A web-based evaluation system for CBIR

Henning Müller, Wolfgang Müller,
Stephane Marchand-Maillet, Thierry Pun
Vision Group, University of Geneva, Switzerland
Henning.Mueller@cui.unige.ch

David Squire
Computer Science and Software Engineering
Monash University, Melbourne, Australia
David.Squire@csse.monash.edu.au

ABSTRACT

This papers describes a benchmark test for content-based image retrieval systems (CBIRSs) with the query by example (QBE) query paradigm. This benchmark is accessible via the Internet and thus allows to evaluate any CBIRS which is compliant with the Multimedia Retrieval Markup Language (MRML) for query formulation and result transmission. Thus it allows a quick and easy comparison between different features and algorithms for CBIRSs. The benchmark is based on a standardized communication protocol to do the communication between benchmark server and benchmarked system and it uses a freely downloadable image database (DB) to make the results reproducible. A CBIR system that uses MRML and other components to develop MRML-based applications can be downloaded free of charge as well.

The evaluation is based on several queries and known relevance sets for these queries. Several answer sets for one query image are possible if judgments of several users exist, thus almost any sort of judgment can be incorporated into the system. The final results are averaged over all queries.

The evaluation of several steps of relevance feedback (RF) based on the collected relevance judgments is also included into the benchmark. The performance of RF is often regarded to be even more important than the performance in the first query step because only with RF the adaptation of the system to the users subjective goal can be measured. For the evaluation of a system with RF, the same evaluation measures are used as for the first query step.

1. INTRODUCTION

Proper evaluation has always been regarded as very important for content-based image retrieval (CBIR). Despite this importance, most systems were evaluated with a variety of non-standardized performance measures and with image DBs not accessible to reproduce and compare the results. Many systems were only evaluated with one example result, which gives no objective impression of the quality of a system. Evaluation was a widely neglected topic in CBIR.

In text retrieval (TR), a closely related field, standardized performance comparisons were proposed as early as the 60s with SMART in 1961 [16] and the Cranfield test in 1966 [1]. With the Text REtrieval Conference (TREC) ([19], `http://trec.nist.gov/`) a clearly defined and accepted benchmark started in 1992 and has been done every year since. This helped the research field because good techniques could be distinguished from poor techniques.

MIRA (`http://www.dcs.gla.ac.uk/mira/`, Evaluation Frameworks for Interactive Multimedia Information Retrieval Applications) was the first iniative to take a more formal approach to evaluate Multimedia Retrieval systems. Several conferences and workshops were held within this framework.

In 1997 Narasimhalu [14] gives a formal comparison of different sorts of CBIRSs and how the systems can be evaluated based on users giving ranked relevance sets for a number of query images. Concrete performance measures or image DBs to use were not proposed and there is also no example of an evaluation given.

More publications on the subject appeared in recent years. Starting from discussions at conferences such as Visual99 (LNCS 1614) and SPIE Photonics West 2000 the need for a standardized performance evaluation, standardized image DBs, and especially an event to compare systems, became apparent. In 1999 Dimai [3] described a rank-based measure to compare two different feature sets or CBIRSs to overcome the shortcomings of precision and recall. To compare two systems this might work, but for a benchmark many systems need to be compared. It is also important not to compare the systems based on a single performance measure, but on several measures because for different application areas different characteristics are important. Koskela et al. [7] describe performance measures to quantify how close together clusters of images are in feature space based on their ranks. This can be used to compare different features and techniques. Leung [8] gives a detailed proposal for a benchmark with stating performance measures and the approximate sizes of the DBs. He proposes to have a DB of roughly 1000 images for a start and have a number of categories with not more than $15 - 20$ relevant images for a query. An example evaluation with the measures is not given in the article. In [10, 11] an approach similar to TREC is used for CBIRS evaluation. Measures are proposed and an automatic benchmark is implemented based on these measures, with an example evaluation based on one CBIR system. None of these papers discusses the hard question of how to obtain a large, freely available image DB and relevance judgments.

By far the most promising approach to a CBIR benchmark is the Benchathlon (`http://www.benchathlon.net/`). It started from discussions at SPIE Photonics West 2000 and had its first prototypical system at Photonics West 2001. The techniques of the benchmark are described in [4]. For the conference in 2002 a larger DB and a more sophisticated benchmark is planned. Several researchers from different fields are currently working on this benchmark.

Most of the proposals are evaluating the performance of Query by Example (QBE) systems. For browsing systems (or target search) measures are proposed in [2], where a real user has to interact with a DB and an automatic approach in [12], where extensive annotation is used to simulate the users behavior in selecting images.

To compare systems relying on different paradigms such as QBE, image browsing or search by keyword is a very hard task if feasible at all. This aspect is not addressed here.

This paper describes a solution based on the measures and techniques used by TREC for TR, but transferred to CBIR. The system can use different kinds of relevance judgments and several image DBs to allow a maximum flexibility with respect to the images and judgments that are available.

2. THE TECHNOLOGICAL BASIS

For a fully automatic evaluation of CBIRSs there has to be a common access method, so that the benchmark server can automatically perform all the queries, including relevance feedback (RF) queries, and receive the results for a performance evaluation. Other problems for any evaluation are freely available image DBs to make the obtained results comparable and reproducible. The hardest and most work-consuming task is most likely to obtain ground truth data for the images, especially for a large image DB.

2.1 MRML

The Multimedia Retrieval Markup Language (MRML, `http://mrml.net/`) is an XML-based communication protocol for CBIR, which was developed to separate the query interface from the query engine. It was developed for QBE and thus contains tags for query by positive and negative examples. A technical description can be found in [13].

The client can open a session on the server, and configure it according to the needs of its user (interactive client) or its own needs (eg. benchmark test).

A query consists of a list of images and the corresponding relevance levels, assigned by the user. In the following example, the user has marked two images: `1.jpg` positive and `2.jpg` negative. All images are referred to by URLs.

```
<mrml session-id="1" transaction-id="44">
<query-step session-id="1"
 resultsize="30"
 <user-relevance-list>
  <user-relevance-element
   image-location="http://viper.unige.ch/1.jpg"
   user-relevance="1"/>
  <user-relevance-element
   image-location="http://viper.unige.ch/2.jpg"
   user-relevance="-1"/>
 </user-relevance-list>
</query-step> </mrml>
```

The server will return the result as a list of image URLs, ordered by their relevance to the query.

2.2 Image DBs

A general approach to a CBIR benchmark has to have a maximum flexibility with respect to the image DBs used. Because of this, we allow the benchmark to use any kind of DB where the images can be accessed by URL.

Many existing CBIRSs use the Corel image collections for an evaluation that contain groups of 100 images each with roughly the same subject. Still, the images are rather expensive and copyrighted and the choice of groups determines the difficulty of the query task. The images of MPEG-7 [9] are as well copyrighted and might not be used in publications or on the Internet which makes them unusable for a performance comparison between systems. Another possibility is the image collection of the Department of Water Resources (DWR, `http://elib.cs.berkeley.edu/photos/tarlist.txt`) in California that is available without charge for non-commercial use from UC Berkeley . This DB is relatively large (more than $25,000$ images) but has only a limited number of different contents. No relevance judgments are currently available for this DB.

We decided to use the DB of the University of Washington (UW), Seattle (`http://www.cs.washington.edu/research/imagedatabase/groundtruth/`) for the evaluation in this paper because it offers ground truth in form of image clusters and annotations for most of the images. It is available free of charge and without copyright. Unfortunately it is still very small with only 922 images in 14 image clusters, but we hope to enlarge the set with the help of other research groups. We also tested the benchmark on the DB of the Télévision Suisse Romande (TSR) where user judgments of five users are available but this DB is copyrighted.

We plan to include other DBs that we can get with relevance judgments. A standard DB as the Benchathlon DB will be very useful for the context of a web-based evaluation.

2.3 Relevance Judgments

One of the hardest tasks for the evaluation of CBIRSs is the process of obtaining relevance judgments. Often image DBs contain clusters of images with the same objects ("cars", "airplanes") like the Corel collection or images of regions ("mountains", "cities") like the DB of the UW. In this case the clusters can be regarded as ground truth and one image of the cluster can be taken as example image for a QBE query. Unfortunately an image from a cluster has often more similarities with images from other clusters than with the same cluster. Visual similarity within a cluster can vary in a large span. For these reasons predefined clusters are not always a very good choice as relevance judgments.

These fixed image clusters also neglect the subjectivity of the users. With the same query image users can look for a completely different answer set [18]. To model this user subjectivity, real user tests have to be performed with several users as in [17]. There is also the possibility to use textual annotations of images for the generation of groundtruth. More about the classification of images can be read in [6].

The evaluation in this paper is based on the clusters of the image DB of the UW, because this DB is freely available and thus the results are reproducible. Tests have also been performed with real user judgments on DBs of the TSR. The TSR DB can be evaluated via the web interface.

2.4 Performance measures

In this paper we mainly use the performance measures described in [11] to have a set of measures similar to those used in TREC. These measures can be used for all image DBs and also the different kinds of relevance judgments mentioned. The measures are:

- $Rank_1$, $\bar{Rank}$ and $\widetilde{Rank}$: rank at which first relevant image is retrieved, average rank and normalized average rank of relevant images (see Eq. 1);
- $P(20)$, $P(50)$ and $P(N_R)$: *precision* after 20, 50 and N_R images are retrieved;
- $R_P(.5)$ and $R(100)$: *recall* at *precision* .5 and after 100 images are retrieved;
- PR graph, DB size and execution time for a query.

A simple average rank is difficult to interpret, since it depends on both the collection size N and the number of relevant images N_R for a given query. Consequently, we normalize the average rank by these numbers and propose the *normalized average rank*, $\widetilde{Rank}$:

$$\widetilde{Rank} = \frac{1}{NN_R}\left(\sum_{i=1}^{N_R} R_i - \frac{N_R(N_R-1)}{2}\right) \qquad (1)$$

where R_i is the rank at which the ith relevant image is retrieved. This measure is 0 for perfect , 0.5 for random performance, and approaches 1 as performance worsens.

We use several measures based on precision and recall despite criticism on precision and recall already in the 60s [16] and as well in [3, 7] for the use in CBIR. Precision and recall, especially in form of the PR graph and in form of precision or recall at important cutoff point, are still the standard in TR and are easy to interpret.

For a user who is in general looking at $20-50$ images on screen it is very important how many relevant images he can actually see as a result. For a more user-centered evaluation it does not really make a difference whether a relevant image is retrieved at position 1000 or 2000, whereas position 50 or 51 can make a larger difference if the user cannot see the image in the response set displayed on screen.

We are also integrating the normalized average rank measure proposed in [4, 15], that basically proposes a penalization for images that are not retrieved at all.

3. THE WEB-BASED BENCHMARK

A description of the web-based benchmark is available from `http://viper.unige.ch/evaluation/`. This page contains the prerequisites for the execution of the benchmark and links to a number of benchmark resources. An example system using MRML can be downloaded at `http://www.gnu.org/software/gift/`.

3.1 Overview

Figure 1 shows the general structure of the benchmark. The communication between the benchmark server and the benchmarked systems is done in MRML. The benchmarked systems basically only need to know the URLs of the images in the DB. The performance measures are openly visible as well. The ground truth data for the images and even the images chosen as query images should not be known by the benchmarked systems as they can try to cheat when this information is available. If a system knows the image classes, it can of course always return a perfect response. Normally the phase of getting the ground truth should be done after all the systems have returned the results.

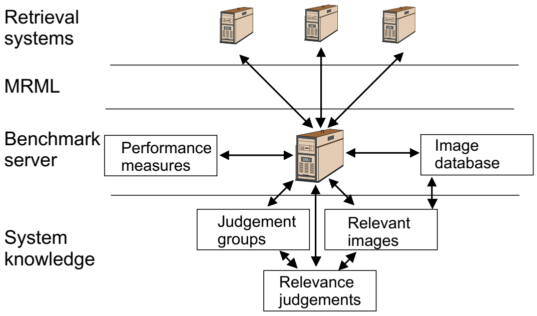
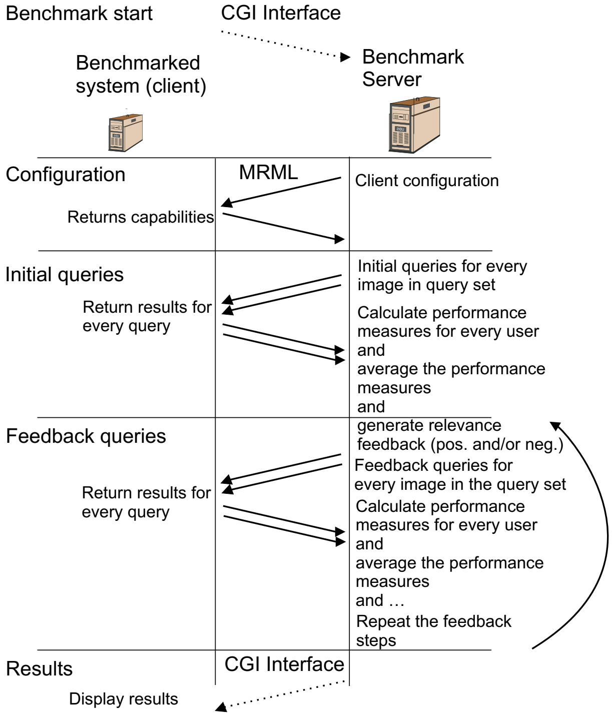


Figure 1: Structure of the automated benchmark.

For the web-based benchmark the image groups are for now not hidden. A system could thus cheat and get good results. However, this web-based benchmark is so far a research tool and not an official benchmark, so there is no need to hide the ground truth data.

3.2 Communication framework

We can see in Figure 2 that only the first step in the communication, the configuration of the benchmark and the last step of the communication are not done in MRML. The configuration is done via a CGI interface, and the results are displayed on a web browser after the execution of the benchmark. With the first step of communication in MRML the benchmark server contacts the system to benchmark and verifies that the chosen DB is available and that the system speaks MRML. If this is the case, all queries are done as single image queries (QBE). After the first query step,



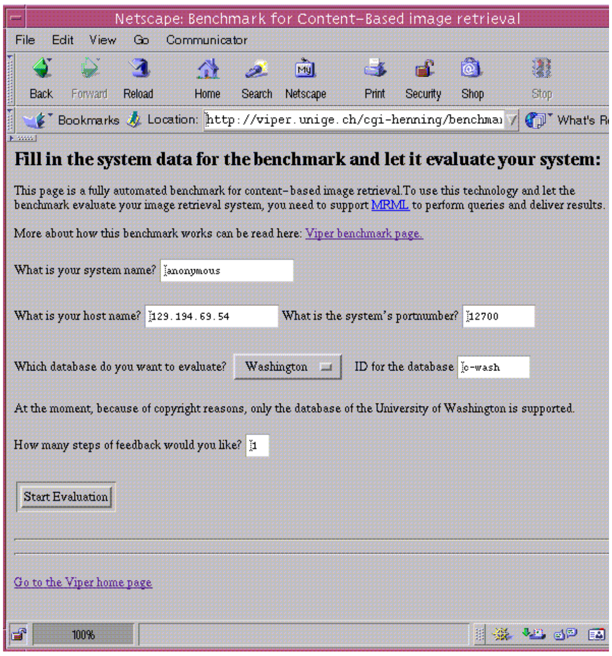
Benchmark start
CGI Interface
Benchmarked system (client)
Benchmark Server
Configuration
MRML
Client configuration
Returns capabilities
Initial queries
Initial queries for every image in query set
Return results for every query
Calculate performance measures for every user and average the performance measures and generate relevance feedback (pos. and/or neg.)
Feedback queries
Feedback queries for every image in the query set
Return results for every query
Calculate performance measures for every user and average the performance measures and ... Repeat the feedback steps
Results
CGI Interface
Display results

Figure 2: Communication steps for the benchmark.

the performance measures are calculated for the available relevance judgments and are averaged over all the relevance judgments and queries. Now positive and negative RF is generated for every judgment group (respectively every user) and every query. Based on this generated RF, a new set of queries is executed and again the performance measures are calculated based on the relevance judgments, and the performance measures are averaged in the end. This step can be repeated as often as necessary. Finally, all the averaged performance measures are displayed on the web browser.

3.3 Configuring the benchmark

The CGI Interface shown in Figure 3 allows the user to enter a number of parameters that the system needs to execute the benchmark. The *system name* is only an identification of the benchmarked system to the server, it can be left at anonymous if the developers want their system to stay unknown. Important for the communication are the *host name* and the *port number* of the system to benchmark. These two parameters are absolutely needed to start the MRML communication on this socket. The choice of a *DB* determines the queries and the relevance judgments the web-based benchmark will use. The *DB ID* is important for the benchmark server to chose the DB via MRML. The number of *RF steps* finally determines the number of query steps that are done with the system. The first step is in this context the step with only one query image and no RF.



http://viper.unige.ch/cgi-henning/benchma

Fill in the system data for the benchmark and let it evaluate your system:

This page is a fully automated benchmark for content–based image retrieval.To use this technology and let the benchmark evaluate your image retrieval system, you need to support MRML to perform queries and deliver results.

More about how this benchmark works can be read here: Viper benchmark page.

What is your system name? anonymous

What is your host name? 129.194.69.54 What is the system's portnumber? 12700

Which database do you want to evaluate? Washington ID for the database c-wash

At the moment, because of copyright reasons, only the database of the University of Washington is supported.

How many steps of feedback would you like? 1

Go to the Viper home page

Figure 3: A screenshot of the web-based benchmark

4. RESULTS OF AN EXAMPLE RUN

To demonstrate the benchmark, we use the GIFT (GNU Image Finding Tool) as a client with an MRML interface. The evaluation with this client was executed for the DB of the UW and for the DB of the TSR as well. As an example we used four steps of RF for the evaluation, to show the possibility to evaluate RF. The results are put into tables for readability. Normally the results are directly shown on screen.

4.1 Image DB University of Washington

The DB of the UW consists of 922 images that are in 14 different categories, normally geographical areas. We always took the first image of a group as a query image and all the images of a group as the relevance set, no matter how visually similar or different they are. The results in Table 1 show that the first two steps of RF strongly improve the results. Steps three and four only bring minor improvements. The rank of the first relevant image shows that from the initial query there were very good results for every query. The $P(20)$ shows that after 4 steps of RF there are an average of more than 18 relevant images in the top 20 which also shows that this is a relatively easy DB for similarity queries.

Measure	no RF	RF 1	RF 2	RF 3	RF 4
N_R	65.14	65.14	65.14	65.14	65.14
t	1.23 s.	2.18 s.	2.49 s.	2.62 s.	2.70 s.
$Rank_1$	1.5	1	1	1	1
$R(P(.5))$	.3798	.5520	.6718	.6594	.7049
$Rank$	176.44	152.28	116.13	107.04	104.37
$\widetilde{Rank}$	.1583	.1318	.0921	.0821	.0793
$P(20)$	.5392	.7357	.8642	.8892	.9107
$P(50)$	.4057	.5271	.6085	.6328	.6257
$P(N_R)$	.3883	.5256	.6138	.6640	.6553
$R(100)$	.4839	.6070	.6924	.7279	.7208

Table 1: Results for GIFT with the Washington DB

4.2 Image DB of the TSR

The image DB of the TSR consists of 500 images with only few clusters of really similar images. Five users gave relevance judgments for ten query images. They had to find all images that they regarded as similar to the query images, with no strict policy given for similarity. Table 2 shows a different result from Table 1. The first query step brings a significant improvement, but afterwards the results do not improve very much. The fact that the first relevant image found does not get to 1 means that for at least one query there was no relevant image in the first $n = 20$ result images. The fact that the precision at 50 images does not improve at all means that the RF might have improved the ordering of the relevant images within the first 50, but not many new images are shown in there. As there are only an average of 10

Measure	no RF	RF 1	RF 2	RF 3	RF 4
N_R	10.56	10.56	10.56	10.56	10.56
t	1.83 s	2.44 s	2.65 s	2.77 s	2.92 s
$Rank_1$	14.6	9.96	10.16	9.94	10.1
$R(P(.5))$	.3411	.447	.4575	.4334	.4399
$Rank$	58.14	54.48	53.93	52.64	51.14
$\widetilde{Rank}$	.1065	.0992	.098	.0955	.0925
$P(20)$	.263	.282	.293	.298	.302
$P(50)$	.1572	.1488	.1496	.1528	.1548
$P(N_R)$	.4789	.6068	.6305	.6441	0.65
$R(100)$	.7901	.7988	.8031	.8044	.8123

Table 2: Results for GIFT with the TSR DB

relevant images, the maximum precision at this point can be 20 %. It shows that the queries on this DB are significantly more difficult than on the DB of the UW.

5. CONCLUSION AND FUTURE WORK

This paper presents a working benchmark for CBIR that can be configured via the World Wide Web and displays the evaluation results on any web browser. The DBs used in this paper are just an example and we are aware that for a proper CBIRS evaluation larger, free DBs are necessary which causes problems for the generation of proper relevance judgments. We hope that the Benchathlon effort will provide this. With a large DB including good ground truth, the web-based benchmark can be a very helpful tool for system developers to test the system performance for new features or new access methods on the fly.

A regular benchmark event like the Benchathlon can of course not be replaced by such a web accessible benchmark, because it is necessary to compare a number of systems. The web-based benchmark is more meant to complement the Benchathlon and give system developers the possibility to try out their system from time to time to be able to identify performance differences. It can also be used to test the MRML interface of a system in an automated way.

For the future we plan to include more performance measures to also be able to compare these measures with respect to their information content. Graphical evaluation methods, like precision/recall graphs, are also being developed for their use on a WWW platform. For a final benchmark there should be a small number of performance measures and, even more important, all measures should contain different information. We would be happy to include any DB we can get relevance judgments and a URL list for.

6. ACKNOWLEDGMENTS

This work was supported by the Swiss National Foundation for Scientific Research (grant no. 2000-052426.97).

7. REFERENCES

[1] C. W. Cleverdon, L. Mills, and M. Keen. Factors determining the performance of indexing systems. Technical report, Cranfield Project, Cranfield, 1966.

[2] I. J. Cox, M. L. Miller, S. M. Omohundro, and P. N. Yianilos. Target testing and the `PicHunter` Bayesian multimedia retrieval system. In *Advances in Digital Libraries (ADL'96)*, pages 66–75, Library of Congress, Washington, D. C., May 13–15 1996.

[3] A. Dimai. Assessment of effectiveness of content-based image retrieval systems. In Huijsmans and Smeulders [5], pages 525–532.

[4] N. J. Gunther and G. Beretta. A benchmark for image retrieval using distributed systems over the internet: Birds-i. Technical report, HP Labs, Palo Alto, Technical Report HPL-2000-162, San Jose, 2001.

[5] D. P. Huijsmans and A. W. M. Smeulders, editors. *Third International Conference On Visual Information Systems (VISUAL'99)*, number 1614 in Lecture Notes in Computer Science, Amsterdam, The Netherlands, June 2–4 1999. Springer-Verlag.

[6] C. Jörgensen. Classifying images: Criteria for grouping as revealed in a sorting task. In *Proceedings of the 6th ASIS SIG/CR Classification research Workshop*, pages 65–78, Chicago, Il, USA, October 1995.

[7] M. Kokela, J. Laaksonen, S. Laakso, and E. Oja. Evaluating the performance of content-based image retrieval systems. In *International Conference on Visual Information Systems (Visual 2000)*, Lyon, France, November 2–4 2000.

[8] C. Leung and H. Ip. Benchmarking for content-based visual information search. In *International Conference on Visual Information Systems (Visual 2000)*, Lyon, France, November 2–4 2000.

[9] MPEG Requirements Group. MPEG-7: Context and objectives (version 10 Atlantic City). Doc. ISO/IEC JTC1/SC29/WG11, International Organisation for Standardisation, October 1998.

[10] H. Müller, W. Müller, S. Marchand-Maillet, D. M. Squire, and T. Pun. Automatic benchmarking in content-based image retrieval. In *International conference on Multimedia and Exhibition(ICME 2001)*, Tokyo, Japan, October 2001. (to appear).

[11] H. Müller, W. Müller, D. M. Squire, S. Marchand-Maillet, and T. Pun. Performance evaluation in content-based image retrieval: Overview and proposals. *Pattern Recognition Letters*, 2001.

[12] W. Müller, S. Marchand-Maillet, H. Müller, and T. Pun. Towards a fair benchmark for image browsers. In *SPIE Photonics East, Voice, Video, and Data Communications*, Boston, MA, USA, nov 5–8 2000.

[13] W. Müller, H. Müller, S. Marchand-Maillet, T. Pun, D. M. Squire, Z. Pečenović, C. Giess, and A. P. de Vries. MRML: A communication protocol for content-based image retrieval. In *International Conference on Visual Information Systems (Visual 2000)*, Lyon, France, November 2–4 2000.

[14] A. D. Narasimhalu, M. S. Kankanhalli, and J. Wu. Benchmarking multimedia databases. *Multimedia Tools and Applications*, 4:333–356, 1997.

[15] P. S. Salembier and B. S. Manjunath. Audiovisual content description and retrieval: Tools and mpeg-7 standardization techniques. In *IEEE Internation Conference on Image Processing (ICIP 2000)*, Vancouver, BC, Canada, December 2000.

[16] G. Salton. *The SMART Retrieval System, Experiments in Automatic Document Processing.* Prentice Hall, Englewood Cliffs, New Jersey, USA, 1971.

[17] D. M. Squire, W. Müller, and H. Müller. Relevance feedback and term weighting schemes for content-based image retrieval. In Huijsmans and Smeulders [5], pages 549–556.

[18] D. M. Squire and T. Pun. A comparison of human and machine assessments of image similarity for the organization of image databases. In M. Frydrych, J. Parkkinen, and A. Visa, editors, *The 10th Scandinavian Conference on Image Analysis (SCIA'97)*, pages 51–58, Lappeenranta, Finland, June 1997. Pattern Recognition Society of Finland.

[19] E. M. Voorhees and D. Harmann. Overview of the seventh text retrieval conference (TREC-7). In *The Seventh Text Retrieval Conference*, pages 1–23, Gaithersburg, MD, USA, November 1998.