**Searching in large image databases**

- state of the art -

The idea of this project (build an image retrieval system) came from the fact that very few systems exist that allows users to query an image database via images. One big player in image retrieval system is Google which uses a set of algorithms to analyze image attributes such as color, shape, texture along side with keywords to refine searches on extremely large collections.

In article [1] they present and implement an exploration system for large image databases that allows users to find, in an efficient way, similar images from a query image. The similarity level of images is given by the color information. The system is built in three stages:

1. Feature extraction stage in which images are represented in a way that allows efficient storage and fast results retrieval.
2. Clustering the images using k-means clustering in which the clustroid would allow quick human comprehension of the type of images within the corresponding cluster
3. Display the results to the user

The system performance was evaluated based on the retrieval accuracy, on the perceptual similarity order among retrieved images and time complexity. For that, a database of 2100 images was used. Time complexity for retrieving results can be reduced by using a hierarchy of clusters.

|  |  |  |
| --- | --- | --- |
| **Clusters** | **1** | **37** |
| **Query Space** | 1382 | 74 |
| **Feature Space** | 265344 | 14208 |
| **Time to Cluster (seconds)** | 0.296838 | 1.19904 |
| **Time to seach (seconds)** | 0.080869 | 0.003182 |
| **Time to extract features in 1 image (seconds)** | 1.879 | 1.09525 |
| **Compare images per second** | 17089.37 | 23255.81 |

Table 1: System timing data comparing the clustering approach with no clustering

|  |  |
| --- | --- |
|  | **Time (seconds)** |
| **Reading Features vectors** | 0.822919 |
| **Computing Features Vectors** | 1513.636 |
| **Reading distance matrix** | 9.87374 |
| **Computing Distance matrix** | 64.12314 |

Table 3: System Performance timing data for loading/computing the off-line information

|  |  |
| --- | --- |
| **Description** | **Percentage** |
| **% all retrieved color similar** | 100% |
| **% all retrieved semantic similar** | 78% |
| **% highest rank semantic similar** | 91% |
| **% at least 1 semantic similar** | 100% |

Table 5: Query retrieval performance results

In article [2] they propose a system similar to the one in [1], differing in the algorithm used for feature extraction: FCH or CVAAS. For the results, on the average, the total time to execute 500 queries is 42.67 seconds for the FCH based image retrieval system, 28.89 seconds for the CVAAS based image retrieval system, and 27.53 seconds for the CVPWIS based image retrieval system in these experiments

Technologies available:

1. **OpenCV**: OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library written in C and CUDA that allows the use of GPU.
2. **JavaEE**: it is a platform provides an API and runtime environment for developing and running large-scale, multi-tiered, scalable, reliable, and secure network applications. In our project we use it to develop the web application and to communicate between modules.
3. **Spring**: a MVC framework meant to simplify the process of developing the web application.
4. **Hibernate**: an object-relational mapping framework for the Java language. It provides a framework for mapping an object-oriented domain model to a relational database. Hibernate solves object-relational impedance mismatch problems by replacing direct, persistent database accesses with high-level object handling functions.
5. **JSF/Primefaces**: JavaServer Faces (JSF) is a Java specification for building component-based user interfaces for web applications.

PrimeFaces is an open source user interface (UI) component library for JavaServer Faces (JSF) based applications, created by PrimeTek, Turkey.

1. **Oracle Database**: database to store and retrieve related information such as image link, path to local stored feature vector for a specific image, account information of users.

Articles and books:

[1] “Searching Large Image Databases using Color Information”, Ioan Raicu, Department of Computer Science, University of Chicago. CMSC350: Artificial Intelligence – Winter Quarter 2004.

[2] “An image retrieval system based on the color complexity if images”, Yung-Kuan Chan, Department of Management Information Systems, Yi-Tung Liu, Rung-Ching Chen, Department of Information Management, Chaoyang University of Technology, Computing and Informatics, Vol. 24, 2005, 495–511.

[3] “State-of-the-Art in Content-Based Image and Video Retrieval”, [Remco C. Veltkamp](http://link.springer.com/search?facet-creator=%22Remco+C.+Veltkamp%22),  [Hans Burkhardt](http://link.springer.com/search?facet-creator=%22Hans+Burkhardt%22), [Hans-Peter Kriegel](http://link.springer.com/search?facet-creator=%22Hans-Peter+Kriegel%22). [Computational Imaging and Vision](http://link.springer.com/bookseries/5754) Volume 22 2001.