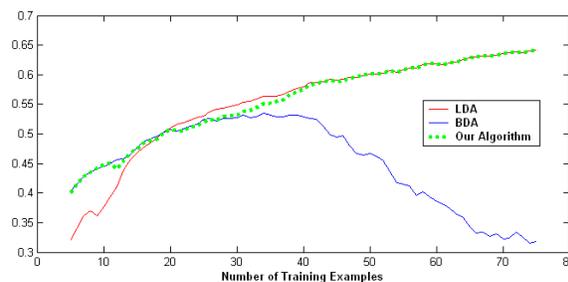


Fig. 3 Using FLD to switch between LDA and BDA

4 System and experiments

We have developed a 3D Web search engine, *Geogle*, that lets the user provide relevance feedback, in order to refine the search results [20]. The input is a 3D model, which can be supplied by the user or be found in the database using a text search. Refining the search results is done by marking

some of the results as relevant or irrelevant. Then, the algorithm described in Section 3 is applied and the new results are displayed to the user. The process can be iterated until the system retrieves the models the user “has in mind”.

Section 4.1 describes the database organization. Section 4.2 describes several methods for evaluating the quality of retrieval results. In Section 4.3, the quality of the signature is evaluated using these measures. Section 4.4 estimates the performance of the relevance feedback algorithm.

4.1 Database overview

Our experiments were performed on a database containing 1850 3D models, which were collected from the Internet. In order to evaluate the methods described in this paper, out of these models, 725 were semantically classified into 25 classes and 1125 were free models (Table 1). Obviously, the models need not be classified in order for the database to be used. Classification is performed solely to enable the evaluation described hereafter.

Table 1 Database organization

Class	Size	Class	Size
4-legged animals	32	Airplanes	95
Bottles	15	Cars	57
Chairs	50	Chess	12
Couches	18	Doors	11
Faces	6	Glasses	5
Guitars (violins)	17	Helicopters	12
Knives (swords)	38	Missiles	24
People	78	Plants	47
Race cars	20	Rifles	28
Space Ships	55	Submarines	10
Tanks	12	Teapots	11
Trees	53	Vases	9
Zeppelins	10		
		Non-Classified	1125
		TOTAL	1850

In a pre-processing step, all the models in the database were normalized to achieve invariance to translation, scale and rotation [8]. Then, a signature was generated for every model and stored in the database, as described in Section 2. Each signature has 219 entries, choosing $m = 12$, $n = 6$ (Equation 3) and three Betti numbers.

4.2 Evaluation methods

Various evaluation methods have been proposed in information retrieval. In this section we discuss several methods that are used in the evaluation of our proposed signature.

1. *Nearest neighbor* [11,21]: Check whether the most similar retrieved model belongs to the same class as the query model.

2. *Precision/recall measurements* [2,19]: Let C be the set of models that belong to the same class as the query, S be

the set of all retrieved models and $I = C \cap S$. *Recall* and *Precision* are defined as $R = \frac{|I|}{|C|}$ and $P = \frac{|I|}{|S|}$, respectively.

A common measure that addresses the difficulty in evaluating the effectiveness by a pair of numbers that may covary in a loosely specified way is the *F-Measure* [24]. This measure is high if both recall and precision are high:

$$F = \frac{2PR}{P + R} = \frac{2}{1/P + 1/R}. \quad (12)$$

3. *First/second tier* [11,34]: The first/second tier measure the success percentage among the first k retrieved models. In the first tier, $k = (\text{size of the model's class})$, while in the second tier $k = 2 \times (\text{size of the model's class})$.

4. *Cumulated gain based measurements* [14]: Let G be the gain vector, whose i^{th} entry G_i is 1 if the i^{th} retrieved model is in the same class as the query and 0 otherwise. The *cumulated gain vector* CG is defined recursively by:

$$CG_i = \begin{cases} G_1 & i = 1 \\ CG_{i-1} + G_i & \text{otherwise.} \end{cases} \quad (13)$$

The *cumulated gain vector with a discount factor*, DCG , is defined recursively by:

$$DCG_i = \begin{cases} G_1 & i = 1 \\ DCG_{i-1} + G_i / \log_2 i & \text{otherwise.} \end{cases} \quad (14)$$

Here, less similar models are considered less relevant, to accommodate for users who might be less likely to examine results down the list.

Measuring the algorithm’s performance with a single value, is done by normalizing by the best possible result:

$$\overline{DCG} = \frac{DCG_k}{1 + \sum_{j=2}^{|C|} \frac{1}{\log_2(j)}}, \quad (15)$$

where k is the number of retrieved models and $|C|$ is the size of the class the query belongs to.

4.3 Signature results

To evaluate our signature, it is compared to: (1) shape moments [8], (2) shape distribution [21] and (3) lightfield descriptors [5]. Figure 4 presents some examples and demonstrates the differences between the results. For instance, ten 4-legged animals were retrieved among the top ten using the sphere projection signature, while six, four and eight were retrieved using shape distributions, moments and lightfields, respectively.

To thoroughly evaluate the performance, each classified model in the database is used as a query model and the results are averaged over all the queries. A retrieved model is considered relevant if it belongs to the same class of the query model.

Figure 5 displays the results according to the measures described in Section 4.2, averaged over all queries. In Figure 5(a)-(d), the first 9×4 bars show the average performance for 9 representative large classes and the last 4 bars show the total average over all 25 classes.

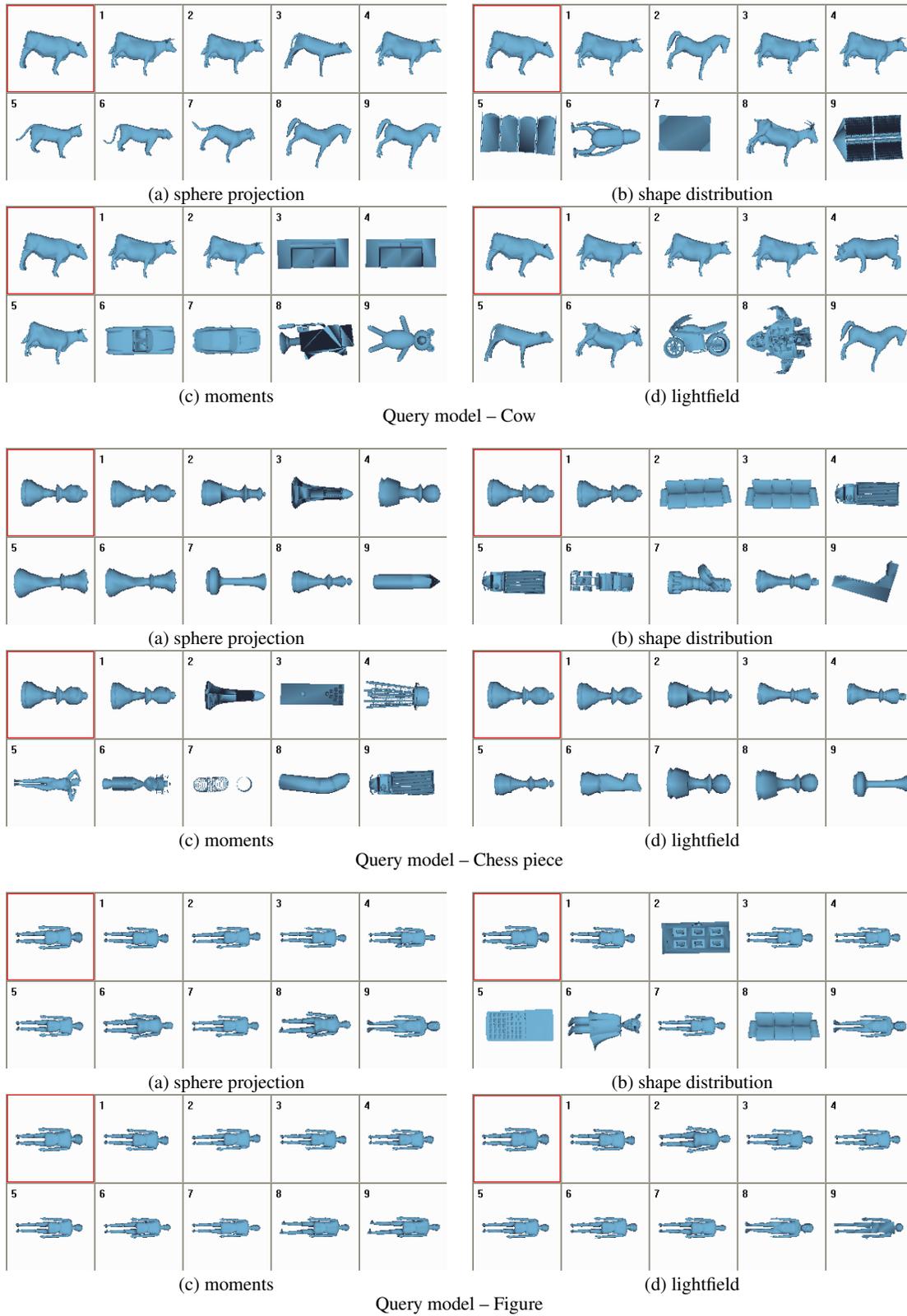


Fig. 4 Retrieval results – the query model is at the left upper corner

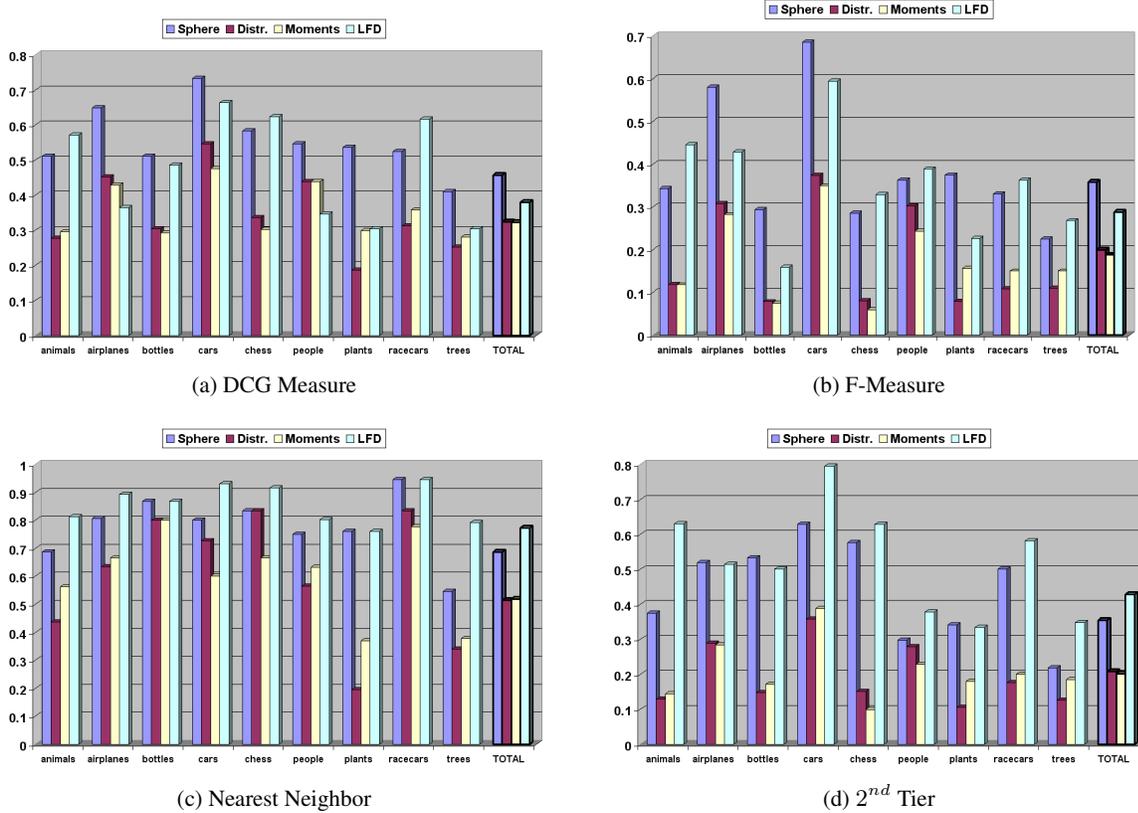


Fig. 5 Average performance using various evaluation criteria: The different colors indicate four different signatures: blue - sphere, red - distributions, yellow - moments; cyan - lightfield

These results indicate that the sphere projection signature performs better than shape distributions and moments and compares well with lightfields, which is considered a very good signature [29].

An interesting observation is that the lightfield descriptor performs slightly better than the sphere projection for the k -nearest neighbors-related measures (i.e., nearest neighbor and second tier) while the sphere projection signature performs slightly better than the lightfield descriptor for the precision/recall measure (F-measure) and the cumulated gain measure. The latter two measures (precision/recall and DGC) are often considered more indicative measures.

Though similar in quality, the sphere projection has several benefits over the lightfield descriptor in terms of storage and computational costs. Table 2 summarizes our findings for the average signature size, signature generation time and query time, when executed on a Pentium 4 1.6GHz, 256MB RAM machine. The query time we achieved for the lightfields is taken from the original paper, since our results were higher. A model in the database has 13, 000 faces on average.

Figure 6 illustrates the average performance of the topological, the geometric and the combined signatures according to the DCG measurement. Seven classes are presented, where the first five have a similar geometric structure and the last two (plants and trees) do not. Using a combined signature achieves a high performance for all the classes. Averag-

Table 2 Time and space complexity

Signature	Size (Kb)	Generation Time (sec)	Query Time (sec)
Spheres	2.3	2.1	0.1
Lightfields	4.7	6.1	0.4
Distributions	2.0	1.9	0.1
Moments	0.4	0.9	0.04

ing over all the queries (right column) shows that the combined signature is more effective for a wide range of queries.

4.4 Relevance feedback results

Figures 1, 7–8 demonstrate different uses of relevance feedback (RF). Figure 1 shows how it is used to filter out geometrically-similar, but semantically-dissimilar, models, i.e., only missiles are retrieved in the nine top-ranked results. Figure 7 is an example of using RF to narrow down the retrieval results. Using an open-roof car as a query model, both regular and race cars are retrieved. By marking the race cars as irrelevant and some regular cars as relevant, the next iteration retrieves only regular cars. Figure 8 is an example of using RF when the query model has only a remote similarity to the

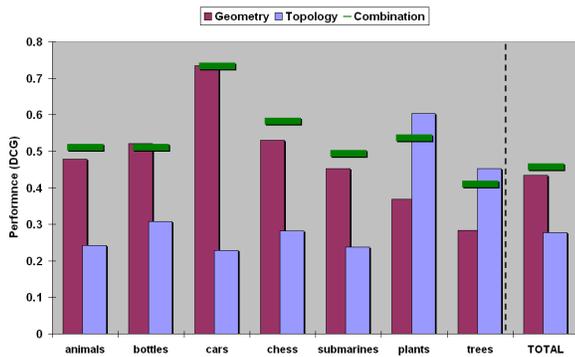


Fig. 6 Average performance using topological, geometric and combined signatures

models searched for. Using a helicopter as a query, airplanes are retrieved after a single RF iteration.

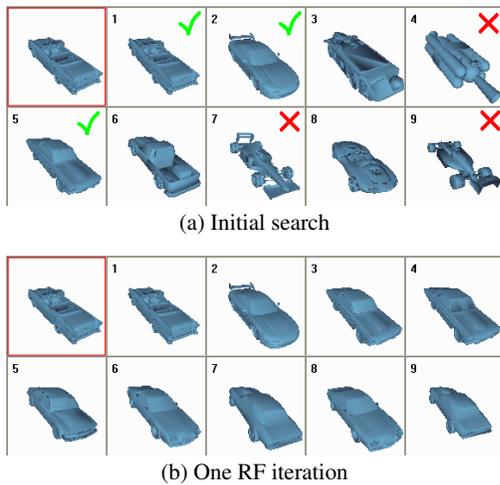


Fig. 7 Retrieving cars – narrowing down the results

To comprehensively evaluate the average performance of our relevance feedback algorithm, the experiments were performed as follows. Each classified model was used as a query. For each query, after the initial search, the top results that belong to the query’s class were automatically marked as relevant and the rest were marked as irrelevant. The performance of a query was evaluated using DCG. DCG was chosen as a measure, not only because it takes into account the position of the relevant results, but also because it has the lowest standard deviation among all the standard measures, making it the most stable measure. The performance was averaged over all the classified models.

Figure 9 shows the average performance as a function of the number of training examples. The most drastic improvement is achieved after the first and the second RF iterations, while the third and the fourth iterations improve the results only slightly. This is an important point because users are unlikely to perform many iterations. Moreover, increasing

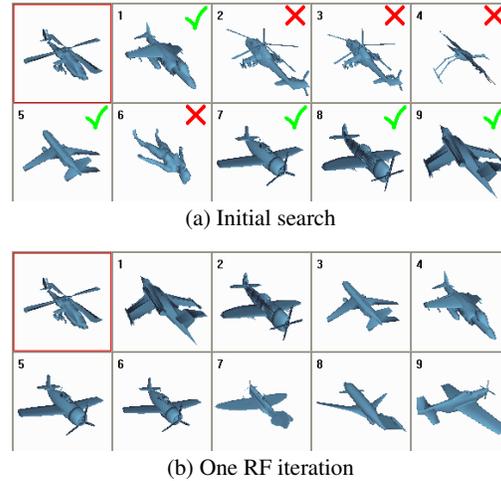


Fig. 8 Retrieving airplanes using a helicopter as a query – the query has a remote similarity to the desired models

the number of training examples improves the performance, as expected. Finally, relative to the initial search, the overall performance is almost doubled.

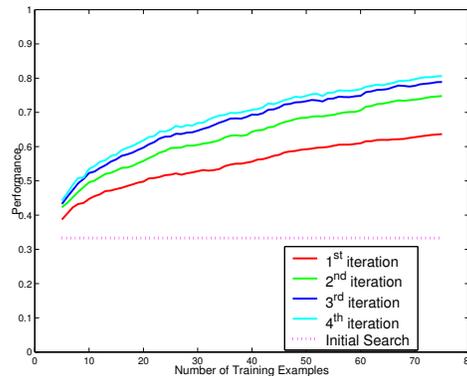


Fig. 9 Performance (DCG) vs. # of training examples

Figure 10 compares the performance of our relevance feedback algorithm to the SVM-based approach [8] and to the feature space warping approach [3]. It can be seen that our algorithm outperforms both algorithms.

5 Conclusions

Signatures for 3D models have improved in the last several years and will undoubtedly keep improving. This, however, will not suffice to retrieve from a database what the user “has in mind”. Using the same query model, different users are likely to expect different retrieval results. Relevance feedback provides the user with the added ability of influencing the search as it is being conducted. In particular, relevance feedback provides a convenient interactive way to retrieve semantically-similar objects.

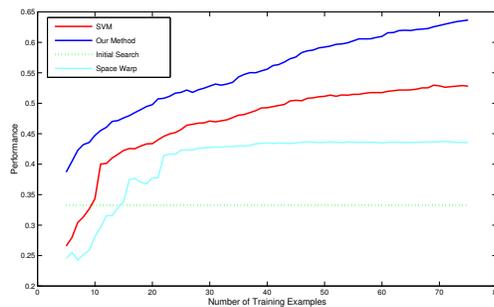


Fig. 10 The performance of our algorithm, [ETA01] and [BC02]

This paper has proposed a novel relevance feedback scheme. The algorithm builds upon some of the best known techniques in information retrieval and combines them in a new, completely automatic, manner, so as to outperform the existing techniques. Most of the improvement is gained in the first couple of iterations, which is an important aspect in interactive techniques.

The paper has also proposed a novel signature for 3D models that attempts to capture the global characteristics of the geometry and the topology of the model. It has been shown that this combination provides a good signature using various criteria prevalent in information retrieval.

Several aspects of this study can be extended. First, additional features, such as colors and textures, can be considered. The relevance feedback scheme provides a convenient way to weigh the various features automatically. Second, the proposed relevance feedback algorithm can be extended, to allow the user to provide the degree of relevance, instead of just marking the result as relevant or not. Finally, an intriguing future direction is partial matching.

References

- Atmosukarto, I., Leow, W., Huang, Z.: Feature combination and relevance feedback for 3D model retrieval. In: MMM, pp. 334–339 (2005)
- Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. ACM Press (1996)
- Bang, H., Chen, T.: Feature space warping: An approach to relevance feedback. In: ICIP (2002)
- Bespalov, D., Shokoufandeh, A., Regli, W., Sun, W.: Scale-space representation of 3D models and topological matching. In: ACM Symposium on Solid Modeling and Applications, pp. 208–215 (2003)
- Chen, D.Y., Tian, X.P., Shen, Y.T., Ouhyoung, M.: On visual similarity based 3D model retrieval. *Eurographics* **22**(3), 223–232 (2003)
- Delfinado, C., Edelsbrunner, H.: An incremental algorithm for Betti numbers of simplicial complexes on the 3-sphere. *Computer Aided Geometric Design* **12**, 771–784 (1995)
- Duda, R., Hart, P., Stork, D.: Pattern Classification. John Wiley & Sons, New York (2000)
- Elad, M., Tal, A., Ar, S.: Content based retrieval of vrmf objects - an iterative and interactive approach. *EG Multimedia* **39**, 97–108 (2001)
- Fukunaga, K.: Introduction to Statistical Pattern Recognition. Academic Press, 2nd edition (1990)
- Funkhouser, T., Min, P., Kazhdan, M., Chen, J., Halderman, A., Dobkin, D., Jacobs, D.: A search engine for 3D models. *ACM Transactions on Graphics* **22**(1) (2003)
- Hilaga, M., Shinagawa, Y., Kohmura, T., Kunii, T.: Topology matching for fully automatic similarity estimation of 3D shapes. *SIGGRAPH* pp. 203–212 (2001)
- Ishikawa, Y., Subramanya, R., Faloutsos, C.: MindReader: Querying databases through multiple examples. In: Proceedings of 24th Int. Conference Very Large Data Bases, pp. 218–227 (1998). URL citeseer.nj.nec.com/article/ishikawa98mindreader.html
- Iyer, N., Jayanti, S., Lou, K., Kalyanaraman, Y., Ramani, K.: A multi-scale hierarchical 3D shape representation for similar shape retrieval. In: TMCE (2004)
- Jarvelin, K., Kekalainen, J.: IR evaluation methods for retrieving highly relevant documents. In: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (2000)
- Jolliffe, I.: Principal Component Analysis. Springer-Verlag (1986)
- Kazhdan, M., Chazelle, B., Dobkin, D., Funkhouser, T.: A reflective symmetry descriptor for 3D models. *Algorithmica* **38**(2) (2003)
- Kazhdan, M., Funkhouser, T., Rusinkiewicz, S.: Rotation invariant spherical harmonic representation of 3D shape descriptors. In: Symposium on Geometry Processing (2003)
- Konke, S., Moran, P., Hamann, B., Joy, K.: Fast methods for computing isosurface topology with betti numbers. In: Proceedings of Dagstuhl Seminar on Scientific Visualization, pp. 14–18 (2002). Data Visualization: The State of the Art
- Korfhage, R.: Information Storage and Retrieval. John Wiley & Sons, New York (1997)
- Leifman, G.: Geogle. <http://www.technion.ac.il/~gleifman/>
- Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D.: Matching 3D models with shape distributions. In: Proceedings of the International Conference on Shape Modeling and Applications, pp. 154–166 (2001)
- Paquet, E., Murching, A., Naveen, T., Tabatabai, A., Rioux, M.: Description of shape information for 2-d and 3-d objects. *Signal Processing: Image Communication* pp. 103–122 (2000)
- Peng, J., Bhanu, B., Qing, S.: Probabilistic feature relevance learning for content-based image retrieval. *Computer Vision and Image Understanding* **75**, 150–164 (1999)
- van Rijsbergen, C.: Information retrieval, 2 edn. Butterworth, London (1979)
- Rui, Y., Huang, T., Mehrotra, S.: Relevance feedback techniques in interactive content-based image retrieval. In: Storage and Retrieval for Image and Video Databases (SPIE), pp. 25–36 (1998). URL citeseer.nj.nec.com/rui98relevance.html
- Santini, S., Jain, R.: Integrated browsing and querying for image database. *IEEE Multimedia* **7**, 26–39 (2000)
- Scholkopf, B., Smola, A.: Learning with Kernels. MIT Press, Cambridge, MA (2002)
- Scholkopf, B., Smola, A., Muller, K.: Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation* **10**, 1299–1319 (1998)
- Shilane, P., Min, P., Kazhdan, M., Funkhouser, T.: The princeton shape benchmark. In: Shape Modeling International (2004)
- Tieu, K., Viola, P.: Boosting image retrieval. In: Proceedings of IEEE Conference Computer Vision and Pattern Recognition, pp. 228–235 (2000). URL citeseer.nj.nec.com/tieu00boosting.html
- Veltkamp, R.: Shape matching: Similarity measures and algorithms. In: Shape Modelling International, pp. 188–197 (2001)
- Vranic, D.: An improvement of rotation invariant 3D-shape descriptor based on functions on concentric spheres. In: ICIP, vol. 3, pp. 757–760 (2003)
- Vranic, D., Saube, D.: Description of 3D-shape using a complex function on the sphere. In: Proceedings IEEE International Conference on Multimedia and Expo, pp. 177–180 (2002)
- Zhou, X., Hunag, T.: Small sample learning during multimedia retrieval using biasmap. In: Proceedings of IEEE Computer Vision and Pattern Recognition Conference. Hawaii (2001)
- Zuckerberger, E., Tal, A., Shlafman, S.: Polyhedral surface decomposition with applications. *Computers & Graphics* **26**(5), 733–743 (2002)