CoPhIR: COntent-based Photo Image Retrieval

(White paper)

Fabrizio Falchi
ISTI-CNR
Pisa, Italy
fabrizio.falchi@isti.cnr.it

Claudio Lucchese
ISTI-CNR
Pisa, Italy
claudio.lucchese@isti.cnr.it

Raffaele Perego
ISTI-CNR
Pisa, Italy
raffaele.perego@isti.cnr.it

Fausto Rabitti
ISTI-CNR
Pisa, Italy
fausto.rabitti@isti.cnr.it

May 30, 2008

Abstract

As the number of digital images is growing fast and Content-based Image Retrieval (CBIR) is gaining in popularity, CBIR systems should leap towards Web-scale datasets. In this paper, we report on our experience in building a test collection of more than 50 million images, with the corresponding descriptive features, to be used in experimenting new techniques for similarity searching. Since no collection of this scale was available for research purpose, we had to tackle the non-trivial process of image crawling and descriptive feature extraction (we used five MPEG-7 features) using the European EGEE computer GRID. The result of this effort is a test collection, the first of such scale, that will be opened to the research community for experiments and comparisons.

1 Introduction

It is common in database community to open a discussion with some reference to the data explosion. According to recent studies, in the next three years, we will create more data than has been produced in all of human history. But where is all this data coming from? The Enterprise Strategy Group[1] estimates that more than 80 billion photographs are taken each year. To store them would require 400 petabytes of storage. If an average digital photo occupies 750 KB, it takes as much space as 30 pages of digital text, i.e., about 18,000 words. The

[1] http://www.enterprisestrategygroup.com/

management of digital images promises to emerge as a major issue in many areas providing a lot of opportunities in the next couple of years, particularly since a large portion of pictures still remains as "unstructured data".

Current State of the Art

Current searching engines headed by Google are in the center of current information age; Google answers daily more than 200 million queries against over 30 billion items. However, the search power of these engines is typically limited to text and its similarity. Since less than 1 % of the Web data is in textual form, the rest being of multimedia/streaming nature, we need to extend our next-generation search to accommodate these heterogeneous media. Some of the current engines search these data types according to textual information or other attributes associated with the files.

An orthogonal approach is the Content-based Image Retrieval (CBIR). It is not a new area as demonstrated by a recent survey [3] which reports on nearly 300 systems, most of them exemplified by prototype implementations. However, the typical database size is in the order of thousands of images. Very recent publicly-available systems, such as ImBrowse[2], Tiltomo[3] or Alipr[4], declare to index hundreds of thousands of images. There is a high discrepancy between these numbers and the volumes of images available on current Web, so we decided to investigate the situation by shifting the current bounds up by two orders of magnitude.

Background, Objectives and Approach

This work is a result of cooperation of research groups within the European project SAPIR (Search on Audio-visual content using Peer-to-peer Information Retrieval)[5]. This project aims at finding new content-based methods to analyze, index, and retrieve the tremendous amounts of speech, image, video, and music which are filling our digital universe. In this context, we intended to develop a large-scale distributed architecture for indexing and searching in image collections according to visual characteristics of their content. The system should be able to scale to the order of tens of millions. To reach this goal, a collection of such size together with respective descriptive features is needed. We have crawled the images from a photo-sharing system Flickr[6] and have extracted five MPEG-7 features from every image. This source has also the advantage of having associated user-defined textual information, which could be used for future experiments with combining search on text and visual content.

[2] http://media-vibrance.itn.liu.se/
[3] http://www.tiltomo.com/
[4] http://www.alipr.com/
[5] SAPIR European Project, IST FP6: http://www.sapir.eu/
[6] http://www.flickr.com/

Scalability Challenge

Our scalability objective goes very far beyond the current practice. On the data acquisition level, this would require to download and process 15 TB to 50 TB of data, depending on the image resolution. Moreover, we need a storage space for the image descriptors (including the MPEG-7 features) and the thumbnails of about 1.5 TB. In practice, we have to bear in mind that the image crawling and feature extraction process would take about 12 years on a standard PC and about 2 years using a high-end multi-core PC.

Paper Structure

The rest of the paper is organized as follows. Section 2 describes the process of building the test collection: crawling the images, extracting the MPEG-7 features, and organizing the result into the test collection which will be opened to research community. Finally, Section 3 analyzes the results obtained.

2 Building the Image Collection

Collecting a large amount of images for investigating CBIR issues is not an easy task, at least from a technological point of view. The challenge is mainly related to the size of the collection we are interested in. Shifting state-of-the-art bounds of two orders of magnitude means building a 100 million collection, and this size makes very complex to manage every practical aspect of the gathering process. However, since the absence of a publicly available collection of this kind has probably limited the academic research in this interesting field, we tried to do our best to overcome these problems. The main issues we had to face were:

1. identification of a valid source;
2. efficient downloading and storing of such a large collection of images;
3. efficient extraction of metadata (MPEG-7 visual descriptors and others) from the downloaded images;
4. providing reliable data-access to metadata.

In the following we will discuss the above issues by describing the challenges, the problems we encountered, and the decisions we took.

2.1 Choosing the Data Source

Crawling the Web is the first solution if you are looking for a practically unlimited source of data. There are plenty of images on the Web, varying in quality from almost professional to amateur photos, from simple drawings to digital cartoons.

There are also many different ways to retrieve such data. The first option is to exploit spider agents that crawl the Web and download every image found on

the way. Of course this would result in a large amount of time and bandwidth wasted in downloading and parsing HTML pages, possibly gathering only a few images. The authors of [2] report that the average number of images hyperlinked by HTML pages is varying. In their experiments with the Chilean Web, they repeatedly downloaded each time about 1.3 million Web pages. The number of images retrieved were 100,000 in May 2003, 83,000 in August 2003 and 200,000 in January 2004. Thus, assuming that these percentages are still valid today, we can expect that:

> ***Fact:*** To gather 100 million images, we would have to download and parse 650 million to 1.5 billion Web pages.

A second option, which may be more efficient, is to take advantage of the image search service available on most commercial Web search engines. Just feeding the search engine with queries generated synthetically, or taken from some real query log, would provide us with plenty of images.

This abundance and diversity of Web images is definitely a plus. Not only because we want a large collection, but also because we want our collection to spread over different kinds of images. A problem is instead given by the large differences in the quality and size of the retrieved images. A large portion of them are decoration elements like buttons, bullet list icons, and many other are very small images or photo thumbnails. These images are not suitable for our purposes and would pollute the corpus, but some of them could be filtered out automatically as the feature extraction software is likely to fail on images with non-standard sizes.

However, for the need of high-quality data, we finally decided to follow a third way: crawling one of the popular photo sharing sites born in the last years with the goal of providing permanent and centralized access to user-provided photos. This approach has several advantages over the aforementioned approaches.

Image Quality In fact photo sharing sites like Flickr, PhotoBucket, Picasa, Kodak EasyShare Gallery, Snapfish, etc. mainly store high-quality photographic images. Most of them are very large since they come from 3–8 Megapixel cameras, and have a standard 4:3 format.

Collection Stability These sites provide quite static, long term and reliable image repositories. Although images may be deleted or made private by the owners, this happens quite rarely. Most photos stay available for a long time and they are always easily accessible. Conversely, the Web is much more dynamic, images change or are moved somewhere else, pages are deleted and so on.

Legal Issues The above consideration is very important also when considering the legal issues involved in the creation of such collection of images. In fact, storing for a long time a publicly available image may in some case violate author's copyrights. We are mainly interested in the visual descriptors extracted from the images, but any application of CBIR has to access the original files

for eventually presenting the results retrieved to a human user. Since Photo sharing sites are fairly static, we can build a quite stable collection without permanently storing the original files, but maintaining only the hyperlinks to the original photos that can be accessed directly at any time.

Rich Metadata Finally, photo sharing sites provide a significant amount of additional metadata about the photos hosted. The digital photo file contains information about the camera used to take the picture, the time when it was taken, the aperture, the shutter used, etc. More importantly, each photo comes with the name of the author, its title, a description, often with user-provided tags. Sometimes also richer information is available such as comments of other users on the photo, the GPS coordinates of the location where the photo was taken, the number of times it was viewed, etc.

Among the most popular photo sharing sites, we chose to crawl Flickr, since it is one with the richest additional metadata and provides an efficient API[7] to access its content at various levels.

2.2 Crawling the Flickr Contents

It is well known that the graph of Flickr users, similarly to all other social media applications, is scale free [5]. We thus exploited the small-world property of this kind of graphs to build our huge photo collection. By starting from a single Flickr user and following friendship relations, we first downloaded a partial snapshot of the Flickr graph. This snapshot of about one million distinct users was crawled in February 2007. We then exploited the Flickr API to get the whole list of public photo IDs owned by each of these users. Since Flickr Photo IDs are unique and can be used to unequivocally devise an URL accessing the associated photo, in this way we have easily created a 4.5 GB file with 300 million distinct photo IDs.

In the next step, we decided what information to download for each photo. Since the purpose of the collection is to enable a general experimentation on various CBIR research solutions, we decided to retrieve almost all information available. Thus, for each photo: title and description, identification and location of the author, user-provided tags, comments of other users, GPS coordinates, notes related to portions of the photo, number of times it was viewed, number of users who added the photo to their favourites, upload date, and, finally, all the information stored in the EXIF header of the image file. Naturally, not all these metadata are available for all photos. In order to support content based search, we extracted several MPEG-7 *visual descriptors* from each image [7]. A visual descriptor characterizes a particular visual aspect of the image. They can be, therefore, used to identify images which have a similar appearance. Visual descriptors are represented as vectors, and the MPEG-7 group proposed a distance measure for each descriptor to evaluate the similarity of two objects [6].

[7] http://www.flickr.com/services/api/

Finally, we have chosen the five MPEG-7 visual descriptors described below [6, 4]:

Scalable Colour It is derived from a colour histogram defined in the Hue-Saturation-Value colour space with fixed colour space quantization. The histogram values are extracted, normalized and nonlinearly mapped into a four-bit integer representation. Then the Haar trasform is applied. We use the 64 coefficients version of this descriptor.

Colour Structure It is also based on colour histograms but aims at identifying localized colour distributions using a small structuring window. We use the 64 coefficients version of this descriptor.

Colour Layout It is obtained by applying the DCT transformation on a 2-D array of local representative colours in Y or Cb or Cr colour space. This descriptor captures both colour and spatial information. We use the 12 coefficients version of this descriptor.

Edge Histogram It represents local-edge distribution in the image. The image is subdivided into 4×4 sub-images, edges in each sub-image are categorized into five types: vertical, horizontal, 45° diagonal, 135° diagonal and non-directional edges. These are then transformed in a vector of 80 coefficients.

Homogeneous Texture It characterizes the region texture using the mean energy and the energy deviation from a set of 30 frequency channels. We use the complete form of this descriptors which consist of 62 coefficients.

There are several other visual descriptors in the MPEG-7 standard which can be useful, for example, for specialized collections of images (e.g. medical). Our experience [1] suggests that these five descriptors perform quite well on non-specialized images, such as the ones in our collection.

Unfortunately, the extraction of MPEG-7 visual descriptors from high-quality images is very computationally expensive. Although the MPEG-7 standard exists for many years, there is not an optimized extraction software publicly available. To extract descriptors, we used the MPEG eXperimentation Model (MPEG-XM) [4] that is the official software certified by the MPEG group that guarantees the correctness of the extracted features. This software running on a AMD Athlon XP 2000+ box takes about 4 seconds to extract the above five features from an image of size 500×333 pixels. Therefore, even without considering the time needed to download the image and all additional network latencies involved, we can estimate that:

> ***Fact:*** A single standard PC would need about 12 years to process a collection of 100 million images.

It was thus clear that we needed a large number of machines working in parallel to achieve our target collection of 100 million images in a reasonable amount of time. For this reason, we developed an application that allows to

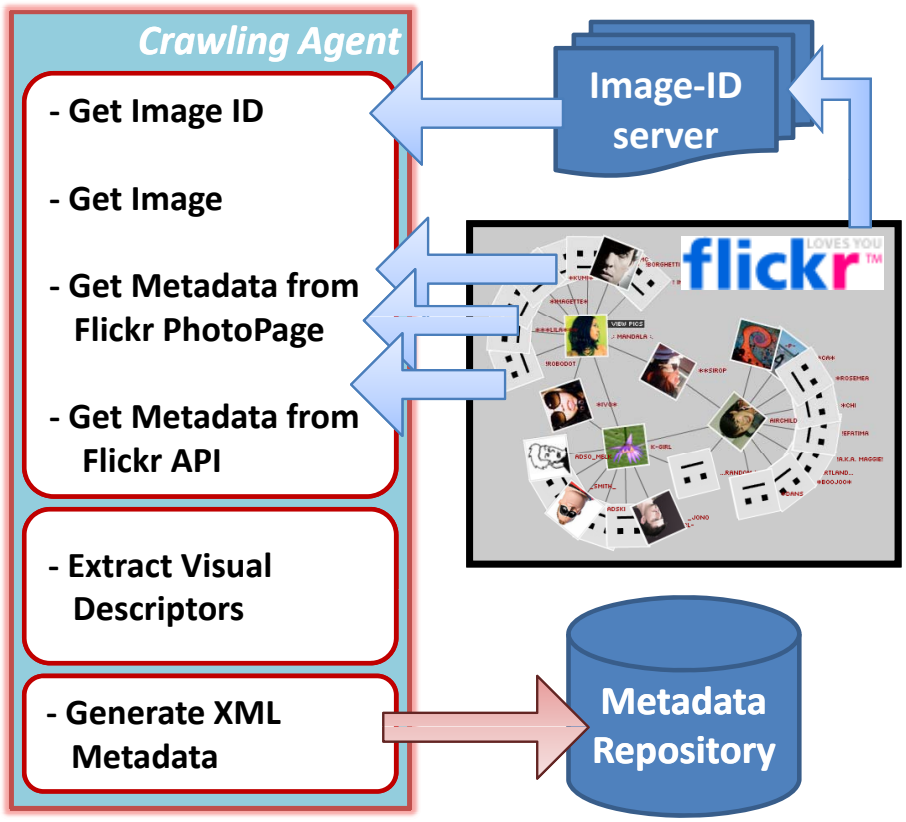

Figure 1: Organization of the crawling and feature extraction process.

process images in parallel on an arbitrary (and dynamic) set of machines. This application is composed of three main components: the *image-id server*, the *crawling agents*, and the *repository manager* as shown in Figure 1.

The image-id server was implemented in PHP as a simple Web application accomplishing the task of providing crawling agents with an arbitrary number of photo identifiers not yet processed.

The crawling agent is the core part of our application. It loops asking the image-id server for a new set of image identifiers to process. Once it obtains a set from the server, it starts the actual retrieval and feature extraction process. Given a photo ID, the first step is to issue an HTTP request and download the corresponding *Flickr photo-page*. This is parsed to retrieve the URL of the image file and some of the metadata discussed above. Thanks to Flickr APIs, this metadata is then enriched with other information (title of the photo, description, tags, comments, notes, upload date, user name, user location, GPS coordinates, etc.).

We downloaded medium-resolution version of the photos, which have the larger dimension 500 pixels. This improves the independence of extracted features from image size and reduces the cost of processing large images. The MPEG-XM [7] software is used to extract the aforementioned five visual descriptors.

The extracted features and all the available metadata are used to produce an XML file containing the knowledge we collected about the image. Finally, a thumbnail is also generated from the photo. The XML file and the thumbnail of the image are sent to a Web-service provided by the *repository manager*.

The repository manager runs on a large file-server machine providing 10 TB of reliable RAID storage. In addition to receive and store the results processed by the crawling agents, the repository manager also provides statistic information about the state of the crawling process and basic access methods to the collection.

Fact: Disks provide a potentially unreliable storage and actually two disks had to be replaced.

2.3 Using the GRID for Crawling and Feature Extraction

We have considered GRID to be the right technology to obtain large amount of computing power we needed. GRID is a very dynamic environment that allows to transparently run a given application on a large set of machines. In particular, we had the possibility to access the EGEE (Enabling Grids for E-sciencE) European GRID infrastructure[8] provided to us by the DILIGENT IST project[9].

We were allowed to use 35 machines spread across Europe (see Figure 2). We did not have an exclusive access to these machines and they were not available all the time. Both hardware and software configurations were heterogeneous: they had various CPUs, memory, disk space, but also in the libraries, software (e.g. Java), and Linux versions installed. Thus, we had to build a self-contained crawling agent.

The crawling agent is logically divided into two modules. The first one accomplishes the communication with the image-id server, crawls Flickr website, uses Flickr APIs, and sends the result of the computation to the repository manager. This was coded in Java to improve portability. However, since we could not assume the presence of the Java virtual machine on every machine, we incorporated into the crawling agents also a JVM and the required Java libraries. Due to the latencies of the crawling task, the crawling agent can instantiate a number of threads, each of them taking care of processing a different image. The settings which proved well is to have four threads per agent (per one CPU core) and to process a maximum of 1,000 images. These parameters induced computations times of 20 to 60 minutes depending on the CPU speed.

[8] http://www.eu-egee.org/
[9] http://www.diligentproject.org/

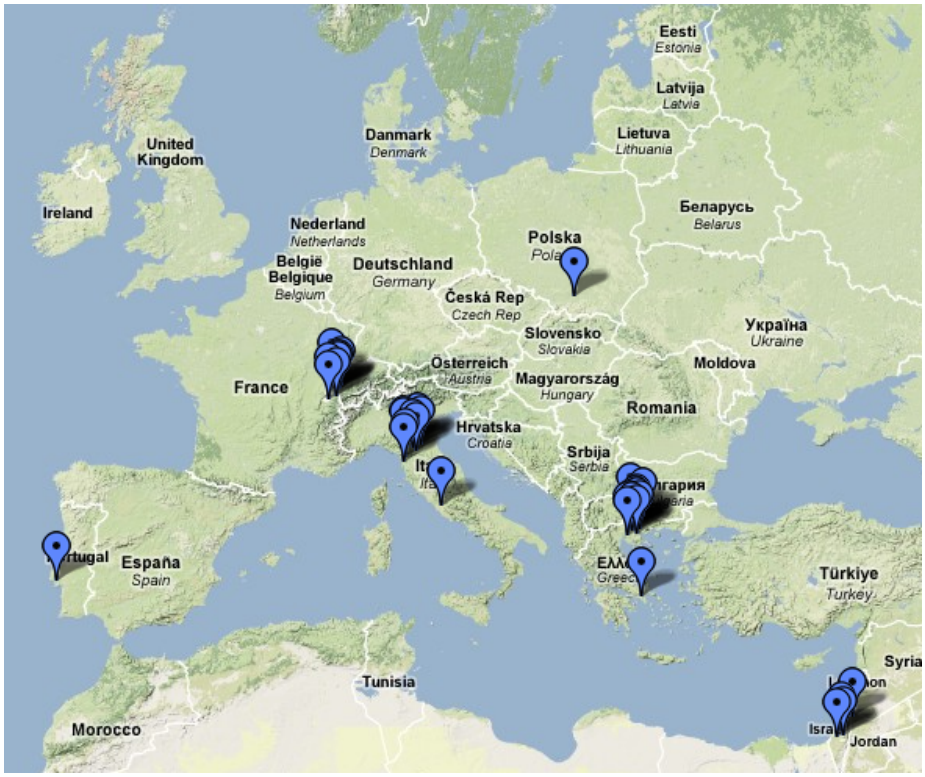

Figure 2: Machines collaborating on the crawling.

The second module of the crawling agent is the MPEG-XM feature extraction software. Since the MPEG-XM software is not maintained, it has become incompatible with the recent compilers and libraries versions. For this reason and for the heterogeneity of the GRID, we encapsulated into the crawling-agents also all the libraries it uses.

Submitting a job to a GRID infrastructure, the user does not have a full control on the time and location where the job runs. The GRID middleware software accepts the job description and schedules it on the next available machine according to internal policies related to the load of each node, the priority of the different organization using the GRID infrastructure, etc. In our case, the job always first downloads the crawling-agent package from our repository-manager and then runs the software contained in the package. The GRID provides a best-effort service, meaning that a job submitted to the GRID may be rejected and never executed. Indeed, there are several factors that may cause the failure of a job submission.

Fact: From the 66,440 jobs submitted, only 44,333 were successfully executed that means that 33,3 % of the jobs failed for GRID resources unavailability.

Our straightforward approach together with the self-scheduling of images by each crawling agent has two important advantages. First, in case the GRID middleware is not able to deploy the given job, there would be no consequences in the remainder of the system, especially, no image will be skipped. Second, in case of a software update, it is just needed to replace the old version on the repository manager with the new one.

Not all of the GRID machines were available through the crawling period and, therefore, we also used a set of local machines in Pisa which processed the images during the GRID idle time. We thus reached the total of 73 machines participating in the crawling and feature extraction process.

The crawling process took place in two separate periods, both because of GRID availability and because we needed to consolidate the data after the first period. In Figure 3, we report on the number of machines available during the crawling process. During the first period, the GRID provided an average of 14.7 machines out of the 35 and, simultaneously, there were 2.5 local machines available, on average. Also the availability of the machines during the day was unstable: The local machines were mainly available over night while some of the GRID machines were available only for a few hours per day. During the second period, only one powerful multiprocessor machine was available from the GRID, and we could continue the process only with our local resources.

Figure 4 reports the total number of images processed by each site. The best machine (provided by the GRID) processed about 17 % of the whole collection – this is a very powerful machine equipped with seven quad-core CPUs. The second best is a local machine used only during the second phase is equipped with two quad-cores Intel Xeon 2.0 GHz and it processed about 13 % of the collection. These machines were the most powerful and the most constantly

available over time.However, the largest total contribution came from a number of machines each of which was able to process only a small number of images.

2.4 The CoPhIR Test Collection

The result of this complex crawling and image processing activity is a test collection that served as the basis of the experiments with content-based image retrieval techniques and their scalability characteristics, in the context of the SAPIR project.

We have not yet reached the target of 100 million images: we have made a check-point at about half of the target. The current image test collection has the following quantitative characteristics:

- number of images: 54.58 million (about 200,000 were removed by the data cleaning process);
- storage space: 245.3 GB for image descriptors, 54.14 GB for the image content index, 355.5 GB for thumbnails;
- on average, each photo is described by 3.1 textual tags, has been viewed 42 times by Flickr users, and received 0.53 user comments.

Given the effort required in building such test collection, and the potential interest to the international research community, to make experiments in large-scale CBIR, we decided to make it available outside the SAPIR project scope.

The result is the CoPhIR (Content-based Photo Image Retrieval) Test Collection, managed by ISTI-CNR research institute in Pisa. The data collected so far represents the world largest multimedia metadata collection available for research purposes, with the target to reach 100 million images till the end of 2008. Each entry of the CoPhIR collection is an XML structure containing:

- link to the corresponding image on the Flickr Web site,
- the thumbnail of the photo image,
- the photo textual metadata: author, title, location, GPS, tags, comments, views, etc.,
- an XML sub-structure with 5 standard MPEG-7 visual descriptors.

Note that our use of the Flickr image content is compliant to the most restrictive Creative Commons license. Moreover, the CoPhiIR test collection complies to the European Recommendation 29-2001 CE, based on WIPO (World Intellectual Property Organization) Copyright Treaty and Performances and Phonograms Treaty, and to the current Italian law 68-2003. The scientific organizations (universities, research labs, etc.) interested in experiments on CoPhIR have to register at the CoPhIR Web site[10] and to sign the CoPhIR Access Agreement establishing conditions and terms of use for the collection.

[10] http://cophir.isti.cnr.it

Moreover, any experimental system build on the CoPhIR test collection should provide on the user interface, when displaying Flickr images or thumbnails, an acknowledgment that the original comes from the site www.flikr.com and all rights are reserved to the author of the original image. In the agreement are stated several conditions on the experimental applications, e.g., if the application finds that the original image is no more available (deleted or made private) the entry should be removed from the index.

3 Conclusions

No doubts that the scalability problem for new digital data types is real, which can be nicely illustrated by difficulties with the management of the fast growing digital image collections. In this paper, we focus on two strictly related challenges of scalability: (1) to obtain a non-trivial collection of images with the corresponding descriptive features, and (2) to develop indexing and searching mechanisms able to scale to the target size.

We have crawled a collection of over 50 million high-quality digital images, which is almost two orders of magnitude larger in size than existing image databases used for content-base retrieval and analysis. The images were taken from the Flicker photo-sharing system which has the advantage of being a reliable long-term repository of images and which offers quite a rich set of additional metadata. Using a GRID technology, we have extracted five descriptive features for each image. The features are defined in MPEG-7 standard and express a visual essence of each image in terms of colors, shape, and texture. This information is kept handy in XML files – one for each image – together with the metadata and links to original images in Flicker. We also store thumbnails of the original images, so the search engines built for this dataset can quickly show the overview of the results. This unique collection will be opened to the research community for experiments and comparisons.

Using this image test collection, we have proved that the distributed content-based retrieval system developed in the SAPIR project can scale to tens of millions of objects. The system offers a search for visually-similar images giving answers in approximately one second on our current database of 50 million images. The presented technology is highly flexible as it is based on the metric space model of similarity and on the principles of structured peer-to-peer networks. We would like that similar experiments could be performed in different projects, by different research groups, on the same image test collection, so that results could be usefully compared.

In the future, we plan to continue in the process of crawling and indexing images in order to reach a boundary of 100 million objects. We also plan to investigate application of additional descriptive features of the images and research the relevance feedback to further improve effectiveness.

References

[1] G. Amato, F. Falchi, C. Gennaro, F. Rabitti, P. Savino, and P. Stanchev. Improving image similarity search effectiveness in a multimedia content management system. In *Proc. of Workshop on Multimedia Information System (MIS)*, pages 139–146, 2004.

[2] R. A. Baeza-Yates, J. R. del Solar, R. Verschae, C. Castillo, and C. A. Hurtado. Content-based image retrieval and characterization on specific web collections. volume 3115 of *LNCS*, pages 189–198, 2004.

[3] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 2008. To appear.

[4] ISO/IEC. Information technology - Multimedia content description interfaces. Part 6: Reference Software, 2003. 15938-6:2003.

[5] R. Kumar, J. Novak, and A. Tomkins. Structure and evolution of online social networks. In *KDD '06: Proceedings of the 12th ACM SIGKDD International Conference on Knowedge Discover and Data Mining*, pages 611–617. ACM Press, 2006.

[6] B. Manjunath, P. Salembier, and T. Sikora, editors. *Introduction to MPEG-7: Multimedia Content Description Interface*. John Wiley & Sons, Inc., New York, NY, USA, 2002.

[7] MPEG-7. Multimedia content description interfaces. Part 3: Visual. ISO/IEC 15938-3:2002, 2002.

[8] P. Zezula, G. Amato, V. Dohnal, and M. Batko. *Similarity Search: The Metric Space Approach*, volume 32 of *Advances in Database Systems*. Springer-Verlag, 2006.

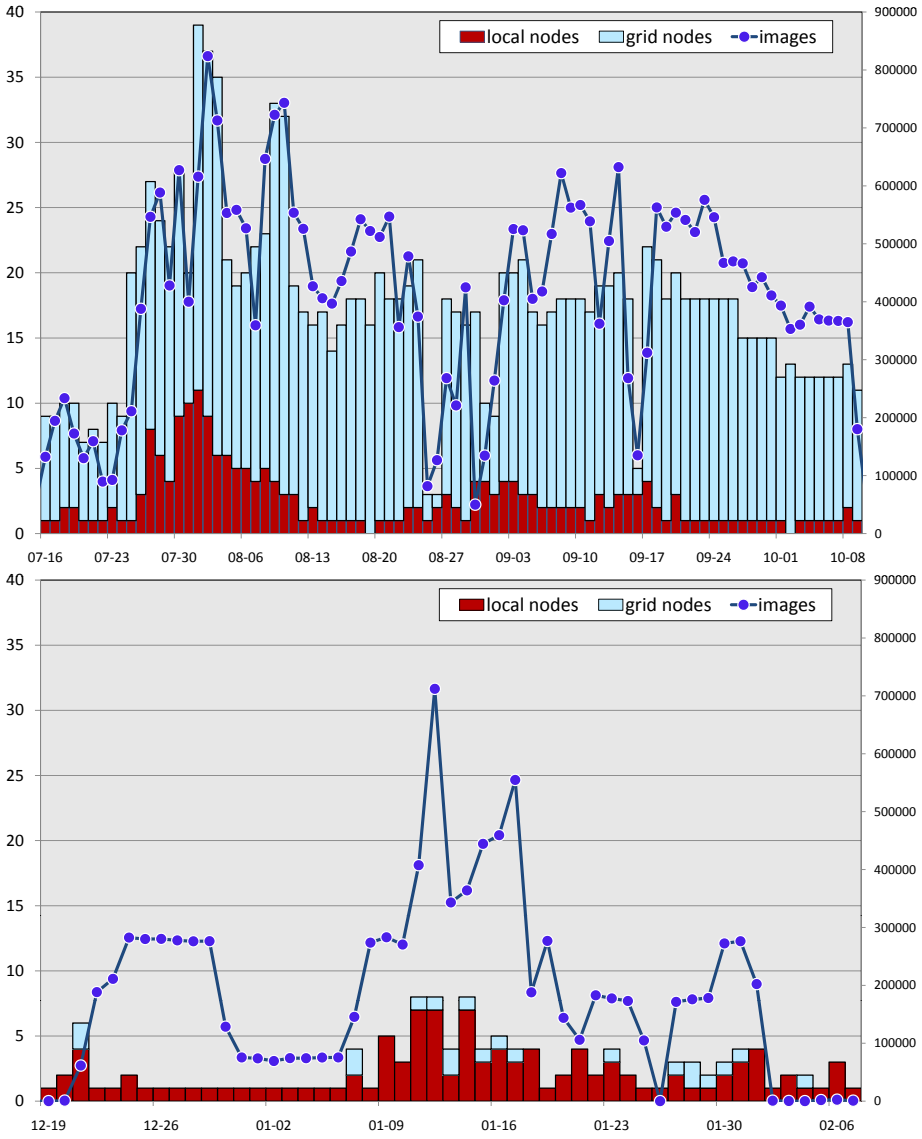

Figure 3: Number of GRID and local machines available during the two crawling periods: from July 16^{th} to October 9^{th} 2007 (left) and from December 19^{th} 2007 to February 7^{th} 2008 (right).

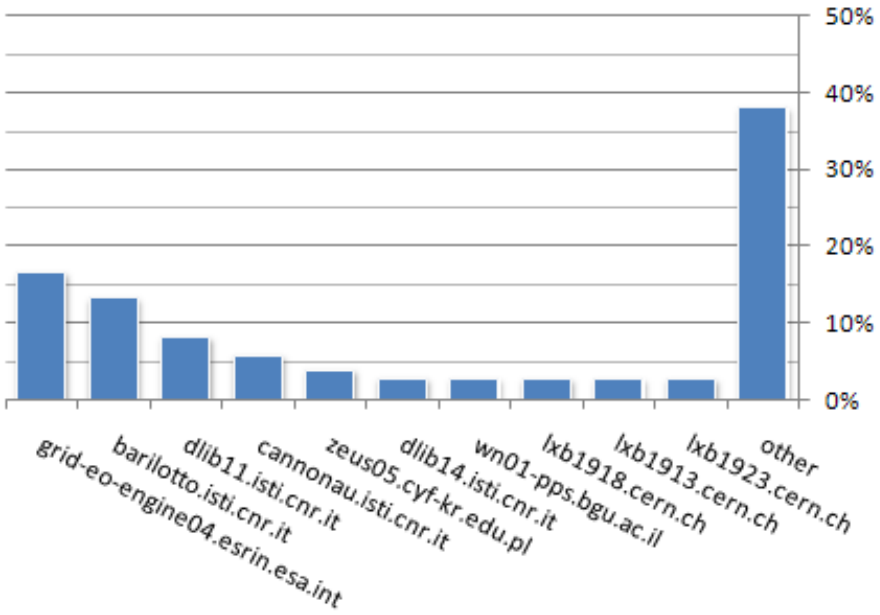

Figure 4: Number of images processed by each site.