Visual Rank Analysis for Search Engine Benchmarking and Efficient Navigation

Tiziana Catarci*
University of Roma "La Sapienza"

Daniel Keim†
University of Konstanz

Giuseppe Santucci‡
University of Roma "La Sapienza"

Tobias Schreck§
Technische Universität Darmstadt

Gloria Iervella¶
University of Roma "La Sapienza"

Stefano Iannarelli‖
University of Roma "La Sapienza"

Fabio Veltri**
University of Roma "La Sapienza'

Abstract

In many important applications, the search for non-standard data types is essential. E.g., digital libraries and multimedia database systems offer content-based search functionality for images and 3D documents. Contrary to the annotation-based approach, where information manually attached to the data objects if used for retrieval, in content-based retrieval, automatically derived meta-data is used. However, the quality of the meta data is crucial, and often, it a priori is not clear which meta data is best suited to execute a user-issued query. Owing to the multi-meta data problem, two crucial questions arise: (a) how can different meta data (feature vector) schemas be benchmarked to assess their suitability for solving the retrieval problem effectively, and (b) how to support the user with issuing queries to the retrieval system, considering different choices for the type of meta data to engage in the search.

In this paper, we address these questions in a two-fold contribution. Based on the DARE visualization system, we first introduce an approach for the *visual benchmarking* of multiple meta data formats on a ground truth benchmark, supporting the optimization stage of the multimedia database design. We secondly propose a simple, yet effective *visual interface* to multiple, long lists (rankings) of answer objects for the user. The latter, based on relevance feedback information supplied by the user, allows the effective identification of the meta data schema best suited for executing the similarity queries at hand.

Index Terms: H.3.3 [Information Systems]: Information Search and Retrieval—Selection Process; I.3.3 [Computing Methodologies]: Computer Graphics—Picture/Image Generation

1 Introduction

Digital libraries and multimedia database systems offer content-based search functionality for non-standard data types such as images, video, 3D documents, audio recordings, among others. Contrary to the annotation-based approach, where information manually attached to the data objects if used for retrieval, in content-based retrieval, automatically derived meta data is used. This meta data can arise in the form of so-called feature vectors (FVs), or other forms such as graphs or symbolic representations. However, due to the inherent fuzziness of the concept of *similarity*, for most data collections it is a priori not clear which features of the objects are best suited to conduct a user-issued query. Moreover, the user often is not a meta-data expert who is capable or willing to manually specify the type and configuration of meta data to use for retrieval.

The availability of multiple meta data types for executing a similarity search query can be accommodated in two ways. First, in an *off* line approach, it is possible to determine which of the potential meta data types offers best expected retrieval quality. To this end, benchmarks are used to evaluate the discrimination power based on supervised benchmark information. While from information retrieval, different statistical methods are known for rating the quality of a meta data standard under a benchmark, it is desirable to complement this statistical analysis by visual representations of the discrimination power. To this end, in this paper we develop a methodology for visual benchmarking using the DARE visualization system.

The *online* approach lets the user decide at query time which of the meta data types or which combination thereof should be used to execute the query. Relevance feedback is a technique for capturing user feedback on the relevance and/or irrelevance of seen answer, and adjusting the meta data selection in an appropriate way to reflect the relevance feedback. However, most of these approaches work in form of a black box, meaning that the user does not have an understanding of the outcome of her or his relevance selections. To address this shortcoming, we develop a simple, yet powerful visual interface for analyzing the positions of marked relevant answers, in several meta data spaces simultaneously. This interface visualizes the ranks of selected objects in the ranking that would result by the user choosing each of the available meta data formats. It is proposed as a powerful means of interaction, letting the user navigate different feature spaces in an effective and intuitive way, and support her or him in identifying promising retrieval paths to follow.

The remainder of this paper is structured as follows. Section 2 presents background and related work on multimedia retrieval, visual benchmarking, and visual user interfaces. Section 3 describes the DARE system based on which a visualization will be designed for the visual benchmarking of multi feature vector described data sets in Section 4. Also in Section 4, we will introduce a simple but powerful user interface for the visualization of relevance feedback information in context of sets of feature vectors. In Section 5, we will apply both approaches and demonstrate their usefulness for visual benchmarking and as a visual user interface. Section 6 concludes and outlines future work in the area.

*e-mail: catarci@dis.uniroma1.it
†e-mail: keim@inf.uni-konstanz.de
‡e-mail: giuseppe.santucci@dis.uniroma1.it
§e-mail: tobias.schreck@gris.informatik.tu-darmstadt.de
¶e-mail: gloria_ig@libero.it
‖e-mail: stefanoiannarelli@yahoo.it
**e-mail: fabio_veltri@yahoo.it

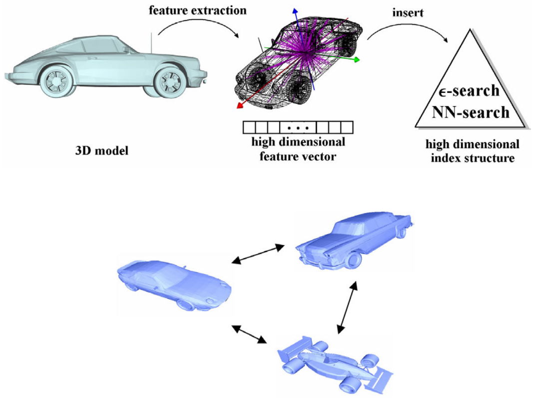


Figure 1: Top: Under the Feature Vector approach, multimedia objects are described by vectors in high-dimensional feature space, generated by certain media analysis algorithms. The vector representations of the objects in conjunction with a vector space metric then allow the calculation of distances which in turn are associated with the degree of (dis)similarity between the objects. Usually, it is difficult to define features that work well most of the time, due to inherent fuzziness of the concept similarity. The bottom image shows several 3D models, which may be considered similar or non similar to each other, based on the understanding of the concept.

2 Background and Related Work

2.1 Multimedia Similarity Search

Many modern applications rely on calculating similarity scores between non-standard data types such as multimedia objects, or multidimensional data from business, scientific, or engineering applications. E.g., content-based retrieval uses similarity scores to produce lists of answer objects given a query, and data mining algorithms such as clustering and classification rely on similarity scores to find clusters of similar objects, or to assign class labels. Extracting so-called *feature vectors* is a standard approach to map complex objects into vector space, where a suitable metric can be employed to calculate distances between vectors that are associated with distances in object space [11, 13, 15]. However, for a given type of data, there does not exists a single scheme for extracting features, but usually, many different schemes are possible, and whether a given feature representation is optimal depends on the application and data set at hand. E.g., for the 3D model data type, to date many different feature vector extractors have been proposed, still the search for efficient and effective 3D feature vector extractors continues [4, 5, 18]. Figure 1 shows a model of 3D feature vector extraction, and illustrates the inherent fuzziness of 3D similarity by an example.

Benchmarking of competing feature vector extractors is usually done by statistical benchmarking methods, e.g. known from Information Retrieval [3]. Visual benchmarking [25] applies visualization to communicate and help explore the statistical benchmarking results, and to interactively compare the discrimination power of competing feature spaces for different data sets. For many important data types, benchmark data sets (benchmarks) have been proposed to date. E.g., in 3D model retrieval, the Princeton Shape Benchmark [26], or the Purdue Engineering Shape Benchmark [20], are popular benchmarks consisting of object data and carefully compiled similarity classification information (ground truth).

An alternative to benchmark-based pre-selection of a single feature vector for use in a multimedia application is to allow *multiple* feature vectors, dynamically composing ensembles of feature vectors. Recent research addressed the automatic combination of different feature representations in retrieval and data mining applications [22, 2]. Also, methods have been proposed that interactively capture relevance feedback information from the user. This information in turn is leveraged to select and combine different feature vectors by solving an optimization problem, based on the supplied input. E.g., in [19], a multi feature image retrieval system is enhanced by relevance feedback. Specifically, the user is allowed to continuously rearrange retrieved answers in 2D space, effectively supplying relevance judgments.

Figure 2 finally illustrates a query-by-example for a 3D model. The query is executed on a database of 3D models, using 3 different feature vector representations of the database. Each feature vector space yields a different ranking, with relevant and irrelevant answer objects at different positions in the ranking. Relevant objects have been identified and marked by the user.

2.2 Visual User Interfaces

In order to effectively access a large information system it is mandatory to design a friendly interface that allows the end user to easily access the data of interest. Most of the available proposals exploit the power of visualization and direct manipulation mechanisms [8] and a widely diffused solution is to represent the data on a 3D scatter plot allowing the user to restrict the final result by changing the value of the attributes through suitable widgets. However, the choice of the visualization is not trivial [9], [17], [27], [28], [6] and it is not sufficient to associate "any" visual representation to a database but the visual representation should be carefully chosen to effectively convey all and only the database information content. To reach these goals most of the available proposals have been manually tailored for specific applications and it is very difficult to generalize them.

There is a large amount of literature on this topic, starting from Mackinlay's pioneering work on automatic design of graphical presentations [23], to a variety of other projects including the ZOO project of the University of Wisconsin [14]), AI-based proposals [1, 24], the EU funded FADIVA project [12], and many others. Basically, all these proposals share two limitations:

1. they try to automatically build complete representations, while correctness, even if it is considered a very relevant property, cannot be formally checked;

2. they concentrate on the visualization of either the schema or the instances of the database (not on both). Moreover, some proposals restrict to specific domain and/or applications instead of providing a general solution.

Our proposal try to overcome the above drawbacks by 1) defining a general theory for establishing the adequacy of a visual representation, once specified the database characteristics, and 2) developing a system, called *DARE: Drawing Adequate REpresentations* that implements such a theory and is able to automatically associate with any database the most effective visual representation. Such a visual representation has to be not only adequate (as mentioned above), but it has also to convey some database features specified by the designer (e.g., that some concepts are the most relevant).

3 The DARE System

The DARE system is based on a general theory [7] , and relies on a knowledge base containing different kinds of rules, namely:

1. *Visual rules*. Visual rules characterize the different kinds of visual symbols (e.g., they list the *visual attributes*, which are associated with the different kinds of visual symbols).

2. *Data rules*. Data rules specify the characteristics of the data model, the database schema, and the database instances (e.g.,

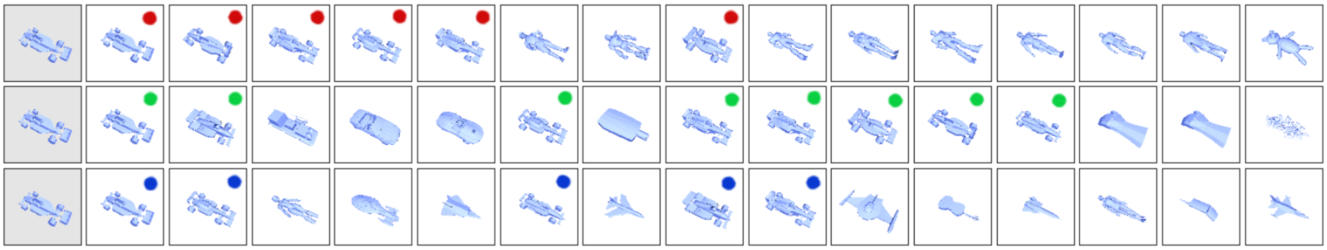


Figure 2: Query-by-example for a Formula-1 racing car model in a 3D repository. Each row shows the nearest neighbors to the query, according to a specific low-level feature space selected by the user. Relevant answer objects are marked. The different feature space representations yield different result sets, and additional relevant answer objects can be expected to appear *beyond* the limited number of thumbnails displayed. In this paper, we develop methods for the visual evaluation of different retrieval algorithms on possibly *very large* result sets.

if the designer is using the Entity-Relationship model, s/he will use a data rule to say that, for instance, `Person` is an Entity, as well as that `John` is a `Person`).

3. *Mapping rules*. Mapping rules specify the link between data and visual elements (e.g., entities are represented as rectangles, `Person` is a red rectangle, `John` is a small red rectangle). Note that special kinds of data objects which are naturally visual, such as images, charts, forms, force the visual representation to adhere to their natural representation.

4. *Perceptual rules*. Perceptual rules tell us how the user perceives a visual symbol (i.e. a line, a geometric figure, an icon, etc.), relationships between symbols (i.e. the mutual placements of two figures on the plane), and which is the perceptual effect of relevant visual attributes such as color, texture, etc.

In the prototype a suitable set of rules, covering a subset of the overall DARE theory, has been implemented, namely:

- visual rules, characterizing points and simple 2D/3D figures;
- data rules, associated with the concepts of relation, tuple, attribute, and domain; mapping rules, specifying the association among the relational attributes and the visual attributes; and
- perceptual rules, concerning the best representation of relational attributes.

Using such rules and based on the query result cardinality, the system is able to automatically choose the suitable visual elements to adopt to represent the tuples and the best visual representation for the attributes involved in the query. The result representation can be further manipulated by the user. It is possible to pick up with the mouse a single point and use sliders to restrict the result. Moreover, several "visual data mining" primitives are available and it is possible to project data on the three planes or to cluster them in order to visually capture data distribution and data relationships.

Dare has been used to display data in medical applications [10] and to display metadata in visual data mining system [21]. The effectiveness of such approaches pushed us to exploit the Dare system to evaluate metadata about feature vectors effectiveness.

4 Rank- and Distance-based Visualization

4.1 Visual Exploration of Retrieval Data Using the DARE System

4.1.1 Cut-off value

A relevant parameter to take into account in the analysis is the cut-off value (COV in what follows). Analyzing the distribution of distances from the query and its result list, a "minimum point" can be found in the distribution. Objects with a distance less then the cut value are presented to the user and considered relevant, with a distance greater than the cut value are hidden from the user and considered non relevant.

It is out of the scope of this paper to discuss in detail the adopted techniques for computing the cut-off values; we sketch here the main strategies:

- cut-off= a value that has 95% relevant objects on the left;
- cut-off= a value that has 10% nonrelevant objects on the left;
- cut-off= optimum of a weighted function (relevant-nonrelevant) objects;
- cut-off= manual setting through visual inspection of relevant/nontrelevant percentages (see, e.g., Figure 3)

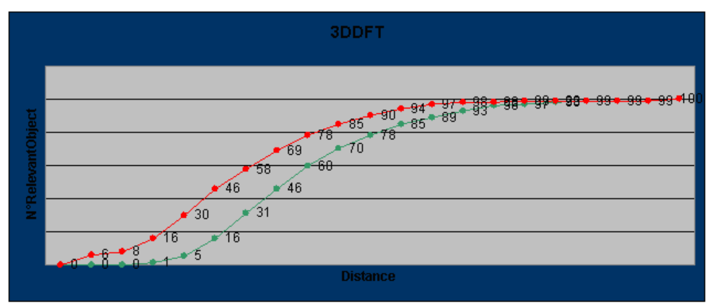


Figure 3: Relevant and non relevant percentages for 3DDFT feature vector

Considering whether an answer object comes from the same class of the query object (relevance=1) or not (relevance=0) and the cut-off value, it is possible to classify the objects in four main categories:

- True-positive (TP): relevance = 1 and distance $\leq$ COV
- False-positive (FP): relevance = 0 and distance $\leq$ COV
- True-negative (TN): relevance = 0 and distance $>$ COV
- False-negative (FN): relevance = 1 and distance $>$ COV

Note that this classification can be used also for R-Precision: we consider the cardinality of the class as a sort of cut-off value obtaining the following categories:

- True-positive (TP): relevance = 1 and distance $\leq$ class cardinality
- False-positive (FP): relevance = 0 and distance $\leq$ class cardinality
- True-negative (TN): relevance = 0 and distance $>$ class cardinality
- False-negative (FN): relevance = 1 and distance $>$ class cardinality

4.1.2 Visualization design

In our work we have first analyzed the data to import in the system and then we have found an appropriate representation for it. DARE allows for mapping data values to several visual variables: three database attributes can be mapped to three orthogonal axes (x, y, z). Colour, size and shape can also be used to map three additional database attributes to distinct visual features.

The main idea is to perform different queries using the different available feature vectors and visually compare the result sets through Dare. We used 906 different 3D objects, classified in 90 distinct classes. The classification depends on the nature of the objects, and is based on human-based classification of object. As an example, the objects are classified as being animals, plants, vehicles, etc. For each benchmark object and feature space, a query was run, ranking all the returning objects by increasing distance.

In the following we describe three different visualization designs that classify the query results comparing the distances of the retrieved objects with their relevance, estimated through R-Precision and cut-off values.

R-Precision based visualizations

The idea is to focus on one classe(es) computing a query for each element in the class(es) using a specific feature vector. Queries can range on the whole dataset or on a set of specific classes and the object in the result are ranked and associated with a four values category, true-positive, false-positive, true-negative, and false-negative. Elements IDs correspond to a categorical attribute and are assigned to X axes, ranks values corresponds to integers and are mapped on the Y axisis, and category values correspond to a categorical attributes and are mapped on four high distinguishable colors:

- GREEN (TP): rank $\leq$ class cardinality $\wedge$ relevance = 1
- YELLOW (FP): rank $\leq$ class cardinality $\wedge$ relevance = 0
- RED (TN): rank $>$ class cardinality $\wedge$ relevance = 0
- ORANGE (FN): rank $>$ class cardinality $\wedge$ relevance = 1

In this way the user can easily perceive how relevant objects (green and orange) are characterized by the actual feature vector in terms of true-positive and false-negative. As an example, Figure 4 is obtained considering the Princeton class 53 (arms), ranking the first 200 closest object in the database using the 3DDFT feature vector and assigning to each result the corresponding category comparing the rank with the class size (19), i.e., considering the R-precision. The x axis contains the 19 identifiers of class 53 objects; the y axis the ranks. Objects are coloured according to their relevance and rank.

Cut-off based visualizations

Using cut-off instead of r-precision leads to a quite similar design. The only difference is that here we consider distances (still mapped on the y axes) instead of ranks, giving the end user the perception of the results distribution. As an example, Figure 8 is obtained considering the Princeton classes 63 (pianofortes) and 77(see-saws) and using the 3DDFT feature vector. In this case objects are coloured according to their relevance and their distance w.r.t. the overall cut-off value and we added two lines, the vertical one separating the two classes (63 on the left) and the horizontal one representing the actual COV value. It is quite evident that 3DDFT performs poorly on class 77. Moreover the image does not contains yellow values: no irrelevant objects are below the cut-off value.

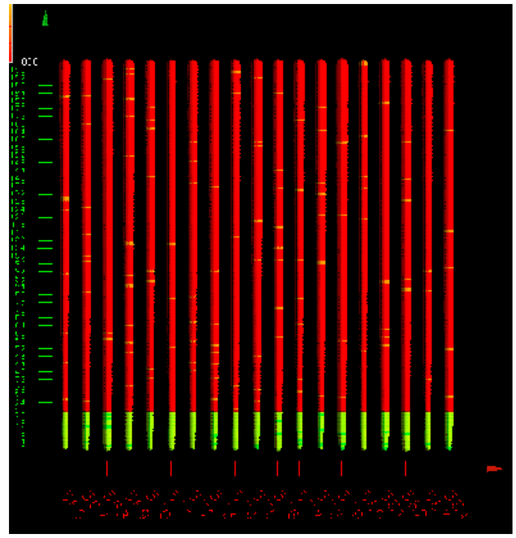


Figure 4: R-Precision for class 59: all ranks

3D visualizations

In order to compare several FVs at the same time we explored different 3D visualizations, e.g., the one depicted in Figure 6. The images refers to the FVs COR, H3D, and SIL. We realized that the more the compared feature vectors the more the occlusion problems and we decided to represent 3D images as a set of 2D projections, as described in Section 5.2.1.

4.2 Rank-Based User Interface

In online retrieval in a system with multiple feature vector representations available, the user can principally chose which feature vector to use for executing a given query. Usually, different feature vectors yield different results, and it is not clear which feature vectors best suit the user need. The standard retrieval paradigm suggests to present a list of top-n matches to the user for inspection. E.g., most Internet and multimedia search engines present the user with a short list, e.g., the first ten answers. However, in multimedia retrieval, due to the fuzziness of the similarity notion, relevant objects may be located much farther behind in the answer list. Another observation often made is that within the neighborhood of relevant objects, we find additional relevant objects. Based on this reasoning, we develop a simple, yet powerful user interface to handle multiple feature spaces for on line retrieval.

The user interface should support two key feature. Firstly, it should let the user effectively access not only a short prefix, but the whole (potentially, very long) list of answers. Secondly, it should let the user access not only the singe ranking of a currently selected feature space, but visualize the results of the query in all the feature spaces in parallel. This in turn requires scalability of the vi-

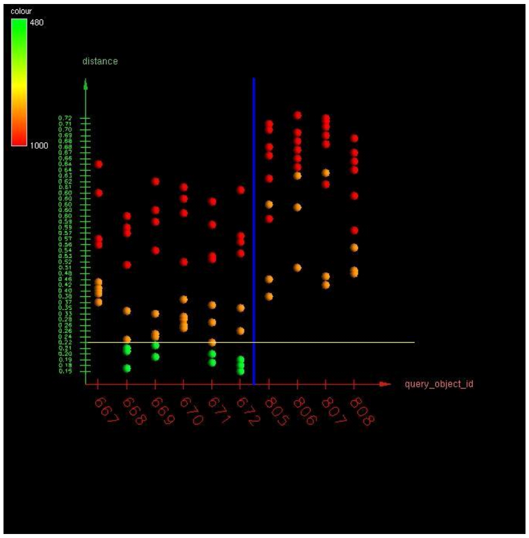


Figure 5: Cut-off value for classes 63 and 77

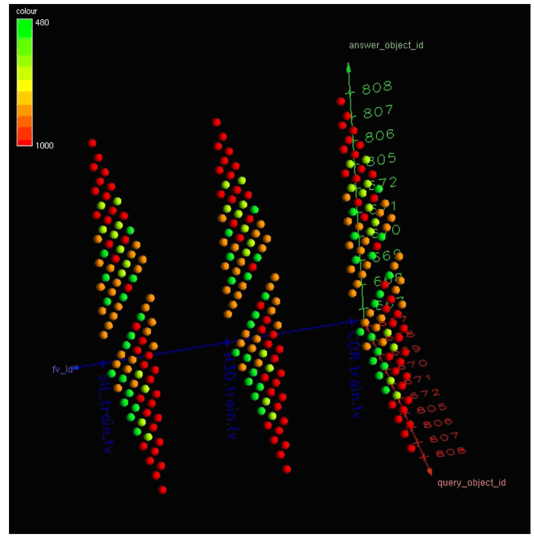


Figure 6: Comparing three FVs capabilities

sualization w.r.t. data size, as the standard thumbnail-preview approach will not work in this case. We therefore resort to the pixel-based visualization paradigm, as this scales much better with the data set size as other, more object-centered result visualization approaches. Specifically, we adapt the multi resolution pixel/rectangle approach introduced in [16]. That work suggested to represent long sequences of values by rectangular cells of varying resolution which get eventually scaled down to the pixel level. Together with an appropriate color mapping, this approach supports the visualization of long (real-valued) time-series data, but may as well be employed for visualization of categorical (boolean) data. Usage of a multi resolution approach to scaling the rectangles was proposed in [16] to improve the scalability with data size, and at the same time, allow easy perception of the most important data sections (e.g., the most current parts of the data).

We configure the display as follows. We represent the full list of answer objects under a given feature vector as a sequence of rectangular cells. We assume a currently selected (main) feature vector space is given, and that the user has already browsed the respective list of answer objects, marking answer objects of interest. For each available feature vector space, we draw a multi resolution position grid in a row-by-row manner. In each row, we highlight the cell positions corresponding to the occurrence of the selected answers in the respective feature vector spaces. Given the user has marked several objects as relevant, this visualization gives a compact, concise overview over the positions of the marked objects in the available feature spaces. The visualization at the same time serves as an interface to switch the different feature spaces for querying, and navigate in the respective feature spaces. Specifically, clicking into the rank display brings up the answers in the surrounding positions of the respective feature space and answer position the user has clicked on. This is proposed as an effective interface to (a) quickly identify candidate feature spaces and rank positions to explore answer objects, and (b) navigate the respective results list. Figure 7 illustrates the concept using a single resolution grid for each feature space.

5 Application and Evaluation

In this Section, we apply the rank-based visualization methods on a data set from the multimedia retrieval domain. We first briefly describe the data set, and then demonstrate the application of visual benchmarking and the relevance-based query interface for multiple feature vectors.

5.1 Used Data

We use the Princeton Shape Benchmark [26] (PSB) train partition to demonstrate our visualization techniques. The PSB train partition comes from the 3D model retrieval domain and consists of 907 3D meshes representing real-world objects like animals, humans, vehicles, and so on. The models were carefully classified into 90 equivalence classes according to geometric shape. We extracted 12 different types of feature vectors of different discrimination power, for each of the models. The feature vectors were obtained by first normalizing the mesh models for scale, position, and orientation, and then, various feature vector descriptors were extracted from surface, volumetric, and image-based mesh properties. The feature vectors describe the geometry of complete models and can each be used for content-based retrieval of 3D models based on global geometric characteristics. The feature vectors are described in more detail in [5], where also detailed retrieval precision results for each of the 3D feature extractors are given.

We used the feature vector representations of the models and engaged the L_1 norm (Manhattan distance) to produce exhaustive object rankings for each of the 3D models (simulating a query) and each feature vector. We evaluate the PSB ground truth classification to flag each answer object as being relevant to the query object, or not. We capture the wealth of rankings in a large table with the following schema: $RANKINGS(INT : fvID, INT : qID, INT : aID, REAL : distance, BOOL : relevant)$, where *fvID* refers to the key of the used feature vector, *qID* and *aID* refer to the keys identifying the query an answer objects, *distance* gives the L_1 distance between the query qID and the answer aID under feature vector fvID, and *relevant* indictes whether qID and aID come from the same benchmark class. This table can conveniently be loaded into the DARE system, where the data for individual queries under different feature vectors is retrieved by selection of respective table tuples.

5.2 DARE application

5.2.1 Explorative Visual Benchmarking

We performed in batch way all the possible queries with all available feature vectors, ranking all results; after that we assigned to

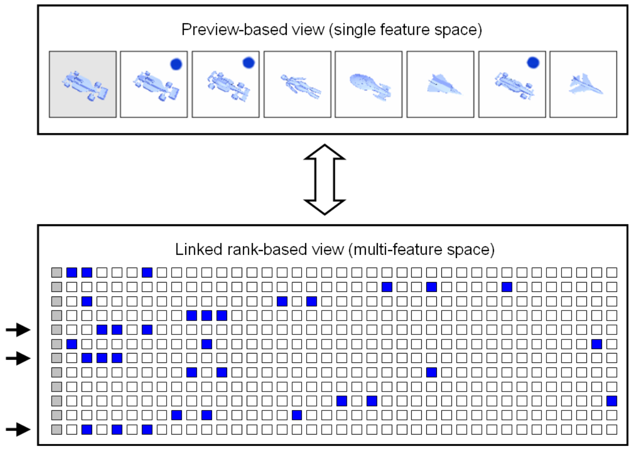


Figure 7: Usage of the rank visualization as an efficient UI tool. Objects marked in the preview/thumbnail answer set are also marked in the rank visualization, which in parallel gives the rankings of the same query object, but within different feature vector spaces. Our technique is an efficient extension of the standard preview interface, which itself does not scale well with the number of elements shown. This UI supports the user in *interactive feature selection* - she can see which feature spaces contain promising object clusters, explore the respective rankings, and possibly find additional relevant objects.

each result item a r-precision based category and a COV based category. Using this set of data we performed several visual analysis, observing the effectiveness of a specific feature vector on different classes and/or the effectiveness of different feature vectors on the same class(es). In the following we report some of the interesting clues the system designers got using the Dare system.

A first example is on Figure 4 exploring the effectiveness of the feature vector 3DDFT on class 53 with respect to r-precision. It is quite evident that the actual FV performs poorly on class 53: few green items are in the first 19 ranks, while orange items (false-negative) exhibit very high ranks. In order to better understand the FV performance on the class the user zooms on the first 19 ranks (see Figure 8).

The designer discovers a quite odd behavior: the first 19 ranks for objects 566, 570, 571, 572, 574, and 580 do not contain relevant objects (only yellow dots) while objects 577, 578, and 581 have three relevant objects in the closest rank. A possible explanation is that class 53 contains non homogeneous objects and we removed it from our experiments.

In order to better understand the 3DDFT behavior the designer compares it against COR and XVT features vectors using the cut-off value (see Figure 9). 3DDFT performs better than CPX and worst than COR. Moreover, it is possible to understand that 3DDFT better preserve the results ordering: in most cases we have a green-orange-red sequence, while COR exhibits puzzled patterns. That implies that in application in which the right ordering is a strong requisite is better to use 3DDFT, even if produces less accurate results (w.r.t. COR). Moreover, it is worth noting the interesting perceptions that the visual representation provides for, e.g., the distance gaps among red items and the green/yellow/orange ones provided by 3DDFT.

Finally, we show a comparison of all 12 feature vectors against two classes, 63 (pianofortes) and 77(see-saws), and using the cut-of value (see Figure 10. Inspecting the image it is possible to get the following interesting clues:

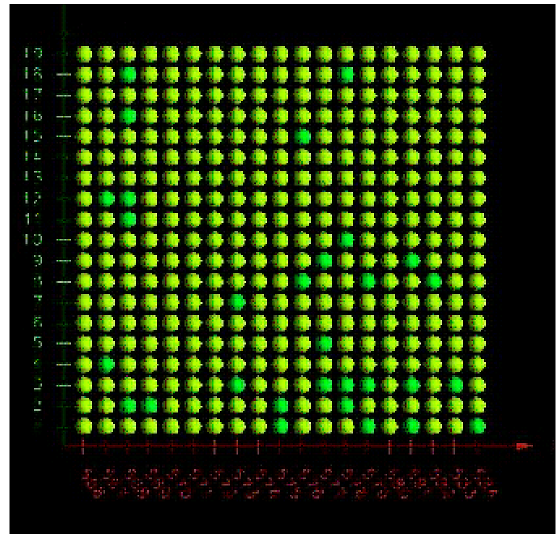
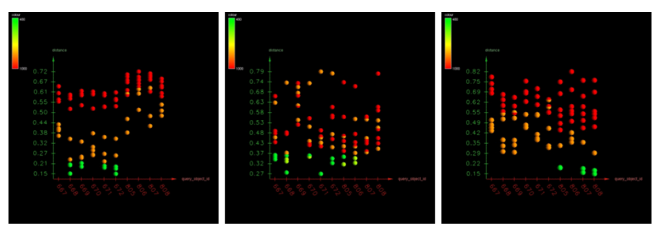


Figure 8: R-Precision for class 59: first 19 ranks

- All the FVs have a better behavior for the class 63 respect to the class 77;
- H3D, DSR, and GRAY have a good behavior for both classes;
- CPX and DBF have a bad behavior in general;
- 3DDFT, RIN , DSR and SD2 have a good behavior for the class 63;
- PMOM and H3D have a good behavior for the class 77;
- 3DDFT, RIN, SD2, and SIL have a bad behavior for the class 77;
- DSR outperforms the other FVs, but has a very bad behavior for object 806.



3DDFT COR CPX

Figure 9: Comparing three FVs capabilities

5.3 Application of the Rank-Based User Interface

We implemented the rank-based interface for querying in systems providing multiple feature vectors, as described in Section 4.2. Specifically, we set the multi resolution display to three different levels of resolution: The first column contains the first 50 positions in a 1×50 grid layout. The second and third columns contain the next 200 and 800 ranks, in grids of size 4×50 and 16×50, respectively. This layout implies that from one level to the other, the number of ranks visualized increases by a factor of 4. The display

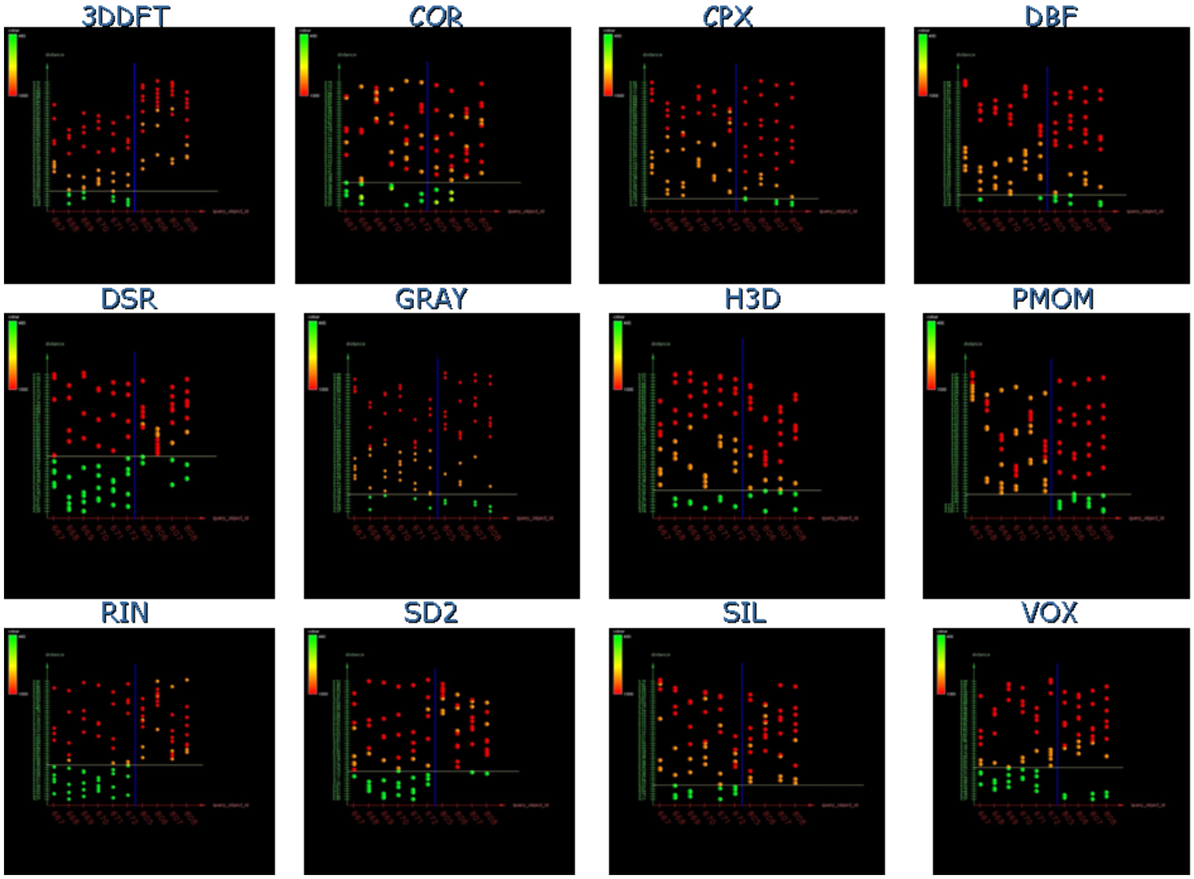


Figure 10: Comparing 12 FVs capabilities

thereby focuses user attention on the first ranks, while keeping the remaining positions in context.

Figure 11 shows the results for two different queries. The rows are arranged by average benchmarking results of the feature vectors, placing the best benchmarked single feature vector on top. That feature vector is usually the one which is initially used for conducting a query. As the user browses the top-n list returned and marks objects as relevant or of interest, the positions of the marked objects are simultaneously highlighted in the multi resolution display. The user is able to effectively search for clusters of the marked objects in different feature spaces in parallel. The user is able to switch to rank intervals of other feature vectors by a simple mouse click. We observe that this condensed rank-based interface is an effective way for spark the users interest for exploration of diverse ranking intervals within the different feature spaces, as guided by the recognition of interesting occurrence patterns.

We point out that the multi resolution display, in addition to showing the occurrence of marked answer objects, may also be used to encode other information available from the query execution. E.g., we may use it to display the distribution of distances in feature space, screening the answer rankings for outlier distances, possibly indicating abrupt changes in the sequence of answer objects. Figure 12 shows a distance image obtained by calculating the difference of distances between the query objects and answer objects adjacent to each other in the ranking. The visualization was obtained by linearly mapping the sequence of distance differences to a bright-dark color scale. In the shown example, the ranking in the 4^{th} row from the bottom exhibits several outlier distances among the first 50 positions, which suggests closer inspection by the (expert) user.

6 Conclusion

In this paper, we have proposed visualizations for visual benchmarking of competing feature vector spaces, and for supporting the interactive feature selection by the user in on line retrieval mode. We applied the techniques on a retrieval benchmark data set from the field of 3D model retrieval.

Future work includes testing the methods on additional data sets and improving the techniques. Regarding the multi feature query interface, we like to integrate it with additional relevance feedback interaction techniques. Combining efficient methods for capturing relevance feedback and advanced rank visualization is expected to lead to retrieval systems of increased retrieval effectiveness.

Acknowledgements

This work was supported by the European Commission within the DELOS Network of Excellence on Digital Libraries (www.delos.info). We thank Dietmar Saupe and Dejan Vranic for providing 3D FV extractors and for valuable discussion.

References

[1] Z. E. Ahmed. Special Issue on Intelligent Visualization Systems. *Journal of Visual Languages and Computing*, 5, 1994.

[2] J. Aßfalg, H.-P. Kriegel, A. Pryakhin, and M. Schubert. Multi-represented classification based on confidence estimation. In *PAKDD*, pages 23–34, 2007.

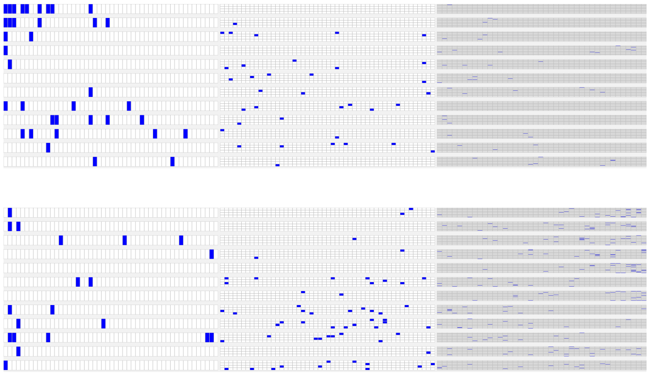


Figure 11: Multi resolution rank visualizations for two different queries. Occurrences of marked obejcts are highlighted in blue. The display is proposed as an efficient feature selection and result navigation interface for multi feature vector retrieval scenarios.

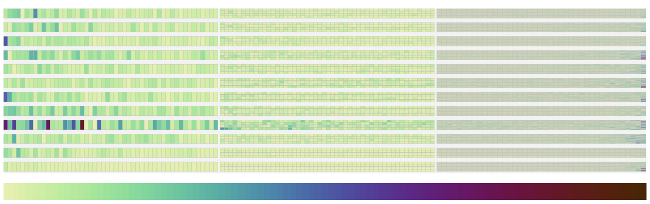


Figure 12: Visualization of normalized distance differences between adjacent answer positions, using the multi resolution layout. The bright-dark color map shown below visualizes normalized distances.

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

[4] B. Bustos, D. Keim, D. Saupe, and T. Schreck. Content-based 3d object retrieval. *IEEE ComputerGraphics & Applications, special issue on 3D documents*, Jul/Aug 2007. to appear.

[5] B. Bustos, D. Keim, D. Saupe, T. Schreck, and D. Vranic. An experimental effectiveness comparison of methods for 3D similarity search. *International Journal on Digital Libraries, Special Issue on Multimedia Contents and Management*, 6(1):39–54, 2006.

[6] S. Card, J. Mackinlay, and B. Shneiderman. *Readings in Information Visualization*. Morgan Kaufmann, 1999.

[7] T. Catarci, M. Costabile, S. Cruz, and G. Santucci. Foundations of the dare system for drawing adequate representations. In *Proc of Itl Symposium on Database Applications in Non-Traditional Environments (DANTE 99)*, 1999.

[8] T. Catarci, M. Costabile, S. Levialdi, and C. Batini. Visual Query Systems for Databases: A Survey. *Journal of Visual Languages and Computing*, 8(2):215–260, 1997.

[9] T. Catarci, M. Costabile, and M. Matera. Visual Metaphors for Interacting with Databases. *ACM SIGCHI Bulletin*, 27(2), 1995.

[10] T. Catarci, G. Santucci, and S. F. Silva. An interactive visual exploration of medical data for evaluating health centers. *Journal of Research and Practice in Information Technology*, 2003.

[11] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. Wiley-Interscience, New York, 2nd edition, 2001.

[12] FADIVA-ESPRIT Working Group. Foundations of Advanced 3D Information Visualization. `http://www-cui.cs.darmstadt.gmd.de:80/visit/activities/IEEE/Fadiva`.

[13] C. Faloutsos. *Searching Multimedia Databases by Content*. Kluwer Academic Publishers, Norwell, MA, USA, 1996.

[14] E. Haber, Y. Ioannidis, and M. Livny. Foundation of Visual Metaphors for Schema Display. *Journal of Intelligent Information Systems*, 3:263–298, 1994.

[15] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kauffman, 2nd edition, 2006.

[16] M. Hao, D. Keim, U. Dayal, and T. Schreck. Multi-resolution techniques for visual exploration of large time-series data. In *Eurographics/IEEE-VGTC Symposium on Visualization, 23 - 25 May 2007, Norrkping, Sweden*, 2007.

[17] D. Harel. On Visual Formalism. *Communications of the ACM*, 31(5):514–530, 1988.

[18] S. Havemann and D. Fellner. Seven research challenges of generalized 3d documents. *IEEE Computer Graphics and Applications, Special Issue on 3D Documents*, 27(3):70–76, May/Jun 2007.

[19] D. Heesch and S. Rüger. Performance boosting with three mouse clicks – relevance feedback for CBIR. In *Proc European Conf Information Retrieval*, pages 363–376. LNCS 2633, Springer, 2003.

[20] S. Jayanti, Y. Kalyanaraman, N. Iyer, and K. Ramani. Developing an engineering shape benchmark for cad models. *Computer-Aided Design*, 38(9):939–953, 2006.

[21] S. Kimani, S. Lodi, T. Catarci, G.Santucci, and C. Sartori. Vidamine: A visual data mining environment. *Journal of Visual Languages and Computing, Special Issue on "Visual Data Mining*, 2003.

[22] H.-P. Kriegel, P. Kröger, P. Kunath, and A. Pryakhin. Effective similarity search in multimedia databases using multiple representations. In *MMM*, 2006.

[23] J. Mackinlay. *Automatic Design of Graphical Presentations, Ph.D. Thesis*. Department of Computer Science, Stanford University, 1986.

[24] R. Reiter and A. Mackworth. A Logical Framework for Depiction and Image Interpretation. *Artificial Intelligence*, 41:125–155, 1989.

[25] T. Schreck and C. Panse. A new metaphor for projection-based visual analysis and data exploration. In *Proc. IS&T/SPIE Conference on Visualization and Data Analysis (VDA)*, 2007.

[26] P. Shilane, P. Min, M. Kazhdan, , and T. Funkhouser. The princeton shape benchmark. In *SMI '04: Proceedings of the Shape Modeling International 2004*, 2004.

[27] E. R. Tufte. *The Visual Display of Quantitative Information*. Graphics Press, 1983.

[28] E. R. Tufte. *Envisioning Information*. Graphics Press, 1990.