A framework for benchmarking in CBIR *

Henning Müller, Wolfgang Müller, Stephane Marchand-Maillet, Thierry Pun
Vision Group, University of Geneva, Switzerland

David McG. Squire
CSSE, Monash University, Melbourne, Australia

Abstract. Content-based image retrieval (CBIR) has been a very active research area for more than ten years. In the last few years the number of publications and retrieval systems produced has become larger and larger. Despite this, there is still no agreed objective way in which to compare the performance of any two of these systems. This fact is blocking the further development of the field since good or promising techniques can not be identified objectively, and the potential commercial success of CBIR systems is hindered because it is hard to establish the quality of an application.

We are thus in the position in which other research areas, such as text retrieval or the database systems, found themselves several years ago. To have serious applications, as well as commercial success, objective proof of system quality is needed: in text retrieval the TREC benchmark is a widely accepted performance measure; in the transaction processing field for databases it is the TPC benchmark that has wide support.

This paper describes a framework that enables the creation of a benchmark for CBIR. Parts of this framework have already been developed and systems can be evaluated against a small, freely-available database via a web interface. Much work remains to be done with respect to making available large, diverse image databases and obtaining relevance judgments for those large databases. We also need to establish an independent body, accepted by the entire community, that would organize a benchmarking event, give out official results and update the benchmark regularly. The *Benchathlon* could get this role if it manages to gain the confidence in the field. This should also prevent the negative effects, e.g. "benchmarketing", experienced with other benchmarks, such as the TPC predecessors.

This paper sets out our ideas for an open framework for performance evaluation. We hope to stimulate discussion on evaluation in image retrieval so that systems can be compared on the same grounds. We also identify query paradigms beyond query by example (QBE) that may be integrated into a benchmarking framework, and we give examples of application-based benchmarking areas.

Keywords: evaluation, content-based image retrieval, benchmarking, Benchathlon, TREC

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

Keywords: evaluation, content-based image retrieval, benchmarking, Benchathlon, TREC

1. Introduction

Performance evaluation was long a neglected topic in content-based image retrieval (CBIR). This changed a few years ago as more and more CBIR systems were developed and the difficulty of comparing their performances on an objective basis became apparent.

In text retrieval, a mature and closely-related field, standardized performance tests have been performed since the 1960s, with SMART in 1961 (Salton, 1971) and the Cranfield tests in 1962 (Cleverdon, 1962) and 1966 (Cleverdon et al., 1966). Some important results were gained from these standardized tests. They showed, for example, that automatic indexing performed comparably to manual indexing (Cleverdon et al., 1966). With the inauguration of the Text REtrieval Conference (TREC) (`http://trec.nist.gov/`) in 1992 a clearly defined and accepted benchmark was established and has been repeated every year since (Harman, 1992, Vorhees and Harmann, 1998).

The TPC benchmark (`http://www.tpc.org/`) similarly brought a standard to the field of transaction processing, with the first results being published in 1990. For both benchmarks discussions on how to measure the performance of systems went on for years before a widely accepted and successful benchmark was established. The key to the success of these benchmarks rests in a strong and independent governing body that has the support of all the various groups. Also, both text retrieval and transaction processing are commercially successful fields and thus more funding is available for benchmark development than in a purely research-based field.

Another governing body for performance evaluations is SPEC (`http://www.spec.org/`, the Standard Performance Evaluation Corporation).

MIRA (`http://www.dcs.gla.ac.uk/mira/`, Evaluation Frameworks for Interactive Multimedia Information Retrieval Applications, 1995) was the first project to take a more formal approach to the evaluation of Multimedia Retrieval systems. Several conferences and workshops were held within this framework.

In 1997, Narasimhalu (Narasimhalu et al., 1997) gave a formal comparison of different sorts of CBIR systems (CBIRSs) and how the

systems could 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 no example evaluation was given.

In 1998, John R. Smith (Smith, 1998) highlighted the necessity of a benchmark in CBIR and proposed the use of TREC as a model. No example evaluation was done. In 1999 Dimai (Dimai, 1999) described a rank-based measure for comparing two different feature sets or CBIRSs to overcome the shortcomings of precision and recall. For a comparison of two systems this might work, but in a benchmark framework many systems need to be compared. It is also important not to compare the systems based only on a single performance measure, but on several measures. This is because different characteristics are important for different application areas and different users might also look for varying performance characteristics. Koskela et al. (Koskela et al., 2000) described performance measures to quantify how close together clusters of images are in feature space based on their retrieval ranks. This only works well when the images can clearly be classified into disjoint groups.

Leung (Leung and Ip, 2000) gave a detailed proposal for a benchmark, stating performance measures and the approximate sizes of the DBs. He proposed an initial DB of roughly 1000 images and a number of categories with not more than 15–20 relevant images for a query. An example evaluation with the measures was not given in the article. In (Müller et al., 2001a, Müller et al., 2001c) an approach similar to TREC was used for CBIRS evaluation. Measures were proposed and an automatic benchmark implemented based on these measures, with an example evaluation based on one CBIR system. A web interface to this benchmark was added in (Müller et al., 2001b)

None of these papers discussed the difficult and important question of how to obtain a large, freely available image database and relevance judgments, a question which has been extensively discussed in the text retrieval community (Sparck Jones and van Rijsbergen, 1975).

By far the most promising approach to a CBIR benchmark is the *Benchathlon* (`http://www.benchathlon.net/`). It arose from discussions at the SPIE Photonics West 2000 conference and the first prototype system appeared at Photonics West 2001. The techniques of the benchmark are described in (Gunther and Beretta, 2001). For the conference in 2002, a larger DB and a more sophisticated benchmark is planned. Several researchers from different fields and various nations are currently working on this benchmark.

Besides a comparison of general purpose QBE systems there is the need for a number of different areas of image retrieval to be benchmarked separately. In this paper we identify a number of different fields

with special characteristics to be benchmarked in further benchmark tests.

2. Problems with benchmarking in CBIR

There are many problems in benchmarking for every domain. The first important step is to identify these problems and then find solutions to them. In this section we define the problems and give ideas for initial solutions.

2.1. Image databases

There is as yet no common database used in image retrieval. Such a database would have to be available free of charge, no copyright should hinder its use on the internet and in publications, and it should be sufficiently diverse and complex to satisfy many needs.

Many existing CBIRSs use the *Corel* (`http://www.corel.com/`) image collections for evaluations that contain groups of 100 images, each with roughly the same subject. These images, however, are expensive and copyrighted, and the choice of groups determines the difficulty of the query task. The *MPEG-7* (MPEG Requirements Group, 1998) images are also copyrighted and may not be used in publications or on the Internet, which makes them unsuitable for 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 subjects. No relevance judgments are currently available for this DB. The DB of the *University of Washington* (UW), Seattle (`http://www.cs.washington.edu/research/imagedatabase/groundtruth/`) is available without charge and copyright and is thus a very good candidate for a benchmark. Unfortunately it is still small, with only 922 images in 14 image clusters, but the hope is to enlarge it with the help of other research groups. For texture analysis several databases, such as the *Brodatz textures* and the *VisTex textures* (`http://www-white.media.mit.edu/vismod/imagery/VisionTexture/vistex.html`), exist. These databases contain a few hundred images with sets of different textures. The *Benchathlon* team is creating a database that contains at the moment roughly 3000 unsorted images without copyright and is available free of charge, but no relevance judgments are yet available.

2.2. Relevance judgments

Relevance judgments contain the knowledge about the images in the database. Often image DBs contain *clusters* of images with the same objects ("cars", "airplanes") such as 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 those from the same cluster. For example, a picture of Paris by night has more similarities with other pictures taken at night than with daylight pictures of Paris that might be in the same cluster. Visual similarity within a cluster can vary over a great range. For these reasons, predefined clusters are not always a very good choice as relevance judgments. These fixed image clusters also neglect the subjectivity of users. With the same query image users can look for a completely different answer set (Squire and Pun, 1997). To model this user subjectivity, *real user tests* should be performed with several users as in (Squire et al., 1999). This is very time consuming and becomes harder for larger databases. For large databases, pooling can be used to limit the number of documents at which a person needs to look (Sparck Jones and van Rijsbergen, 1975). Unfortunately this makes precision/recall graphs inexact (i.e. above approximately 50% recall for TREC (Harman, 1992)). TREC only uses one relevance set for each query, whereas in image retrieval several sets are necessary to better model user and task subjectiveness (Mokhtarian et al., 1996, Squire and Pun, 1997), since it has been shown that target sets vary greatly between users and tasks.

A possibility for real ground truth is to use *expert opinions* in restricted domains such as medical image search (Shyu et al., 1999, Dy et al., 1999), where a diagnosis can be regarded as a relevance judgment. The performance of the system can then be compared with the diagnoses. A similar expert opinion can be taken in trademark retrieval (Eakins et al., 1998), but experts also may sometimes disagree.

There is also the possibility of using textual *annotations* of images for the generation of groundtruth. More about the textual classification of images can be read in (Jörgensen, 1995). An annotation tool for images is described in (Pfund and Marchand-Maillet, 2002) and a way to obtain relevance judgments from annotation in (Jörgensen and Jörgensen, 2002). The great advantage of obtaining relevance judgments from annotations is that the existing annotation can be reused when other query images are chosen, or the database is enlarged, though care needs to be taken to model the user subjectivity well. The *Benchathlon* team is

in the process of generating annotations for the creation of relevance judgments.

2.3. Performance measures

There are many performance measures used in image retrieval under varying names (Müller et al., 2001c) but more and more the methods which have been is use for more than 40 years in text retrieval have become the standard: *precision vs. recall* graphs, as they show the performance of systems well and are easy to interpret. Despite much criticism of *precision* and *recall*, in both the fields of text retrieval (Salton, 1971, Borlund and Ingwersen, 1997) and in the image retrieval, (Dimai, 1999, Koskela et al., 2000), they still remain the standard measures as they are easy to understand and interpret. In order to measure the retrieval performance for several domains, a number of performance measures is needed, since different fields have different requirements. Whereas for trademark retrieval a 100% *recall* is extremely important, a media search system for journalists must lean much more towards a high *precision* in the first $n = 20 \ldots 50$ images retrieved (Markkula and Sormunen, 1998).

Other common measures are rank-based measures such as those described in (Gunther and Beretta, 2001) and used in MPEG-7 (Salembier and Manjunath, 2000).

2.4. Access to systems

There is as yet no commonly accepted access method to CBIRS. The only method proposed so far is the Multimedia Retrieval Markup Language (MRML, (Müller et al., 1999)), which has already been used for a benchmark (Müller et al., 2001a). This retrieval language offers some of the same properties as SQL for doing exact queries on databases, but uses the QBE paradigm, as well as supporting the notion of ranked retrieval. Examples of the use of MRML are given in Section 3.2.

TREC, and also TREC for video, receive retrieval results offline before the actual conference. This, however, is infeasible for CBIR, because the search for images is much more user- and task-dependent (Mokhtarian et al., 1996) than is text retrieval and thus relevance feedback (RF) must be an integral part of the evaluation process. The importance of RF evaluation is shown in Section 4.

2.5. Motivating research groups to participate

The most important part of a benchmarking framework is of course to have as many research groups as possible to support the benchmark

and to participate in a benchmarking event. The *Benchathlon* wrote a call for papers and participation to let groups actively participate in the process of creating such a benchmarking event. It is very important to state that such a benchmark should help every participant to identify good and bad parts of a system. It is not supposed to become a contest with a rivalry between the groups, but an event to explain and compare techniques.

3. A framework for benchmarking

A framework for benchmarking has to contain not only an event for benchmarking, where researchers can exchange ideas and compare techniques, but also the possibility to get performance results of a CBIRS regularly and easily accessible. If possible, the benchmarking event and the regular testing should be performed based on the same access technology, so it is possible to try out the technical infrastructure before real tests, with official results being performed at a benchmarking event.

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 the MRML communication protocol. The benchmarked systems basically only need to know the URLs of the images in the DB. The performance measures are openly visible as well because they should be varied enough so they capture the entire performance of a system and they cannot be manipulated as a single measure could possibly be.

The ground truth data for the images and even the images chosen as query images should not be known by the benchmarked systems as a system 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, although this might not even be a problem. TREC retrieves all results offline and nevertheless it is not thought to be cheated upon, because a possible customer could of course test the system on the same database. Normally the phase of getting the ground truth should be done after all the systems have returned their results because this further prevents cheating. All the meta-data is written in normal text files that are not accessible to the participants so it cannot be used for the query response.

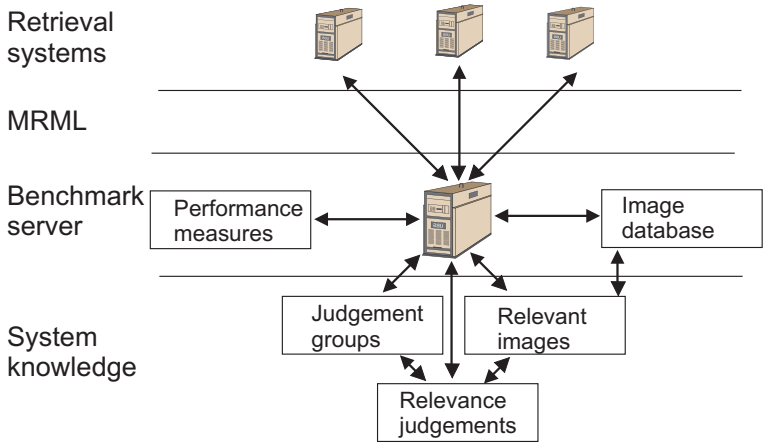

Figure 1. Structure of the automated benchmark.

3.2. What does a CBIRS need to be benchmarked?

The basic prerequisite for a system to be benchmarked in this framework is to talk MRML. 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 (Müller et al., 1999) and several extensions to the protocol have already been proposed.

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 (e.g. benchmark test). In the example below, a client is opening a session on a server and asks for a list of collections available on the server. The server then replies with the list of available collections, in this case one collection with the name UW, for University of Washington. For simplicity not all fields of MRML are shown in these examples.

```
<mrml session-id="1">
  <open-session user-name="anonym" session-name="charm" />
  <get-collections/>
</mrml>

<mrml session-id="1" >
<acknowledge-session-op  session-id="1" />
  <collection-list  >
    <collection  collection-id="c1" collection-name="UW">
    </collection>
  </collection-list>
</mrml>
```

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, and has asked to return two images as the result. All images are referred to by their URLs.

```
<mrml session-id="1" transaction-id="44">
<query-step session-id="1" resultsize="2"
 <user-relevance-list>
  <user-relevance-element user-relevance="1"
   image-location="http://viper.unige.ch/1.jpg" />
  <user-relevance-element user-relevance="-1"
   image-location="http://viper.unige.ch/2.jpg" />
 </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. In the example below two images `1.jpg` and `3.jpg` are returned with relevance 0.9 and 0.75 respectively. Besides the image location a location of a thumbnail to display on screen can be given.

```
<mrml session-id="1" >
<acknowledge-session-op  session-id="1" />
  <query-result>
    <query-result-element-list  >
      <query-result-element  calculated-similarity="0.90"
       image-location="http://viper.unige.ch/1.jpg"
       thumbnail-location="http://viper.unige.ch/1t.jpg" />
      <query-result-element  calculated-similarity="0.75"
       image-location="http://viper.unige.ch/3.jpg"
       thumbnail-location="http://viper.unige.ch/3t.jpg" />
    </query-result-element-list>
  </query-result>
</mrml>
```

To be able to compare systems automatically, they need to use the same set of URLs and the same name as collection id to be chosen by the client, in this case the benchmark. With the help of MRML, all the interaction for example for RF can be automated based on the previous results.

Thus a server to be benchmarked only needs to understand a connection request for opening a session and a query with example images and it needs to create a reply for the opening session by sending all the collections available on the server and a reply to a query by sending a number of results ordered by their similarity to the query. The use of an XLM-based language allows the use of standard parsers.

Figure 2 shows the flow of information of the entire evaluation process. The benchmarking server needs to know the parameters of a CBIRS, i.e. via a web

interface. It can then open a session on the CBIRS and chose the database. After this, queries with all query images the performance can be calculated and as well positive and negative RF can be calculated for the next query steps. The process of generating RF can be repeated several times, and in the end the performance results for the system are shown on screen.

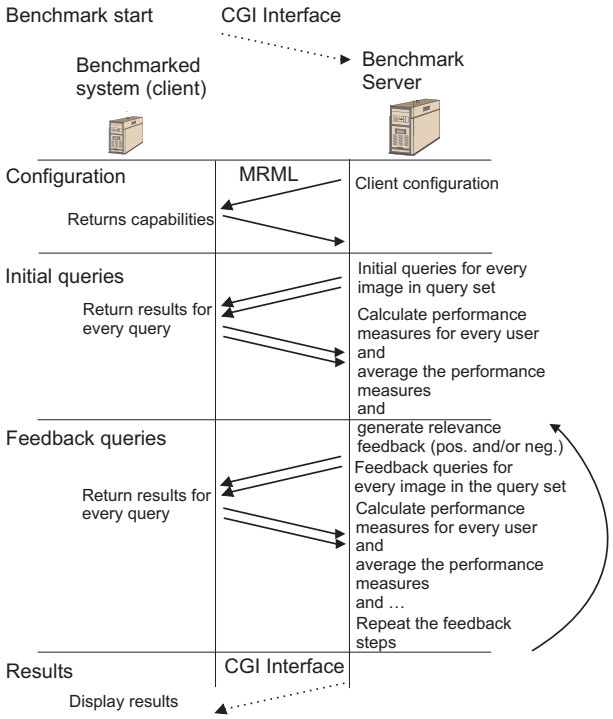

Figure 2. Communication done for executing the benchmark.

In the context of the *Benchathlon*, an interface to MRML is created so a system available via command line can be integrated as well. An MRML-compliant system, the GIFT (GNU Image Finding Tool) is available at http://www.gnu.org/software/gift/.

3.3. What can be evaluated in a benchmark?

There are many different functions that can be tested with a benchmark and a proper benchmark in CBIR definitely needs to incorporate not only one, but several of these tests. All the proposals for evaluation like (Smith, 1998, Dimai, 1999, Leung and Ip, 2000, Müller et al., 2001a) deal only with the QBE paradigm, but (Müller et al., 2000) gives an example for evaluation on browsing and in (Gunther and Beretta, 2001) several methods are proposed to measure the efficiency of systems. It is important to have a mix of measures for efficiency and accuracy.

System developers can then decide to participate at one or several of the benchmarking fields, so specialized systems (i.e. for medical image retrieval) can be tested only in their specialized field. Only the part described in Section 3.3.2 is done in the example evaluation in Section 4

For an evaluation of efficency it is important to state a description of the computer system that the tests are being done with as processor speed and the amount of memory can strongly influence the results.

3.3.1. *Looking for a specific image*

This part is basically taken from (Gunther and Beretta, 2001). Systems have to demonstrate the capability to extract features from a given input image and search for a corresponding image in the image database indexed beforehand. When looking for the exact same image basically the response time is important, whereas the search for an altered input image has to show that it is accurate as well.

- Search for an exact image from the database.
- Search with a cropped part of an image from the database.
- Search with a geometrically altered image from the database, such as rotated, scaled, dilatated or shifted.
- Search with an image where a part is occluded.
- Search with a compressed image of the database, i.e. strong JPEG compression.

This can test the invariances of a retrieval system and especially the retrieval speed. To produce altered versions of images is very easy and the relevance set contains only the original image.

3.3.2. *Looking for a number of similar images*

The search for a number of similar images to a given query image is the standard QBE evaluation. This part will be the main part for a benchmark and a number of measures for efficiency and accuracy have to be developed. We propose to use the average rank measure of the *Benchathlon* and BIRDS-I (Gunther and Beretta, 2001) as a leading measure as it is also used for MPEG-7 (Salembier and Manjunath, 2000) and thus has widespread acceptance. We also propose a set of measures to be able to better compare the systems. The measures proposed in (Müller et al., 2001c) are all well known from the text retrieval field and similar to the TREC benchmark and are also standard measures for CBIR.

Three main areas can be identified, where the last one can be seen as a special case of the second one.

- Evaluation of QBE with known relevance judgments.
- Evaluation of several steps of positive and/or negative feedback.
- Evaluate how well a system can adapt the output for the same starting image but with different ground truth sets and thus different RF.

The adaptation of a system to what a user really wants can only be shown when for the same query several relevance sets are available. An image with a tree in front of a sunset can, for example, be used to search for sunsets or for trees and a good system must be able to adapt the results with RF according to the users' needs. An example comparison of two systems is done in Section 4.

3.3.3. *Looking for a sketch of an image*
This test is very similar to QBE, but the information is in general incomplete as somebody drawing a sketch is normally concentrating on the object and not drawing the background, and normally the time you take to draw a sketch is limited. Otherwise the same measures for efficiency and accuracy can be used.

3.3.4. *Target search (or called image browsing)*
Image browsing was first proposed by (Cox et al., 1996) with the PicHunter system. The goal is to find a given image in the database and the performance is measured by counting the number of image that a user has to look at before finding the target. A benchmark for image browsers is presented in (Müller et al., 2000).

3.3.5. *Practical application tests*
This test models practical functions of a system that are routinely used. An index of an image database can be generated and the time for this is measured. Then a number of images is added into the database and then images are added into the database and a query with this image is executed directly afterwards. Performance measures have to measure the efficiency of the system with respect to a given task.

- Feature extraction and index generation.
- Inserting an image into the database.
- Inserting an image into the database and find a known image similar to this one.

Other functions can be added to this part for completion.

3.3.6. *Measure the scalability of a CBIR system*
For many application it is important that a CBIR system can deal with very large databases in an efficient manner. To show the scalability of a systems, the time for several actions like feature extraction, index generation and image querying can be measured for several collection sizes, for example with $10,000$, $100,000$ and $1,000,000$ images. This gives means to interpolate the response time for even larger image databases.

3.3.7. *Tests for special application areas*

Images from different application fields have different characteristics and specialized programs should be tested accordingly. *Medical images* are for example black and white and also *satellite images* might need color characteristics different from the ones used for *stock photography*. Many of them can be tested with the same performance measures, but fields such as *trademark retrieval* will need different performance measures as for trademark retrieval recall is the essential point and trademark researchers are normally used to look at a large number of images.

The process of getting relevance judgments can be different because in certain fields real ground truth is available and not as subjective as for photographs.

3.3.8. *Evaluation of CBIR interfaces*

It might not be possible to measure CBIR interface completely automatic, but users can for sure determine how well the information is presented and how easy it is to give feedback or find groups of similar images. Measures for the quality of interfaces have to be developed. Interfaces in the 3 dimensional domain such as (Nakazato and Huang, 2001) show that interfaces for CBIRSs can be studied much more than this is the case at the moment.

3.4. Flexibility with respect to the problems in benchmarking

A benchmarking framework has to have a maximum flexibility so it can be used for all the performance tests described above and so the addition and testing of new parameters can easily be done adjusting the system and without a new system design. We chose MRML as the query language but the other aspects important for evaluation can easily be adapted in configuration files.

3.4.1. *Image databases*

The system can use any image database and also databases of other objects that can be specified by a URL as a unique identifier. We would like all the images to be freely available on the internet and we also would like to have the possibility of distributed image databases. We tried out the benchmark server with several image databases. For the example evaluation in Section 4 we chose the image database of the University of Washington because it is available free of charge and without copyright.

3.4.2. *Relevance judgments*

The benchmark server can work with a single set of relevant images, but it can also have several different relevance sets for the same query image. Thus it is possible to use groupings of a database as relevance judgments as well as expert opinions, real user judgments or relevance judgments derived from annotations. The example evaluation in Section 4 is done with the groupings of the UW database, but it has also been tested with several relevance categories

for each query image. In this case the results are averaged over all relevance sets and RF is calculated separately for each relevance set.

3.4.3. *Performance measures*

New performance measures can easily be added to the system and a number of measures for efficiency and accuracy is already calculated in Section 4. Especially in the beginning phases it might be important to try out an even larger number of performance measures to compare the information they contain. As a leading measure a normalized average rank measure as explained in (Gunther and Beretta, 2001) and used in MPEG-7 (Salembier and Manjunath, 2000) is chosen, but the example evaluation shows well that the different measures all have their utility.

3.5. A web interface to a permanently accessible benchmark server

In (Müller et al., 2001b) a web interface is added to a benchmarking server to have a benchmark constantly accessible `http://viper.unige.ch/evaluation/`. The results may not be official, but it is a good means to test the MRML technology, and it gives a quick evaluation of a system, so even small changes in the features or the query mechanism can be checked straight away.

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*

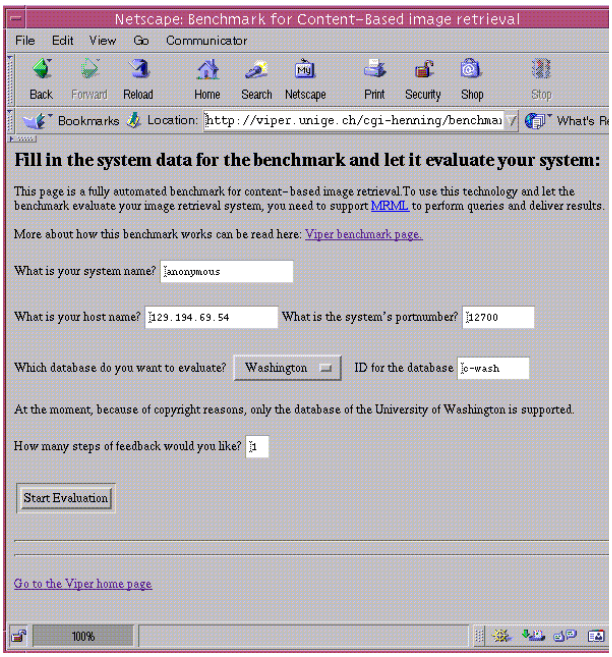

Figure 3. A screenshot of the web-based benchmark.

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.

If a number of systems uses this benchmark it is also possible to do an online ranking of systems performing best at the test.

4. An example comparison

To demonstrate the usefulness of our benchmarking framework, we compared a retrieval system using a simple histogram intersection (HI) based on the HSV space with 166 colors (18 hues, 3 saturations, 3 values and 4 grey levels), with the *Viper* system described in (Squire et al., 1999) using local and global color and texture measures. All tests are done on a four processor PC running Linux with Intel Pentium III 550 MHz CPUs and 1 GB of main memory. The indices are stored and read from hard disk.

The DB of the UW consists of 922 images that are in 14 different categories, normally geographical areas. We use 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 queries are in general relatively easy and often the image sets do contain a few dominant colors so the HI is expected to work well. We also always receive the entire database as a result set, so for the averaged normalized rank we do not need to worry about penalizing missed images as the entire database is retrieved.

Table I. Results for *Viper* with the Washington DB.

Measure	no RF	RF 1	RF 2	RF 3	RF 4
N_R	65.14	65.14	65.14	65.14	65.14
t	1.88 s.	2.88 s.	3.23 s.	3.43 s.	3.54 s.
$Rank_1$	1.5	1	1	1	1
$R(P(.5))$	.3798	.5520	.6718	.6594	.7049
$\bar{Rank}$	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 I shows the results for the Viper system and we can see that the first two feedback steps strongly enhance the results. The rank of the first relevant shows that only in the first step there was a non-relevant in the first position in at least one result and from then on there were always relevant images at the beginning of the results. The speed for a query is getting slower with more feedback images being added, but only the first feedback step is significantly slower than the preceding one. The measure $P(20)$ shows that in the first query step an average of 11 of the first 20 images was relevant and this rises to 18 out of 20 with four steps of feedback.

Table II. Results for HI with the Washington DB.

Measure	no RF	RF 1	RF 2	RF 3	RF 4
N_R	65.14	65.14	65.14	65.14	65.14
t	1.45 s.	1.76 s.	1.84 s.	1.91 s.	1.97 s.
$Rank_1$	7.29	1.07	1	1	1
$R(P(.5))$	.313	.4857	0.454	.4854	.4638
$\bar{Rank}$	182.26	148.08	135.48	133.14	133.73
$\widetilde{Rank}$	.1634	.1273	.1134	.1109	.1115
$P(20)$	.5143	.7393	.7571	.7571	0.775
$P(50)$	.4286	.53	.5571	.57	.5657
$P(N_R)$	.3954	.5313	.5525	.5644	.557
$R(100)$	.4977	.5959	.6268	.6371	.6373

When we compare Table I with Table II we can see that the HI is faster than the *Viper* system and for the one-shot-query and even the first step of feedback the results are very similar with a few measures even being better for the HI. Only the first relevant image is significantly worse for the HI in the first query step. But starting from the second feedback step the histogram intersection does not get much better whereas the *Viper* system has a significantly better performance. After four steps of feedback the precision after 20 and 50 images is 14% and 6% better and the recall where the precision drops below .5 is even 25% better.

Figure 4 shows that in the first query step and the first step of RF the *Viper* system performs only for the first few returned images better than a HI and in the middle part both systems are very similar. A 100% recall is even reached earlier with the HI.

Figure 5 shows that *Viper* is much better with respect to several steps of RF than a simple HI. With each step the gap in the performance widens and the additional feature information in *Viper* proves to be important.

This example evaluation was done completely automatic and it shows that systems can be compared with each other pretty well with such a mix of performance measures. It also highlights the importance of RF as the two systems perform quite similar in the first feedback step but a large gap is visible for further feedback steps.

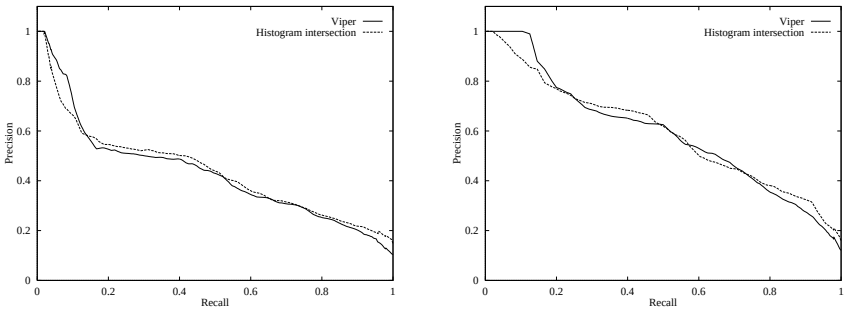

Figure 4. Comparison between Viper and HI of the first query step and the first step of RF.

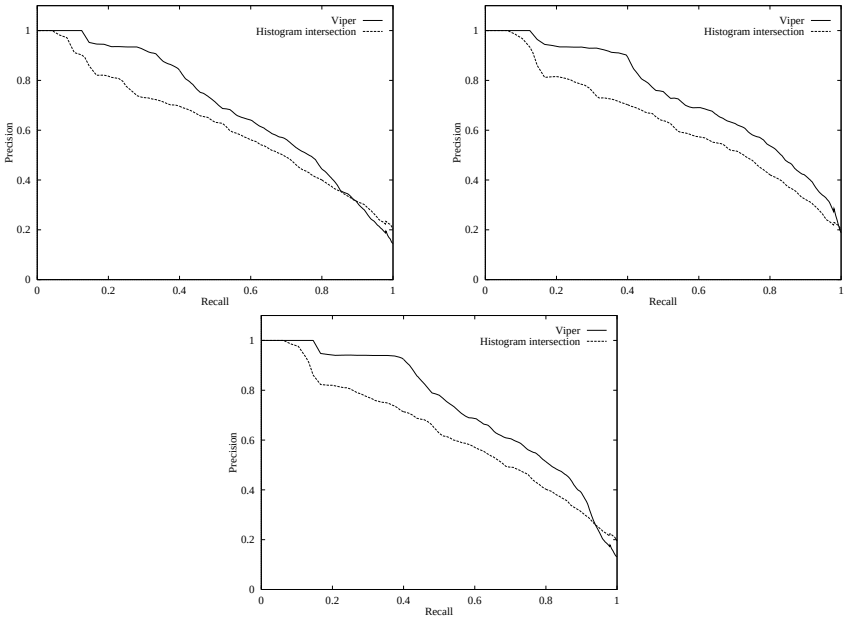

Figure 5. Comparison between *Viper* and HI of RF steps 2, 3 and 4.

5. Conclusion and future work

We present in this article a framework that needs to be established for benchmarking in CBIR. Several parts of the framework are already implemented which is shown with an example evaluation but it is very important for any benchmark to get acceptance in the research community. To get this acceptance an independent governing body needs to be established that promotes the benchmark. To do this we can learn from successful benchmarks such as TREC in text retrieval and TPC for the database community. The *Benchathlon* is an important step into this direction.

It is very important to keep the discussion on performance evaluation going and to be able to convince everyone that in the long run everybody will profit from a proper performance evaluation. A benchmarking event should not so much be seen as a competition but much more as a discussion platform to compare techniques and features and to learn from others. Other benchmarks like the TPC took a while before they became official benchmarks with a governing body, but it is important to keep the discussion on benchmarking in CBIR going.

Much work definitely needs to be done for the foundations of a benchmark. Many images are now freely available, but work needs to be done to construct several large and varied image databases as well as image databases for special application areas such as medical images or trademarks. The most work intensive part will definitely be the generation of relevance judgments for the images and this is also the most important because the quality of the results is directly dependent on the quality of these relevance judgments

References

Beretta, G. and R. Schettini (eds.): 2002, 'Internet Imaging III', Vol. 4672 of *SPIE Proceedings*. San Jose, California, USA: . (SPIE Photonics West Conference).

Borlund, P. and P. Ingwersen: 1997, 'The development of a method for the evaluation of interactive information retrieval systems'. *Journal of Documentation* **53**, 225–250.

Cleverdon, C. W.: 1962, 'Report on the testing and analysis of an investigation into the comparative efficiency pf indexing systems'. Technical report, Aslib Cranfield Research Project, Cranfield, USA.

Cleverdon, C. W., L. Mills, and M. Keen: 1966, 'Factors Determining the Performance of Indexing Systems'. Technical report, ASLIB Cranfield Research Project, Cranfield.

Cox, I. J., M. L. Miller, S. M. Omohundro, and P. N. Yianilos: 1996, 'Target Testing and the `PicHunter` Bayesian Multimedia Retrieval System'. In: *Advances in Digital Libraries (ADL'96)*. Library of Congress, Washington, D. C., pp. 66–75.

Dimai, A.: 1999, 'Assessment of Effectiveness of Content-based Image Retrieval Systems'. In (Huijsmans and Smeulders, 1999), pp. 525–532, Springer-Verlag.

Dy, J. G., C. E. Brodley, A. Kak, C.-R. Shyu, and L. S. Broderick: 1999, 'The Customized-Queries Approach to CBIR using Using EM'. In: *Proceedings of the 1999 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'99)*. Fort Collins, Colorado, USA, pp. 400–406.

Eakins, J. P., B. J. M., and M. E. Graham: 1998, 'Similarity Retrieval of Trademark Images'. *IEEE Multimedia Magazine* **April June**, 53–63.

Gunther, N. J. and G. Beretta: 2001, '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.

Harman, D.: 1992, 'Overview of the first Text REtrieval Conference (TREC-1)'. In: *Proceedings of the first Text REtrieval Conference (TREC-1)*. Washington DC, USA, pp. 1–20.

Huijsmans, D. P. and A. W. M. Smeulders (eds.): 1999, 'Third International Conference On Visual Information Systems (VISUAL'99)', No. 1614 in Lecture Notes in Computer Science. Amsterdam, The Netherlands: Springer-Verlag.

ICME'2001: 2001, 'Proceedings of the second International Conference on Multimedia and Exposition (ICME'2001)'. Tokyo, Japan: IEEE, IEEE.

Jörgensen, C.: 1995, 'Classifying Images: Criteria for Grouping as Revealed in a Sorting Task'. In: *Proceedings of the 6th ASIS SIG/CR Classification research Workshop*. Chicago, Il, USA, pp. 65–78.

Jörgensen, C. and P. Jörgensen: 2002, 'Testing a vocabulary for image indexing and ground truthing'. In (Beretta and Schettini, 2002). (SPIE Photonics West Conference).

Koskela, M., J. Laaksonen, S. Laakso, and E. Oja: 2000, 'Evaluating the performance of Content-Based Image Retrieval Systems'. In (Laurini, 2000), Springer-Verlag.

Laurini, R. (ed.): 2000, 'Fourth International Conference On Visual Information Systems (VISUAL'2000)', No. 1929 in Lecture Notes in Computer Science. Lyon, France: Springer-Verlag.

Leung, C. and H. Ip: 2000, 'Benchmarking for Content-Based Visual Information Search'. In (Laurini, 2000), Springer-Verlag.

Markkula, M. and E. Sormunen: 1998, 'Searching for Photos - Journalists' Practices in Pictorial IR'. In: J. P. Eakins, D. J. Harper, and J. Jose (eds.): *The Challenge of Image Retrieval, A Workshop and Symposium on Image Retrieval*. Newcastle upon Tyne, The British Computer Society.

Mokhtarian, F., S. Abbasi, and J. Kittler: 1996, 'Efficient and Robust Retrieval by Shape Content through Curvature Scale Space'. In: A. W. M. Smeulders and R. Jain (eds.): *Image Databases and Multi-Media Search*. Kruislaan 403, 1098 SJ Amsterdam, The Netherlands, pp. 35–42, Amsterdam University Press.

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

Müller, H., W. Müller, S. Marchand-Maillet, D. M. Squire, and T. Pun: 2001a, 'Automated Benchmarking in Content-Based image Retrieval'. In (ICME'2001, 2001), IEEE.

Müller, H., W. Müller, S. Marchand-Maillet, D. M. Squire, and T. Pun: 2001b, 'A web-based evaluation system for content-based image retrieval'. In: *Proceedings of the ACM Multimedia Workshop on Multimedia Information Retrieval (ACM MIR 2001)*. Ottawa, Canada, pp. 50–54, The Association for Computing Machinery.

Müller, H., W. Müller, D. M. Squire, S. Marchand-Maillet, and T. Pun: 2001c, 'Performance Evaluation in Content-Based Image Retrieval: Overview and Proposals'. *Pattern Recognition Letters* **22**(5).

Müller, W., S. Marchand-Maillet, H. Müller, and T. Pun: 2000, 'Towards a fair benchmark for image browsers'. In: *SPIE Photonics East, Voice, Video, and Data Communications*. Boston, MA, USA.

Müller, W., Z. Pečenović, A. P. de Vries, D. M. Squire, H. Müller, and T. Pun: 1999, 'MRML: Towards an extensible standard for multimedia querying and benchmarking – Draft Proposal'. Technical Report 99.04, Computer Vision Group, Computing Centre, University of Geneva, rue Général Dufour, 24, CH-1211 Genève, Switzerland.

Nakazato, M. and T. S. Huang: 2001, '3D Mars: Immersive Virtual Reality for Content-Based Image Retrieval'. In (ICME'2001, 2001), pp. 45–48, IEEE.

Narasimhalu, A. D., M. S. Kankanhalli, and J. Wu: 1997, 'Benchmarking Multimedia Databases'. *Multimedia Tools and Applications* **4**, 333–356.

Pfund, T. and S. Marchand-Maillet: 2002, 'Dynamic multimedia annotation tool'. In (Beretta and Schettini, 2002). (SPIE Photonics West Conference).

Salembier, P. S. and B. S. Manjunath: 2000, 'Audiovisual Content Description and Retrieval: Tools and MPEG-7 Standardization Techniques'. In: *IEEE Internation Conference on Image Processing (ICIP 2000)*. Vancouver, BC, Canada.

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

Shyu, C.-R., A. Kak, C. Brodley, and L. S. Broderick: 1999, 'Testing for Human Perceptual Categories in a Physician-in-the-loop CBIR System for Medical Imagery'. In: *IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL'99)*. Fort Collins, Colorado, USA, pp. 102–108.

Smith, J. R.: 1998, 'Image Retrieval Evaluation'. In: *IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL'98)*. Santa Barbara, CA, USA, pp. 112–113.

Sparck Jones, K. and C. van Rijsbergen: 1975, 'Report on the need for and provision of an ideal information retrieval test collection'. British Library Research and Development Report 5266, Computer Laboratory, University of Cambridge.

Squire, D. M., W. Müller, and H. Müller: 1999, 'Relevance feedback and term weighting schemes for content-based image retrieval'. In (Huijsmans and Smeulders, 1999), pp. 549–556, Springer-Verlag.

Squire, D. M. and T. Pun: 1997, 'A Comparison of Human and Machine Assessments of Image Similarity for the Organization of Image Databases'. In: M. Frydrych, J. Parkkinen, and A. Visa (eds.): *The 10th Scandinavian Conference on Image Analysis (SCIA'97)*. Lappeenranta, Finland, pp. 51–58, Pattern Recognition Society of Finland.

Vorhees, E. M. and D. Harmann: 1998, 'Overview of the Seventh Text REtrieval Conference (TREC-7)'. In: *The Seventh Text Retrieval Conference*. Gaithersburg, MD, USA, pp. 1–23.