

# iVi: An Enhanced Query Result Visualization for Image Databases

Anke Schneidewind, Ingo Schmitt, Petra Neumann

**Many current content-based image retrieval systems suffer from poor result presentation. A sophisticated visualization can be used to identify differences between human perception of similarity and system-determined similarity. Therefore, we develop a visualization tool named iVi to present feature data of a query result on three different levels of abstraction. Besides a data table we use a parallel coordinate plot, and a color space plot to extend the result presentation. In this paper we will discuss our experiences with iVi.**

## 1 Introduction

The central problem of content-based image retrieval (CBIR) is to deal with the well-known semantic gap [10] between human perception of similarity and computed similarity. Given a query image a retrieval system extracts feature values and applies some similarity or dissimilarity measures between the feature values to find matching result images. Without understanding the system's way of 'perceiving' similarity, the user is not able to give an exhaustive, system oriented, relevance feedback.

Within the framework of our project we want to find out to what extent we can reduce the semantic gap with approaches based in the information visualization. We want to feed the user with internal information without expecting too much of him. Therefore we followed the definition of the user-level pyramid of visualization [2] and defined three levels of abstraction a low one for experts, one for power users, and a high abstracted for the casual user. Each level reveals parts of the extracted features and calculated similarity. Understanding the highest abstracting level a user can switch to a lower one and examine this with the knowledge gained from the higher one.

There are a number of CBIR systems which present more result information than just a list of images [3,7,6,9]. These systems show feature details of the result images or an two- or three-dimensional overall view of the database images for browsing. We differ from these systems by setting the result features as the focus point of our visualizations.

Connections between the areas of Information Visualization and CBIR are made by Chen et al. [4], Nürnberger and Klose [8], or in the CBIR system El Niño [11]. These approaches employ graphs on clusters, visualized self-organizing maps or other visualizations techniques such as fisheye to visualize cluster of similar images or browse through the database. Since our aim is to identify single feature values which separate pairs of images clustering and browsing is not our point of interest.

Disadvantage of these approaches is that the visualization is too abstract for understanding the applied similarity. Our aim is to simplify it in several views. Our iVi-Tool (Information Visualization for Images) allows users to switch among three views of result visualization: a simple data table, a parallel coordinate plot [12] pp. 362–365, and color space plot. Following the re-

strictions to a certain maximum number of presentable dimensions and using graphical attributes like size, position, and color, as described in [12] pp.188–199, our views show more detailed information than it can be achieved with a result list or the above-mentioned approaches. To test our approach we conducted an experimental study on an image database<sup>1</sup>.

The project is embedded in the research work on multimedia databases of the database group in the Otto-von-Guericke-University Magdeburg.

In the following we will introduce our iVi-Tool especially focusing on the visualization part. Using example scenarios we will demonstrate the potentials of our approach.

## 2 The iVi-Tool

Our system is implemented on top of an Oracle database system and comprises an indexing and query component as well as a visualization tool. The images and their respective feature values are managed by the database system. The indexing and query component supports query formulation and fast algorithms for nearest neighbor search in high dimensional space [1]. The visualization tool presents query results graphically and is implemented as a framework in JAVA employing the java3D package. Additional extraction methods and visualization views can easily be inserted into the framework.

The indexing component is responsible for extracting and indexing feature values from images stored in the database system. We implemented two methods for color extraction and one for texture. For our visualization experiments we use the most common image feature, the color histogram. Taking the human perception into consideration we use the three dimensional  $L^*a^*b$  color space [12] pp. 115–118. We use an L-histogram with 15 bins, and an a- and b-histogram each with 10 bins. Concatenating these three histograms altogether provides one color feature. The idea behind the higher amount of L bins for the luminance, is that we want to provide gray scale images.

We provide queries by example: The user can upload a query image and specify the number of result images, the extraction method (in our case only the  $L^*a^*b$  color extraction). For the calculation of the dissimilarity between the features of the images we choose the euclidean distance.

The visualization tool supports three different views on the result data. Every view visualizes data on a different level of abstraction. In an increasing order of abstraction we have:

<sup>1</sup> The colored figures of the study results can also be seen at <http://www.witi.cs.uni-magdeburg.de/~anschnei/forschung/>

1. *data table*: At the lowest abstraction level we suggest a data table containing the feature data of the query result. Every row represents a feature vector and result image respectively. The rows are sorted by calculated image similarity.
2. *parallel coordinate plot*: The parallel coordinate plot visualizes the feature data for the L-, a-, and b-values. The plot is at a higher abstraction level than the data table level since the values are interpreted graphically, see Figure 1.
3. *color space plot*: This view is at the highest level of abstraction. It graphically interprets the average color, the rank and the dissimilarity value. The feature vectors are visualized as annotated spheres (Figure 2) or thumbnails (Figure 3a,4) in a three-dimensional color space.

In a list of ranked result images the user can interactively mark those images which shall be visualized. A menu allows to switch among the different view. Each view applies different examination options like zoom, rotate, and shift, see Figure 1. In the following we will briefly introduce view 2 and 3.

**Parallel coordinate plot**

The parallel coordinate plot visualizes feature data of several images as graphs, shown in Figure 1. The plot does not depend on the feature data type. To ease identification of images and their corresponding features the image borders and the visualized features are colored the same.

The x-axis expresses the dimension and the y-axis visualizes the corresponding parallel coordinate plot values. For the color feature we displayed the allocation of the colors to the axes L, a, and b, see at the bottom of Figure 1.

The parallel coordinate plot allows a visual comparison of feature values. This shall help users to perform feature based relevance feedback.

**Color space plot**

The color space plot view presents the result images in a three-dimensional space. Every image is mapped to a position within the color space as a colored sphere or a thumbnail. We implemented a Lab3D and a FastMap3D view. The Lab3D is adapted to the color feature the FastMap3D is independent of the feature type.

**Lab3D view:** This view is based on the  $L^*a^*b$  color space and is shown in Figure 2 and 3a. By computing the average color of the feature values every image is assigned exactly one position in color space.

In that way we visualize every result image as well as the query image as a sphere with the following parameters:

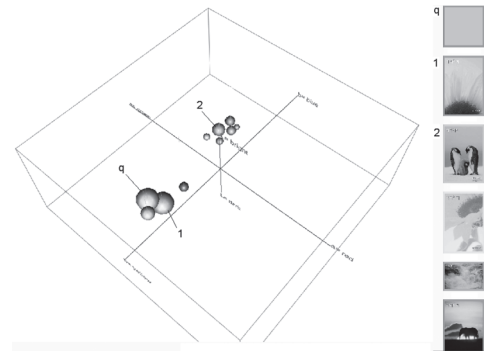


Figure 2: Color space plot, Lab3D spheres (rotated)

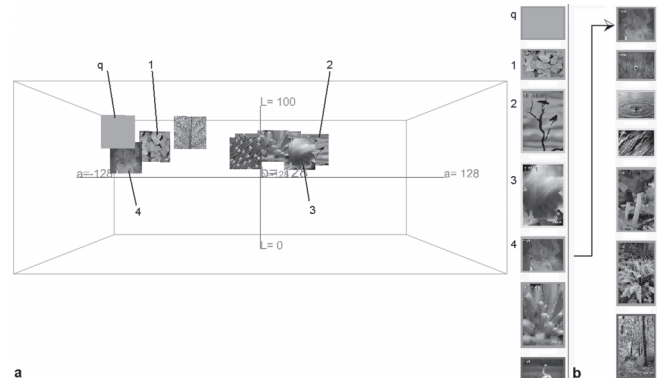


Figure 3: Color space plot, Lab3D thumbnails

- *position*: The center of the sphere is determined by the average color.
- *color*: The color of the sphere equals the average color of the image.
- *size*: The size of a sphere corresponds inversely to the distance to the query image. Similar images according to the query image are large spheres whereas dissimilar images are small spheres.

Although the ranking is based on feature data and the visualization is based on average color this view gives the user some special insights into the internal calculations:

- The user can see whether and how the calculated average color matches the user's perception of color.
- The user can see the relation between average color and calculated distance by examining size and color of similar images.
- The user sees the matching of calculated distances and positions in the  $L^*a^*b$  space by position and size of the spheres.

For a faster identification the spheres can be changed to thumbnails of the images. Especially an unskilled user can get a better understanding of how the ranking is computed.

**FastMap3D view:** With the FastMap algorithm, see [5], we calculate based on feature data, three dimensions with maximum discriminating power. Thus, the resulting FastMap axes usually do not correspond to a single color value, see Figure 4. Then, the user can identify latent dimensions which are mainly responsible for the computed dissimilarity values. Except for the position the visualization of spheres and thumbs is the same as in the Lab3D.

**Application of the views**

The decision which visualization is the suitable one depends on the user's information need, and knowledge of the features. If the user wants to analyze the features in detail the

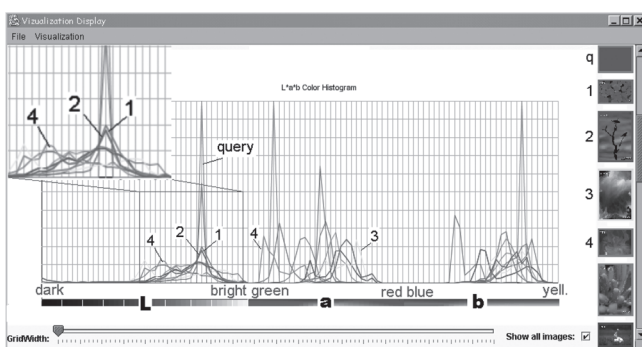


Figure 1: iVi-Tool with parallel coordinate plot view and enlarged detail

choice will be the parallel coordinate plot. For further calculations feature data of the images can be taken from the data table. To obtain a global understanding of the calculated similarity a color space plot is probably the best choice.

Usually, the user starts at a high level of abstraction, for example with the color space plot. From there, a switch to the parallel coordinate plot visualizes the underlying feature data. If an expert found some specific data of interest in both of these views he can take the tangible feature data from the data table.

### 3 Example Scenarios

The next two scenarios will demonstrate how our proposed visualization tool can help to obtain better retrieval results.

In the first scenario we use a simple light green image to demonstrate in a easy way how a user can get help from the color space plot to find in the parallel coordinate plot the regions which have significant influence on the current result.

The first result images are mainly yellow. From the Lab3D view, we observe that the average color of the query image is more yellow than green (Figure 2). The reason is that the query image fell into the yellow interval although it is near to the green interval. Therefore we get a better result by choosing a darker green for the query image (Figure 3a). Nevertheless the result shows red images on positions two and three. Examining the Lab3D view again we see that these images are positioned on nearly the same luminance value (vertical axis). Switching to the parallel coordinate plot we can determine the significant values and out of these the relevant one, see Figure 1. Without relevance feedback, we can create a new query image taking the relevant luminance values out of the data table and combine them with the chrominance values of our query image to a new query. Another possibility is to take a result image that matches the combined query best. In our case we took the fourth image as new query because the parallel coordinate plot has no (single) high luminance values that can influence the result, see result list Figure 3b. It now suits our notion. Images with multiple significant colors (or other feature) need more time to explore. The associated intervals of feature values can only be

distinguish from each other when they don't overlap too much. Therefore the amount of distinguishable colors are limited.

In future we plan to integrate relevance-feedback-mechanisms. Thus, the last two manual steps will be omitted.

In the second scenario we started with the query image seen in the right upper corner of Figure 4. The high similarity of the average color causes the spheres sticking in the Lab3D view together. However, the FastMap3D view finds automatically the right axes to discriminate good from bad result images. We can determine an new axis which meets our notion of similarity best, shown as an arrow in Figure 4. Employing the information of this axis the system can determine which combination of original feature values is mapped to that axis. To create a new query on this feature values would be an additional enhancement of the system.

### 4 Conclusion and Outlook

We presented three types of visualization on different levels of abstraction and discussed their contribution for understanding the retrieval result. Furthermore, we demonstrated their potential to unveil discrepancies between user's and system's notion of similarity.

The novelty can be found in the combination of the approaches of image retrieval and information visualization. Contrary to other systems we give the user tools to interact and examine the pure feature data. Therefore the user can estimate the relevance of a single feature value on its own.

The next step is to use visualization results to modify the retrieval system. Therefore, we will investigate the impact of our results to a feature-oriented relevance feedback algorithm. A next aim is to apply specialized views for the other features of the system like the Lab3D. For example choose three dimensions for texture, from fine granular to coarse granular, and create a texture view like the Lab3D. We also have to conduct a user study.

### References

- [1] S. Balko. *Grundlagen, Entwicklung und Evaluierung einer effizienten Approximationstechnik für Nearest-Neighbor-Anfragen im hochdimensionalen Vektorraum*. PhD thesis, Otto-von-Guericke-Universität Magdeburg, 2004. in German.
- [2] J. R. Brown, R. Earnshaw, M. Jern, and J. Vince. *Visualization: using computer graphics to explore data and present information*. John Wiley & Sons, Inc., 1995.
- [3] C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein, and J. Malik. Blobworld: A system for region-based image indexing and retrieval. In *Third International Conference on Visual Information Systems*, pages 509–516. Springer, 1999.
- [4] C. Chen. *Mapping scientific frontiers: the quest for knowledge visualization*. Springer, 2003.
- [5] C. Faloutsos and King-Ip Lin. FastMap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets. In *Proc. of the ACM SIGMOD International Conference on Management of Data*, pages 163–174, San Jose, California, 22–25 1995.
- [6] B. Moghaddam, Q. Tian, and T. S. Huang. Spatial visualization for content-based image retrieval. In *International Conference on Multimedia and Expo (ICME'01)*, Tokyo, Japan, September 2001.
- [7] M. Nakazato and T. S. Huang. 3D MARS Immersive Virtual Reality for Content-Based Image Retrieval. *Proc. of IEEE International Conference on Multimedia and Expo*, pages 22–25, August 2001.

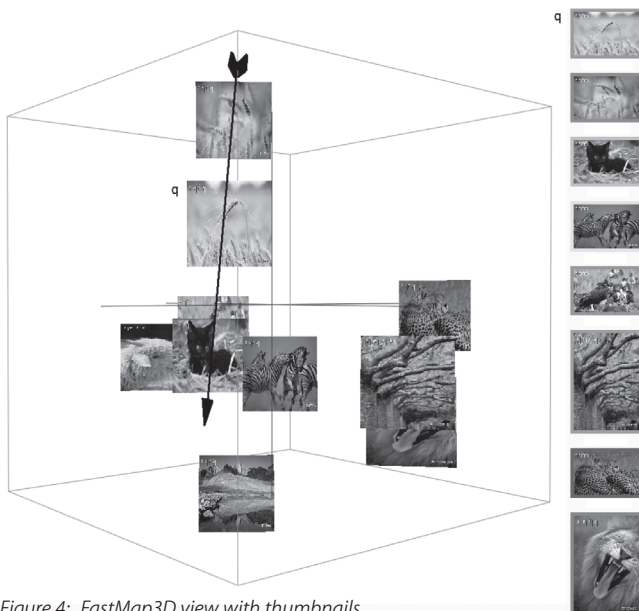


Figure 4: FastMap3D view with thumbnails

- [8] A. Nürnberger and A. Klose. Improving clustering and visualization of multimedia data using interactive user feedback. In *Proc. of the 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU)*, pages 993–1000, 2002.
- [9] Y. Rubner and C. Tomasi. *Perceptual Metrics for Image Database Navigation*, chapter „Feature Similarity“. Kluwer Academic Publishers, Boston, December 2000.
- [10] G. Salton and C. Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41:288–297, 1990.
- [11] S. Santini, A. Gupta, and R. Jain. Emergent semantics through interaction in image databases. *Knowledge and Data Engineering*, pages 337–351, 2001.
- [12] C. Ware. *Information Visualization: Perception for Design*. Morgan Kaufmann Publishers, 2000.

### Contact

Anke Schneidewind  
 Universitätsplatz 2, 39106 Magdeburg  
 email: [anke.schneidewind@iti.cs.uni-magdeburg.de](mailto:anke.schneidewind@iti.cs.uni-magdeburg.de)  
<http://www.iti.cs.uni-magdeburg.de/~anschnei/>



**Anke Schneidewind** received 2002 her computer science degree from the Otto-von-Guericke-University Magdeburg. Currently she is a Ph.D. student at the Institute of Technical and Business Information Systems. Her current research interests are information visualization and relevance feedback in multimedia databases.



**Ingo Schmitt** works in the database group of the Otto-von-Guericke-Universität at Magdeburg. For his doctorate he did research on schema integration for the design of federated databases. After he received his doctorate in 1998 he changed his research topic to multimedia databases. In 1999/2000 he worked as an assistant professor at the University of Windsor in Canada.



**Petra Neumann** is a student of Computational Visualistics at the Otto-von-Guericke University in Magdeburg, Germany. She is currently working on her Diplom thesis in the field of Information Visualization at the Department of Simulation and Graphics. Her research interests include Human-Computer Interaction and Information Visualization.