MACRO-LEVEL SIMILARITY MEASUREMENT IN VIZIR

Horst Eidenberger and Christian Breiteneder

Vienna University of Technology, Institute of Software Technology and Interactive Systems
Favoritenstrasse 9-11 / 188/2, A-1040 Vienna, Austria
{eidenberger, breiteneder}@ims.tuwien.ac.at

ABSTRACT

This paper analyzes the similarity measurement in Content-based Image and Video Retrieval systems (CBIR). The goal is to identify preliminaries for successful queries as the basis for the implementation of a query engine in the Content-based Visual Information Retrieval framework (VizIR). VizIR is an open CBIR framework for researchers, software developers and instructors. Past efforts in CBIR have lead to several general-purpose prototypes. However, these prototypes differ in implemented feature classes, user-interfaces and similarity measurement. VizIR aims at overcoming this unsatisfactory situation. The paper overviews wide-spread techniques for similarity measurement in CBIR, derives a general querying model and proposes conditions for similarity measurement algorithms on the macro-level. Based on these conditions two methods (the Linear Weighted Merging method and the Query Model approach) are evaluated and the superior method chosen for the VizIR project. Additionally, the major goals of the VizIR project are outlined and interested researchers are invited to participate in the project.

1. INTRODUCTION

In this paper we analyze the similarity measurement in Content-based Image and Video Retrieval systems (CBIR). The goal is to identify preliminaries for successful queries as the basis for the implementation of a query engine in the Content-based Visual Information Retrieval framework (VizIR). VizIR is an open CBIR framework for researchers, software developers and instructors (see Section 3 for details).

CBIR ([8]) is the attempt to search for visual content in media databases by deriving meaningful features and measuring the dissimilarity of visual objects by distance functions. Major advantages of CBIR systems are fully automated indexing and the description of visual content by visual features. Recently, the MPEG-7 standard for Multimedia content description was finalized. It contains a visual part with descriptors (features) for image and video objects. Nevertheless, CBIR is still an area of intense research. Each year, prototypes with new intuitive user-interfaces and sophisticated methods for iterative refinement, new querying methods and many other innovations are introduced.

The rest of this paper is organized as follows: Section 2 points out relevant related work, Section 3 is dedicated to the VizIR project goals, and in Section 4 we revisit the content-based querying process and propose conditions for feature merging. In Section 5 we analyze the linear weighted method for feature merging and finally, in section 6 we explain how querying will be implemented in VizIR.

2. RELATED WORK

Past efforts in CBIR have lead to several general-purpose prototypes like QBIC ([3]), Virage ([1]), VisualSEEk ([9]), Photobook ([5]) and MARS ([4]). Next to the implemented feature classes and user-interfaces these prototypes differ in their similarity measurement.

Usually, CBIR similarity measurement follows the Vector Space Retrieval model and is done by measuring the distances of feature vectors with distance functions that are based on the Metric Axioms, combining the distance values of a single object for multiple features by a merging function to a distance sum and presenting the user the objects with the lowest distance sum as the most similar ones. In Section 4 we will introduce a general model for CBIR querying.

According to the Metric Axioms distance measures *d()* have to fulfill four conditions ([6]):

1. Constancy of self-similarity:
$$d(f_A, f_A) = d(f_B, f_B)$$
for the feature vectors f_A and f_B of two stimuli A and B (in CBIR: media objects).
2. Minimality:
$$d(f_A, f_B) \geq d(f_A, f_A)$$
3. Symmetry:
$$d(f_A, f_B) = d(f_B, f_A)$$
4. Triangle inequality:
$$d(f_A, f_B) + d(f_B, f_C) \geq d(f_A, f_c)$$

Distance measures that fulfill the Metric Axioms are Minkowski distances, the Euclidean distance and the City Block measure. Experimental investigations during the last fifty years have turned out that Metric Axioms may be too restrictive for human similarity perception. The triangle inequality (in CBIR sometimes used for query acceleration) was even falsified ([6]). Newer theories as e.g., Monotone Proximity Structures or Tversky's

Feature Contrast model suggest a better representation of human similarity perception.

In many CBIR prototypes (e.g., in [3], [1]), when multiple features are employed for a query, the result set is ordered by a ranking value derived from the weighted sum of the distance values (position value). This method is called Linear Weighted Merging. The position value for each database object is defined by the following equation:

$$Position\,value_{Object} = \sum_{i=1}^{F} w_i d_i$$

F represents the number of features, w_i the weight for feature i and d_i the distance value for feature i between the query object and the database object. This evaluation method assumes that all distance functions are normalized to the same interval (f. e. [0, 1]). Its major advantages are the simple calculation and application. The major disadvantages are first, the fact that not all features show a linear relationship and linear merging therefore is not a suitable method to combine such features and second, that in most systems weights have to be provided by the user who is normally overtaxed by this task ([7]).

For these reasons, the authors of [4] propose the employment of the Boolean Model instead of Linear Weighted Merging. According to this model two stimuli A and B are similar for a certain feature F, if they fulfill the following condition v_i:

$$d_F(f_A, f_B) \leq \delta_f$$

δ_f is called a degree of tolerance. It is a threshold for the maximum distance of two stimuli. In Boolean retrieval multiple conditions v_i can be combined by logical operators. The result set consists of those stimuli that fulfill all *AND*-combined sub-expressions. Boolean retrieval leads to better results than Linear Weighted Merging but has the major drawback that it does not rank the stimuli in the result set. Before we go into details of the querying process in VizIR, we will outline the project goals.

3. VIZIR PROJECT GOALS

The goal of the VizIR project is to develop an open CBIR prototype as a basis for teaching and further research in various directions. The term open means that VizIR will be free software (including the source code) and extensible. VizIR was started in summer 2001 as a conclusion of experiences gained with earlier CBIR projects and is currently evaluated for scientific funding. The motivation behind VizIR is: an open CBIR platform would make research (especially for smaller institutions) easier and more efficient (because of standardized evaluation sets and measures, etc.).

The VizIR project aims at the implementation of successful methods for automated information extraction from images and video streams, definition of similarity measures that can be applied to approximate human similarity judgment and new, better concepts for the user interface aspect of visual information retrieval, particularly for human-machine-interaction for query definition and refinement and video handling. This includes the implementation of a working prototype system that is fully based on the visual part of the MPEG-7 standard for multimedia content description. Reaching this goal requires the careful design of the database structure and an extendible class framework as well as seeking for suitable extensions and supplementations of the MPEG-7 standard by additional descriptors and descriptor schemes, mathematically and logically fitting distance measures for all descriptors (distance measures are not defined in the standard) and defining an appropriate and flexible model for similarity definition. MPEG-7 is not information retrieval-specific. One goal of this project is to apply the definitions of the standard to visual information retrieval problems.

Additionally, we want to develop integrated, general-purpose user interfaces for visual information retrieval. Such user interfaces have to include a great variety of different properties: methods for query definition from examples or sketches, similarity definition by positioning of visual examples in 3D space, appropriate result display and refinement techniques and cognitively easy handling of visual content, especially video. Finally, VizIR will include methods and test sets for benchmarking (measurement of retrieval quality), performance evaluation (query execution time, etc.) and usability testing of the user interfaces.

The VizIR project intends to integrate various directions of past and current research in an open framework to push CBIR research and teaching towards practical usefulness by overcoming some of the serious problems. In the next section we will focus on the querying aspect, outline the general CBIR querying process and propose conditions for feature merging.

4. CONTENT-BASED QUERYING PROCESS

Usually, the CBIR querying process for a set of example stimuli and an input data set consists of the following three steps (see Figure 1 for an example):

1. Feature extraction–The properties of stimuli (e.g. images, video clips) are extracted by feature extraction functions and stored as descriptor vectors. This steps transforms the media space into feature space. Normally, only the features of the example stimuli have to be extracted during the querying process. The descriptors of the data set are fetched from a database.
2. Micro-level similarity measurement–The dissimilarity values for all features between an example stimulus and elements of the data set are measured with distance functions. Ideally, the output of all distance functions in a CBIR system should be normalized to the same range of values. This step transforms feature space into distance space, where each media object is represented by a vector of distance values.
3. Macro-level similarity measurement–In this step a decision is derived from the dissimilarity values of all features for each stimulus in the data set, if it is similar to the example stimuli or not. The most similar stimuli are ranked and returned as an ordered result set.

Today, rules exist for the first and second step, how they should be performed and which constraints should be kept. MPEG-7 descriptors should be used for feature extraction and distance measures should be based on the Metric Axioms (see Section 2), Ordinal Properties (see [6]) or another similarity model. To the authors' surprise no such rule set exists for the third step. Since such rules would be a valuable help for CBIR system developers we will propose four conditions for macro-level similarity measurement in the following.

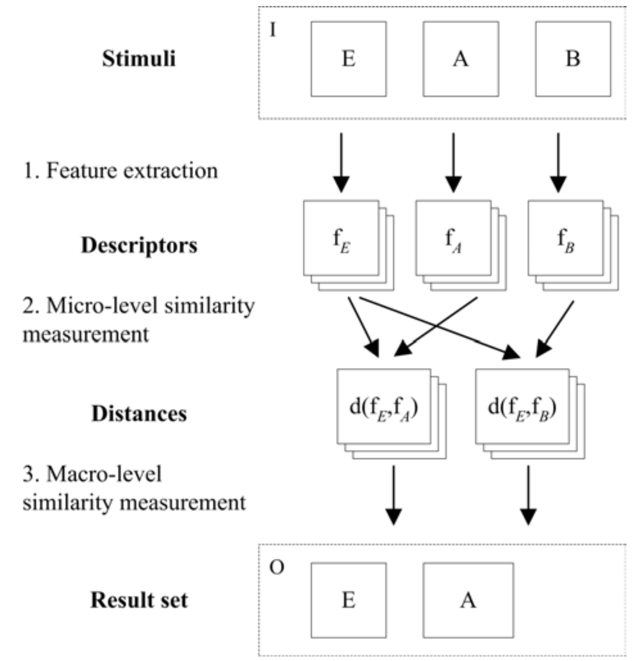

Figure 1: example querying process for three stimuli A, B, E in the input data set *I* and three features (*f* and invisible: *g*, *h*). Stimulus E is the query example. The result set *O* consists of two elements: the query example and stimulus A.

A merging algorithm *m()* for macro-level similarity measurement can be defined as follows:

$$m(I) = O$$

where *I* is the set of input objects (described by their dissimilarity values d_i for all *F* features) and *O* is the result set. *I* has *n* elements.

$$I = \begin{pmatrix} d_1^1 & d_2^1 & .. & d_F^1 \\ d_1^2 & d_2^2 & ... & d_F^2 \\ ... & ... & ... & ... \\ d_1^n & d_2^n & ... & d_F^n \end{pmatrix}$$

$$O = \left[\left(r_1^A, \quad r_2^B, \quad ... \quad r_m^X\right)\right]^T$$

Here, r_i is the *i*-th of m elements in the result set. Index *A* describes that it represents element A of *I*. *i* is the rank of r_i. We propose that each implementation of *m()* has to fulfill the following four merging conditions:

1. Minimality.

$$m(I) = m(I + s(I))$$

for each subset *s(I)* of *I*. That is, the result set has to be independent from duplicates in *I*.

2. Non-discriminating.

$$m(I) = m(p(I))$$

for each permutation *p(I)* of *I*. The result set must be independent of the order of *I*.

3. Ranking condition.

$$i \neq j \Leftrightarrow r_i^X \neq r_j^Y \quad for\, each \;\; i, j \in N_{1,m} \;\; and \;\; r_i^X, r_j^Y \in O$$

where *i* and *j* are the ranks of the result set elements r_i^X and r_j^Y (representing stimuli X and Y) and the result set *O* has *m* elements. This means that *m()* must produce a ranked result set. It must derive at least a partial similarity order (objects with equal similarity may be ranked arbitrary).

4. Linearity.

$$m(I_1 + I_2) = m(I_1) + m(I_2)$$

for all input object sets I_1 and I_2. That is, *m()* should produce the same result set for each partition (I_1, I_2) of *I*.

Valuable similarity information can get lost in the merging step. These conditions should prevent the CBIR system developer from implementing absolute inappropriate merging algorithms. Part of the VizIR project will be the development of new macro-similarity measurement methods that fulfill these conditions. With these methods and the algorithms below we will try to falsify the proposed conditions in human-based evaluations in order to prove their validness.

5. ANALYSIS OF THE LINEAR WEIGHTED MERGING APPROACH

A macro-level similarity measurement algorithm based on Linear Weighted Merging (LWM, Section 2) could look like this:

1. Calculate the position value for each element of *I*.
2. Set *O* as the *n* elements of *I* with the lowest position values. *n* is a parameter provided by the user or the CBIR system.
3. Rank the elements of *O* by the position values. The order of objects with equal position value may be arbitrary.

This algorithm is implemented in QBIC and Virage. If we evaluate this algorithm by our proposed conditions we get the following result:

- LWM does not fulfill the minimality condition. If we set $I_{new} = I_{old} + I_{old}$ then the result set O_{new} contains only halve of the objects of O_{old} and each object twice. This is just a minor problem. We can introduce a new first step in our algorithm: "1. Eliminate all duplicate rows from *I*". Then, LWM fulfils condition 1.
- It fulfills the second and third condition: it is non-discriminating and generates a partial order and a ranked result set.
- LWM does not meet the linearity condition. This is obvious, because for an arbitrary partition (I_1, I_2) both $m(I_1)$ and $m(I_2)$ would produce result sets with *N* elements – no matter if the objects in these result sets are similar or not. This can not be corrected by a new rule. It is a structural problem of LWM. Even if we would allow that $m(I_1)$ and $m(I_2)$ may produce result sets with *n/2* elements, condition 4 would only be

fulfilled for input data sets I_1 and I_2 with half the similar images of I each.

Because LWM does not fulfill the merging conditions and because of our experiences from earlier work, we conclude that LWM is not a suitable algorithm for macro-level similarity measurement. In the next section we will outline the algorithm that will be used in VizIR.

6. QUERYING IMPLEMENTATION IN VIZIR

In our earlier work we have developed a querying paradigm that is based on the Boolean Retrieval Model (see Section 2) but uses a reduced set of logical operators. We call it the Query Model approach. A Query Model consists of a set of layers and each layer of a feature extraction function, a threshold for the maximum distance of two objects and a weight for the importance of the layer. All layers are combined by *AND*. This means that each layer is an information filter, which sorts out all objects from the input data set taken over from the preceding layer that do not have a distance smaller than the threshold (if the threshold is greater or equal 0) or bigger than the threshold (if the threshold is smaller than 0, logical *NOT*). No logical *OR* can be defined in a Query Model. The effect of the *OR* operator can be achieved better by running parallel independent queries. This is more transparent for the user. The querying process consists of the following steps:

1. Apply the first Query Model layer on I. This is done with function $v_1(I) = O_1$. $v_1()$ is an implementation of the first Query Model layer as described in Section 2.
2. Apply the second layer on O_1 using function $v_2()$. $v_2(O_1) = O_2$.
3. Repeat step 2 for all other layers.

If we apply our proposed conditions for macro-level similarity measurement on this algorithm, we get the following result:

- It fulfills the first and second condition: duplicates displace no other objects from the result set O and O is independent from the order of I.
- It does not fulfill condition 3, because it does not rank O. This can be repaired by extending the algorithm with a new final step: “Use the layers weights and Linear Weighted Merging to derive position values and rank the objects in the result set by these position values”.
- It fulfills the linearity condition. Because of the always *AND*-connected layers the result set for each partition (I_1, I_2) is equal to the result set of I_1+I_2.

The Query Model approach fulfills all four conditions. From this result and earlier experiments we are convinced that the Query Model approach is an ideal solution for similarity measurement in CBIR systems. Therefore we will implement a Query Model based querying engine in the VizIR framework.

In addition, the Query Model approach has a nice side-effect on query execution time. Because of using only the logical *AND* to connect layers, the result set of a query is independent from the order of the layers. An algorithm that sorts the layers in a way that those, which sort out most objects and/or use the fastest distance functions, are used first in the querying process, would lead to significant query acceleration. In [2] we have presented the design and implementation of such an algorithm. It reduces the average query execution time in our test environment by 66% (in comparison to a QBIC system with the same feature classes and distance functions).

7. CONCLUSION

In this paper we have presented a general view on the CBIR querying process, pointed out related work in the field of similarity measurement and proposed a set of rules for similarity measurement on the macro-level. Then we have investigated two approaches for macro-level similarity measurement: the widely applied Linear Weighted Merging method and our Query Model approach that is based on the Boolean Retrieval Model. From the results we draw the conclusion to implement the Query Model approach in our Visual Information Retrieval framework.

Finally, we would like to invite interested research institutions to join the discussion and participate in the design and implementation of the open VizIR framework.

8. REFERENCES

[1] J. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. Jain, and C. Shu, “The Virage image search engine: An open framework for image management”, Proc. SPIE Storage and Retrieval for Image and Video Databases IV, San Jose CA, USA, pp. 76-87, February 1996.

[2] C. Breiteneder, and H. Eidenberger, “Performance-optimized feature ordering for Content-based Image Retrieval”, Proc. European Signal Processing Conf., Tampere, Finland, 2000.

[3] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, “Query by Image and Video Content: The QBIC System”, *IEEE Computer*, vol. 28, no. 9, pp. 23-32, 1995.

[4] M. Ortega, R. Yong, K. Chakrabarti, K. Porkaew, S. Mehrotra, and T.S. Huang, “Supporting Ranked Boolean Similarity Queries in MARS”, *IEEE Transactions on Knowledge and Data Engineering*, vol. 10, no. 6, pp. 905-925, November 1998.

[5] A. Pentland, R.W. Picard, and S. Sclaroff, “Photobook: Content-Based Manipulation of Image Databases”, *SPIE Storage and Retrieval Image and Video Databases*, no. 2185, pp. 34-47, 1994.

[6] S. Santini, and R. Jain, “Similarity Measures”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 9, pp. 871-883, September 1999.

[7] G. Sheikholeslami, W. Chang, and A. Zhang, “Semantic Clustering and Querying on Heterogeneous Features for Visual Data”, Proc. ACM Multimedia, Bristol, UK, pp. 3-12, 1998.

[8] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, “Content-Based Image Retrieval at the End of the Early Years”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349-1380, December 2000.

[9] J.R. Smith, and S.F. Chang, “VisualSEEk: a fully automated content-based image query system”, Proc. ACM Multimedia, Boston MA, USA, pp. 87-98, 1996.